
Digital
Communications
John G. Proakis

CONTENTS

1

INTRODUCTION

In this book, we present the basic principles that underlie the analysis and design of digital communication systems. The subject of digital communications involves the transmission of information in digital form from a source that generates the information to one or more destinations. Of particular importance in the analysis and design of communication systems are the characteristics of the physical channels through which the information is transmitted. The characteristics of the channel generally affect the design of the basic building blocks of the communication system. Below, we describe the elements of a communication system and their functions.

1-1 ELEMENTS OF A DIGITAL COMMUNICATION SYSTEM

Figure 1-1-1 illustrates the functional diagram and the basic elements of a digital communication system. The source output may be either an analog signal, such as audio or video signal, or a digital signal, such as the output of a teletype machine, that is discrete in time and has a finite number of output characters. In a digital communication system, the messages produced by the source are converted into a sequence of binary digits. Ideally, we should like to represent the source output (message) by as few binary digits as possible. In other words, we seek an efficient representation of the source output that results in little or no redundancy. The process of efficiently converting the output of either an analog or digital source into a sequence of binary digits is called *source encoding* or *data compression.*

The sequence of binary digits from the source encoder, which we call the

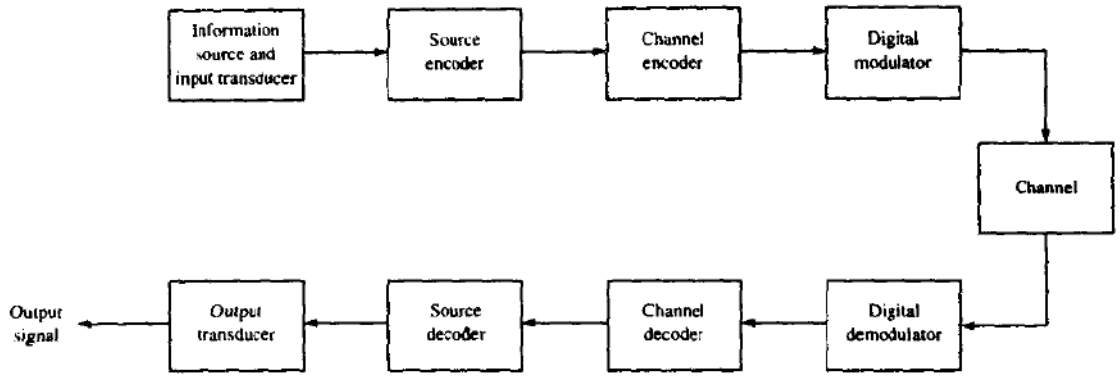


FIGURE 1-1-1 Basic elements of a digital communication system.

information sequence, is passed to the *channel encoder.* The purpose of the channel encoder is to introduce, in a controlled manner, some redundancy in the binary information sequence that can be used at the receiver to overcome the effects of noise and interference encountered in the transmission of the signal through the channel. Thus, the added redundancy serves to increase the reliability of the received data and improves the fidelity of the received signal. In effect, redundancy in the information sequence aids the receiver in decoding the desired information sequence. For example, a (trivial) form of encoding of the binary information sequence is simply to repeat each binary digit m times, where m is some positive integer. More sophisticated (nontrivial) encoding involves taking k information bits at a time and mapping each k-bit sequence into a unique n-bit sequence, called a *code word.* The amount of redundancy introduced by encoding the data in this manner is measured by the ratio n/k. The reciprocal of this ratio, namely k/n, is called the rate of the code or, simply, the *code rate.*

The binary sequence at the output of the channel encoder is passed to the *digital modulator,* which serves as the interface to the communications channel. Since nearly all of the communication channels encountered in practice are capable of transmitting electrical signals (waveforms), the primary purpose of the digital modulator is to map the binary information sequence into signal waveforms. To elaborate on this point, let us suppose that the coded information sequence is to be transmitted one bit at a time at some uniform rate R bits/s. The digital modulator may simply map the binary digit 0 into a waveform $s_0(t)$ and the binary digit 1 into a waveform $s_1(t)$. In this manner, each bit from the channel encoder is transmitted separately. We call this *binary modulation.* Alternatively, the modulator may transmit b coded information bits at a time by using $M = 2^b$ distinct waveforms $s_i(t)$, $i = 0, 1, \ldots, M-1$, one waveform for each of the 2^b possible b-bit sequences. We call this *M-ary modulation* ($M > 2$). Note that a new b-bit sequence enters the modulator

every b/R seconds. Hence, when the channel bit rate R is fixed, the amount of time available to transmit one of the M waveforms corresponding to a b-bit sequence is b times the time period in a system that uses binary modulation.

The *communication channel* is the physical medium that is used to send the signal from the transmitter to the receiver. In wireless transmission, the channel may be the atmosphere (free space). On the other hand, telephone channels usually employ a variety of physical media, including wire lines, optical fiber cables, and wireless (microwave radio). Whatever the physical medium used for transmission of the information, the essential feature is that the transmitted signal is corrupted in a random manner by a variety of possible mechanisms, such as additive *thermal noise* generated by electronic devices, man-made noise, e.g., automobile ignition noise, and atmospheric noise, e.g., electrical lightning discharges during thunderstorms.

At the receiving end of a digital communications system, the *digital demodulator* processes the channel-corrupted transmitted waveform and reduces the waveforms to a sequence of numbers that represent estimates of the transmitted data symbols (binary or M-ary). This sequence of numbers is passed to the channel decoder, which attempts to reconstruct the original information sequence from knowledge of the code used by the channel encoder and the redundancy contained in the received data.

A measure of how well the demodulator and decoder perform is the frequency with which errors occur in the decoded sequence. More precisely, the average probability of a bit-error at the output of the decoder is a measure of the performance of the demodulator–decoder combination. In general, the probability of error is a function of the code characteristics, the types of waveforms used to transmit the information over the channel, the transmitter power, the characteristics of the channel, i.e., the amount of noise, the nature of the interference, etc., and the method of demodulation and decoding. These items and their effect on performance will be discussed in detail in subsequent chapters.

As a final step, when an analog output is desired, the source decoder accepts the output sequence from the channel decoder and, from knowledge of the source encoding method used, attempts to reconstruct the original signal from the source. Due to channel decoding errors and possible distortion introduced by the source encoder and, perhaps, the source decoder, the signal at the output of the source decoder is an approximation to the original source output. The difference or some function of the difference between the original signal and the reconstructed signal is a measure of the distortion introduced by the digital communication system.

1-2 COMMUNICATION CHANNELS AND THEIR CHARACTERISTICS

As indicated in the preceding discussion, the communication channel provides the connection between the transmitter and the receiver. The physical channel

may be a pair of wires that carry the electrical signal, or an optical fiber that carries the information on a modulated light beam, or an underwater ocean channel in which the information is transmitted acoustically, or free space over which the information-bearing signal is radiated by use of an antenna. Other media that can be characterized as communication channels are data storage media, such as magnetic tape, magnetic disks, and optical disks.

One common problem in signal transmission through any channel is additive noise. In general, additive noise is generated internally by components such as resistors and solid-state devices used to implement the communication system. This is sometimes called *thermal noise*. Other sources of noise and interference may arise externally to the system, such as interference from other users of the channel. When such noise and interference occupy the same frequency band as the desired signal, its effect can be minimized by proper design of the transmitted signal and its demodulator at the receiver. Other types of signal degradations that may be encountered in transmission over the channel are signal attenuation, amplitude and phase distortion, and multipath distortion.

The effects of noise may be minimized by increasing the power in the transmitted signal. However, equipment and other practical constraints limit the power level in the transmitted signal. Another basic limitation is the available channel bandwidth. A bandwidth constraint is usually due to the physical limitations of the medium and the electronic components used to implement the transmitter and the receiver. These two limitations result in constraining the amount of data that can be transmitted reliably over any communications channel as we shall observe in later chapters. Below, we describe some of the important characteristics of several communication channels.

Wireline Channels The telephone network makes extensive use of wire lines for voice signal transmission, as well as data and video transmission. Twisted-pair wire lines and coaxial cable are basically guided electromagnetic channels that provide relatively modest bandwidths. Telephone wire generally used to connect a customer to a central office has a bandwidth of several hundred kilohertz (kHz). On the other hand, coaxial cable has a usable bandwidth of several megahertz (MHz). Figure 1-2-1 illustrates the frequency range of guided electromagnetic channels, which include waveguides and optical fibers.

Signals transmitted through such channels are distored in both amplitude and phase and further corrupted by additive noise. Twisted-pair wireline channels are also prone to crosstalk interference from physically adjacent channels. Because wireline channels carry a large percentage of our daily communications around the country and the world, much research has been performed on the characterization of their transmission properties and on methods for mitigating the amplitude and phase distortion encountered in signal transmission. In Chapter 9, we describe methods for designing optimum transmitted signals and their demodulation; in Chapters 10 and 11, we

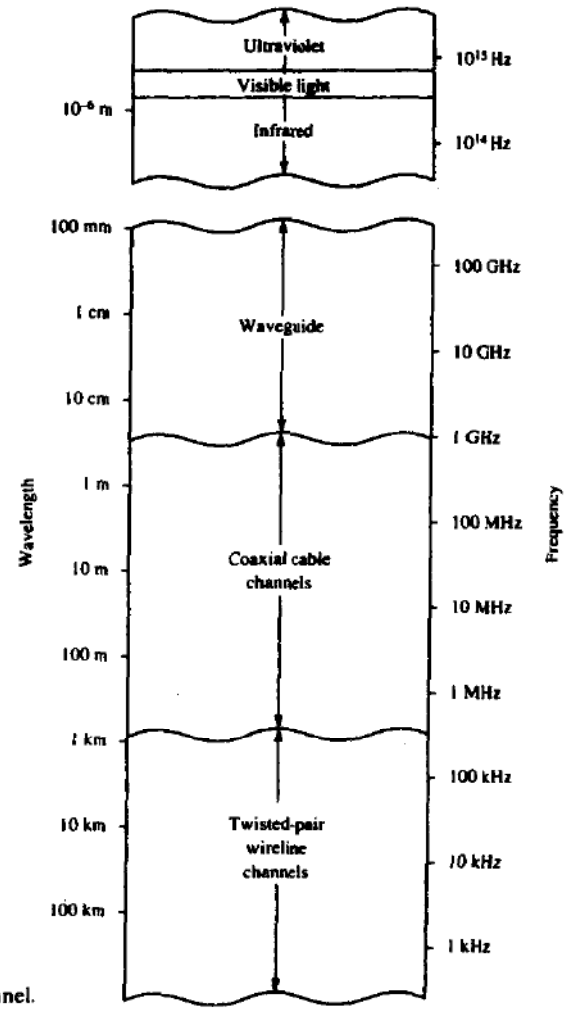


FIGURE 1-2-1 Frequency range for guided wire channel.

consider the design of channel equalizers that compensate for amplitude and phase distortion on these channels.

Fiber Optic Channels Optical fibers offer the communications system designer a channel bandwidth that is several orders of magnitude larger than coaxial cable channels. During the past decade, optical fiber cables have been developed that have a relatively low signal attenuation, and highly reliable photonic devices have been developed for signal generation and signal detection. These technological advances have resulted in a rapid deployment of optical fiber channels, both in domestic telecommunication systems as well as for trans-Atlantic and trans-Pacific communications. With the large bandwidth

available on fiber optic channels, it is possible for telephone companies to offer subscribers a wide array of telecommunication services, including voice, data, facsimile, and video.

The transmitter or modulator in a fiber optic communication system is a light source, either a light-emitting diode (LED) or a laser. Information is transmitted by varying (modulating) the intensity of the light source with the message signal. The light propagates through the fiber as a light wave and is amplified periodically (in the case of digital transmission, it is detected and regenerated by repeaters) along the transmission path to compensate for signal attenuation. At the receiver, the light intensity is detected by a photodiode, whose output is an electrical signal that varies in direct proportion to the power of the light impinging on the photodiode. Sources of noise in fiber optic channels are photodiodes and electronic amplifiers.

It is envisioned that optical fiber channels will replace nearly all wireline channels in the telephone network by the turn of the century.

Wireless Electromagnetic Channels In wireless communication systems, electromagnetic energy is coupled to the propagation medium by an antenna which serves as the radiator. The physical size and the configuration of the antenna depend primarily on the frequency of operation. To obtain efficient radiation of electromagnetic energy, the antenna must be longer than $\frac{1}{10}$ of the wavelength. Consequently, a radio station transmitting in the AM frequency band, say at $f_c = 1$ MHz (corresponding to a wavelength of $\lambda = c/f_c = 300$ m), requires an antenna of at least 30 m. Other important characteristics and attributes of antennas for wireless transmission are described in Chapter 5.

Figure 1-2-2 illustrates the various frequency bands of the electromagnetic spectrum. The mode of propagation of electromagnetic waves in the atmosphere and in free space may be subdivided into three categories, namely, ground-wave propagation, sky-wave propagation, and line-of-sight (LOS) propagation. In the VLF and audio frequency bands, where the wavelengths exceed 10 km, the earth and the ionosphere act as a waveguide for electromagnetic wave propagation. In these frequency ranges, communication signals practically propagate around the globe. For this reason, these frequency bands are primarily used to provide navigational aids from shore to ships around the world. The channel bandwidths available in these frequency bands are relatively small (usually 1–10% of the center frequency), and hence the information that is transmitted through these channels is of relatively slow speed and generally confined to digital transmission. A dominant type of noise at these frequencies is generated from thunderstorm activity around the globe, especially in tropical regions. Interference results from the many users of these frequency bands.

Ground-wave propagation, as illustrated in Fig. 1-2-3, is the dominant mode of propagation for frequencies in the MF band (0.3–3 MHz). This is the frequency band used for AM broadcasting and maritime radio broadcasting. In AM broadcasting, the range with groundwave propagation of even the more

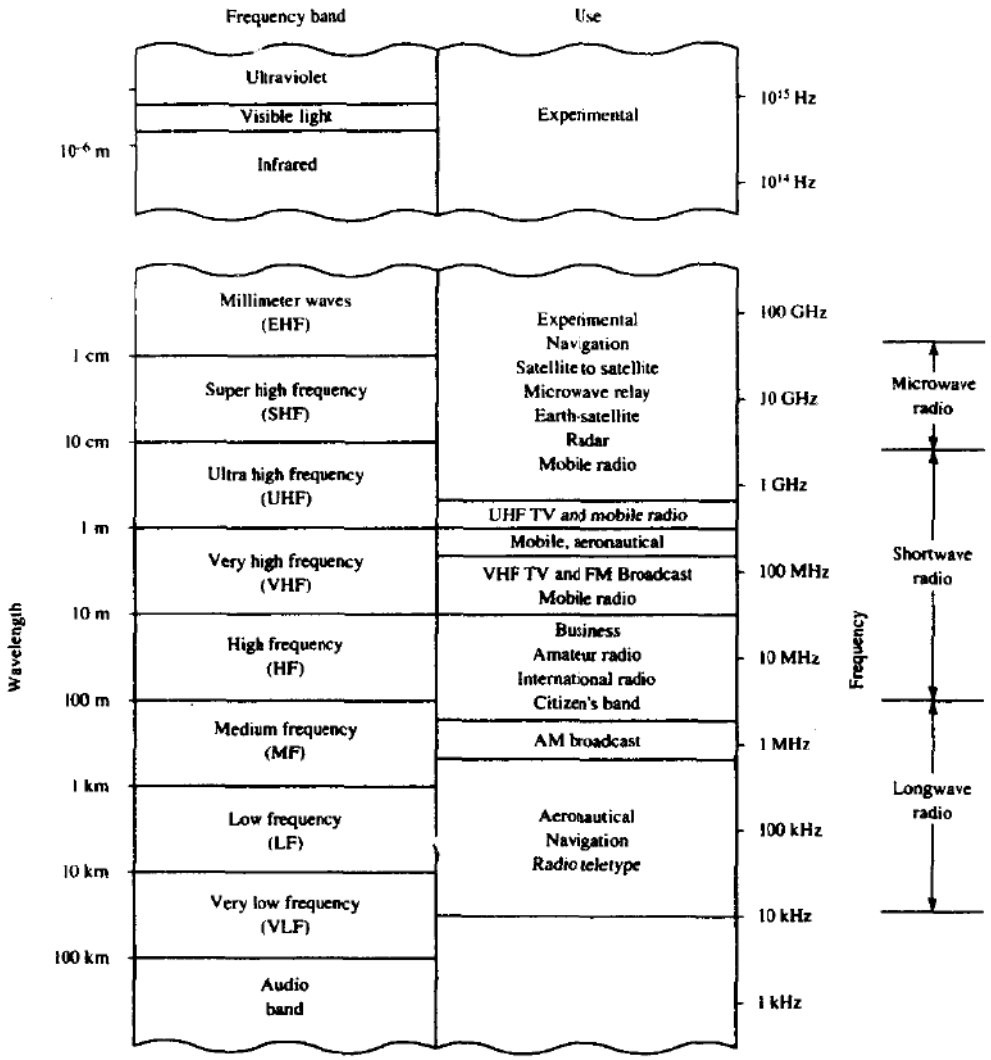


FIGURE 1-2-2 Frequency range for wireless electromagnetic channels. [*Adapted from Carlson (1975), 2nd edition, © McGraw-Hill Book Company Co. Reprinted with permission of the publisher.*]

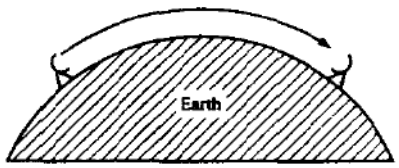


FIGURE 1-2-3 Illustration of ground-wave propagation.

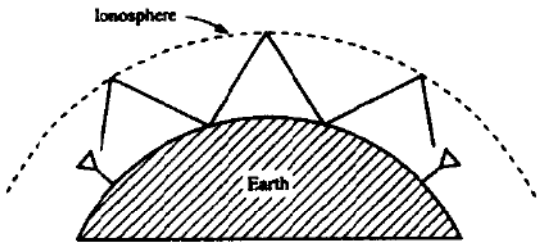


FIGURE 1-2-4 Illustration of sky-wave propagation.

powerful radio stations is limited to about 150 km. Atmospheric noise, man-made noise, and thermal noise from electronic components at the receiver are dominant disturbances for signal transmission in the MF band.

Sky-wave propagation, as illustrated in Fig. 1-2-4 results from transmitted signals being reflected (bent or refracted) from the ionosphere, which consists of several layers of charged particles ranging in altitude from 50 to 400 km above the surface of the earth. During the daytime hours, the heating of the lower atmosphere by the sun causes the formation of the lower layers at altitudes below 120 km. These lower layers, especially the D-layer, serve to absorb frequencies below 2 MHz, thus severely limiting sky-wave propagation of AM radio broadcast. However, during the night-time hours, the electron density in the lower layers of the ionosphere drops sharply and the frequency absorption that occurs during the daytime is significantly reduced. As a consequence, powerful AM radio broadcast stations can propagate over large distances via sky wave over the F-layer of the ionosphere, which ranges from 140 to 400 km above the surface of the earth.

A frequently occurring problem with electromagnetic wave propagation via sky wave in the HF frequency range is *signal multipath*. Signal multipath occurs when the transmitted signal arrives at the receiver via multiple propagation paths at different delays. It generally results in intersymbol interference in a digital communication system. Moreover, the signal components arriving via different propagation paths may add destructively, resulting in a phenomenon called *signal fading,* which most people have experienced when listening to a distant radio station at night when sky wave is the dominant propagation mode. Additive noise at HF is a combination of atmospheric noise and thermal noise.

Sky-wave ionospheric propagation ceases to exist at frequencies above approximately 30 MHz, which is the end of the HF band. However, it is possible to have ionospheric scatter propagation at frequencies in the range 30–60 MHz, resulting from signal scattering from the lower ionosphere. It is also possible to communicate over distances of several hundred miles by use of tropospheric scattering at frequencies in the range 40–300 MHz. Troposcatter results from signal scattering due to particles in the atmosphere at altitudes of 10 miles or less. Generally, ionospheric scatter and tropospheric scatter

involve large signal propagation losses and require a large amount of transmitter power and relatively large antennas.

Frequencies above 30 MHz propagate through the ionosphere with relatively little loss and make satellite and extraterrestrial communications possible. Hence, at frequencies in the VHF band and higher, the dominant mode of electromagnetic propagation is line-of-sight (LOS) propagation. For terrestrial communication systems, this means that the transmitter and receiver antennas must be in direct LOS with relatively little or no obstruction. For this reason, television stations transmitting in the VHF and UHF frequency bands mount their antennas on high towers to achieve a broad coverage area.

In general, the coverage area for LOS propagation is limited by the curvature of the earth. If the transmitting antenna is mounted at a height h m above the surface of the earth, the distance to the radio horizon, assuming no physical obstructions such as mountains, is approximately $d = \sqrt{15h}$ km. For example, a TV antenna mounted on a tower of 300 m in height provides a coverage of approximately 67 km. As another example, microwave radio relay systems used extensively for telephone and video transmission at frequencies above 1 GHz have antennas mounted on tall towers or on the top of tall buildings.

The dominant noise limiting the performance of a communication system in VHF and UHF frequency ranges is thermal noise generated in the receiver front end and cosmic noise picked up by the antenna. At frequencies in the SHF band above 10 GHz, atmospheric conditions play a major role in signal propagation. For example, at 10 GHz, the attenuation ranges from about 0.003 dB/km in light rain to about 0.3 dB/km in heavy rain. At 100 GHz, the attenuation ranges from about 0.1 dB/km in light rain to about 6 dB/km in heavy rain. Hence, in this frequency range, heavy rain introduces extremely high propagation losses that can result in service outages (total breakdown in the communication system).

At frequencies above the EHF (extremely high frequency) band, we have the infrared and visible light regions of the electromagnetic spectrum, which can be used to provide LOS optical communication in free space. To date, these frequency bands have been used in experimental communication systems, such as satellite-to-satellite links.

Underwater Acoustic Channels Over the past few decades, ocean exploration activity has been steadily increasing. Coupled with this increase is the need to transmit data, collected by sensors placed under water, to the surface of the ocean. From there, it is possible to relay the data via a satellite to a data collection center.

Electromagnetic waves do not propagate over long distances under water except at extremely low frequencies. However, the transmission of signals at such low frequencies is prohibitively expensive because of the large and powerful transmitters required. The attenuation of electromagnetic waves in water can be expressed in terms of the *skin depth*, which is the distance a signal is attenuated by $1/e$. For sea water, the skin depth $\delta = 250/\sqrt{f}$, where f is

expressed in Hz and δ is in m. For example, at 10 kHz, the skin depth is 2.5 m. In contrast, acoustic signals propagate over distances of tens and even hundreds of kilometers.

An underwater acoustic channel is characterized as a multipath channel due to signal reflections from the surface and the bottom of the sea. Because of wave motion, the signal multipath components undergo time-varying propagation delays that result in signal fading. In addition, there is frequency-dependent attenuation, which is approximately proportional to the square of the signal frequency. The sound velocity is nominally about 1500 m/s, but the actual value will vary either above or below the nominal value depending on the depth at which the signal propagates.

Ambient ocean acoustic noise is caused by shrimp, fish, and various mammals. Near harbors, there is also man-made acoustic noise in addition to the ambient noise. In spite of this hostile environment, it is possible to design and implement efficient and highly reliable underwater acoustic communication systems for transmitting digital signals over large distances.

Storage Channels Information storage and retrieval systems constitute a very significant part of data-handling activities on a daily basis. Magnetic tape, including digital audio tape and video tape, magnetic disks used for storing large amounts of computer data, optical disks used for computer data storage, and compact disks are examples of data storage systems that can be characterized as communication channels. The process of storing data on a magnetic tape or a magnetic or optical disk is equivalent to transmitting a signal over a telephone or a radio channel. The readback process and the signal processing involved in storage systems to recover the stored information are equivalent to the functions performed by a receiver in a telephone or radio communication system to recover the transmitted information.

Additive noise generated by the electronic components and interference from adjacent tracks is generally present in the readback signal of a storage system, just as is the case in a telephone or a radio communication system.

The amount of data that can be stored is generally limited by the size of the disk or tape and the density (number of bits stored per square inch) that can be achieved by the write/read electronic systems and heads. For example, a packing density of 10^9 bits per square inch has been recently demonstrated in an experimental magnetic disk storage system. (Current commercial magnetic storage products achieve a much lower density.) The speed at which data can be written on a disk or tape and the speed at which it can be read back are also limited by the associated mechanical and electrical subsystems that constitute an information storage system.

Channel coding and modulation are essential components of a well-designed digital magnetic or optical storage system. In the readback process, the signal is demodulated and the added redundancy introduced by the channel encoder is used to correct errors in the readback signal.

1-3 MATHEMATICAL MODELS FOR COMMUNICATION CHANNELS

In the design of communication systems for transmitting information through physical channels, we find it convenient to construct mathematical models that reflect the most important characteristics of the transmission medium. Then, the mathematical model for the channel is used in the design of the channel encoder and modulator at the transmitter and the demodulator and channel decoder at the receiver. Below, we provide a brief description of the channel models that are frequently used to characterize many of the physical channels that we encounter in practice.

The Additive Noise Channel The simplest mathematical model for a communication channel is the additive noise channel, illustrated in Fig. 1-3-1. In this model, the transmitted signal $s(t)$ is corrupted by an additive random noise process $n(t)$. Physically, the additive noise process may arise from electronic components and amplifiers at the receiver of the communication system, or from interference encountered in transmission (as in the case of radio signal transmission).

If the noise is introduced primarily by electronic components and amplifiers at the receiver, it may be characterized as thermal noise. This type of noise is characterized statistically as a *gaussian noise process*. Hence, the resulting mathematical model for the channel is usually called the *additive gaussian noise channel*. Because this channel model applies to a broad class of physical communication channels and because of its mathematical tractability, this is the predominant channel model used in our communication system analysis and design. Channel attenuation is easily incorporated into the model. When the signal undergoes attenuation in transmission through the channel, the received signal is

$$r(t) = \alpha s(t) + n(t) \tag{1-3-1}$$

where α is the attenuation factor.

The Linear Filter Channel In some physical channels, such as wireline telephone channels, filters are used to ensure that the transmitted signals do not exceed specified bandwidth limitations and thus do not interfere with one another. Such channels are generally characterized mathematically as linear filter channels with additive noise, as illustrated in Fig. 1-3-2. Hence, if the

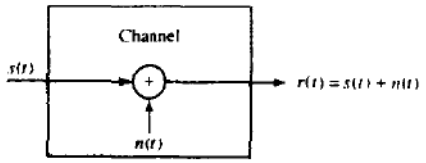


FIGURE 1-3-1 The additive noise channel.

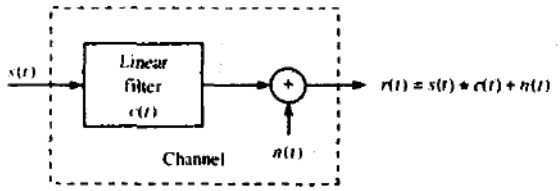


FIGURE 1-3-2 The linear filter channel with additive noise.

channel input is the signal $s(t)$, the channel output is the signal

$$r(t) = s(t) \bigstar c(t) + n(t)$$
$$= \int_{-\infty}^{\infty} c(\tau)s(t-\tau)\,d\tau + n(t) \tag{1-3-2}$$

where $c(t)$ is the impulse response of the linear filter and $\bigstar$ denotes convolution.

The Linear Time-Variant Filter Channel Physical channels such as underwater acoustic channels and ionospheric radio channels that result in time-variant multipath propagation of the transmitted signal may be characterized mathematically as time-variant linear filters. Such linear filters are characterized by a time-variant channel impulse response $c(\tau; t)$, where $c(\tau; t)$ is the response of the channel at time t due to an impulse applied at time $t - \tau$. Thus, τ represents the "age" (elapsed-time) variable. The linear time-variant filter channel with additive noise is illustrated in Fig. 1-3-3. For an input signal $s(t)$, the channel output signal is

$$r(t) = s(t) \bigstar c(\tau; t) + n(t)$$
$$= \int_{-\infty}^{\infty} c(\tau; t)s(t-\tau)\,d\tau + n(t) \tag{1-3-3}$$

A good model for multipath signal propagation through physical channels, such as the ionosphere (at frequencies below 30 MHz) and mobile cellular radio channels, is a special case of (1-3-3) in which the time-variant impulse response has the form

$$c(\tau; t) = \sum_{k=1}^{L} a_k(t)\delta(\tau - \tau_k) \tag{1-3-4}$$

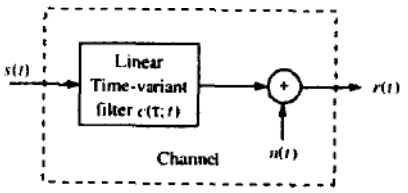


FIGURE 1-3-3 Linear time-variant filter channel with additive noise.

where the $\{a_k(t)\}$ represents the possibly time-variant attenuation factor for the L multipath propagation paths and $\{\tau_k\}$ are the corresponding time delays. If (1-3-4) is substituted into (1-3-3), the received signal has the form

$$r(t) = \sum_{k=1}^{L} a_k(t)s(t - \tau_k) + n(t) \tag{1-3-5}$$

Hence, the received signal consists of L multipath components, where each component is attenuated by $\{a_k(t)\}$ and delayed by $\{\tau_k\}$.

The three mathematical models described above adequately characterize the great majority of the physical channels encountered in practice. These three channel models are used in this text for the analysis and design of communication systems.

1-4 A HISTORICAL PERSPECTIVE IN THE DEVELOPMENT OF DIGITAL COMMUNICATIONS

It is remarkable that the earliest form of electrical communication, namely *telegraphy*, was a digital communication system. The electric telegraph was developed by Samuel Morse and was demonstrated in 1837. Morse devised the variable-length binary code in which letters of the English alphabet are represented by a sequence of dots and dashes (code words). In this code, more frequently occurring letters are represented by short code words, while letters occurring less frequently are represented by longer code words. Thus, the *Morse code* was the precursor of the variable-length source coding methods described in Chapter 3.

Nearly 40 years later, in 1875, Emile Baudot devised a code for telegraphy in which every letter was encoded into fixed-length binary code words of length 5. In the *Baudot code*, binary code elements are of equal length and designated as mark and space.

Although Morse is responsible for the development of the first electrical digital communication system (telegraphy), the beginnings of what we now regard as modern digital communications stem from the work of Nyquist (1924), who investigated the problem of determining the maximum signaling rate that can be used over a telegraph channel of a given bandwidth without intersymbol interference. He formulated a model of a telegraph system in which a transmitted signal has the general form

$$s(t) = \sum_{n} a_n g(t - nT) \tag{1-4-1}$$

where $g(t)$ represents a basic pulse shape and $\{a_n\}$ is the binary data sequence of $\{\pm 1\}$ transmitted at a rate of $1/T$ bits/s. Nyquist set out to determine the optimum pulse shape that was bandlimited to W Hz and maximized the bit rate under the constraint that the pulse caused no intersymbol interference at the

sampling time k/T, $k = 0, \pm 1, \pm 2, \ldots$. His studies led him to conclude that the maximum pulse rate is $2W$ pulses/s. This rate is now called the *Nyquist rate*. Moreover, this pulse rate can be achieved by using the pulses $g(t) = (\sin 2\pi Wt)/2\pi Wt$. This pulse shape allows recovery of the data without intersymbol interference at the sampling instants. Nyquist's result is equivalent to a version of the sampling theorem for bandlimited signals, which was later stated precisely by Shannon (1948). The sampling theorem states that a signal of bandwidth W can be reconstructed from samples taken at the Nyquist rate of $2W$ samples/s using the interpolation formula

$$s(t) = \sum_n s\left(\frac{n}{2W}\right) \frac{\sin [2\pi W(t - n/2W)]}{2\pi W(t - n/2W)} \tag{1-4-2}$$

In light of Nyquist's work, Hartley (1928) considered the issue of the amount of data that can be transmitted reliably over a bandlimited channel when multiple amplitude levels are used. Due to the presence of noise and other interference, Hartley postulated that the receiver can reliably estimate the received signal amplitude to some accuracy, say A_δ. This investigation led Hartley to conclude that there is a maximum data rate that can be communicated reliably over a bandlimited channel when the maximum signal amplitude is limited to $A_{\max}$ (fixed power constraint) and the amplitude resolution is A_δ.

Another significant advance in the development of communications was the work of Wiener (1942), who considered the problem of estimating a desired signal waveform $s(t)$ in the presence of additive noise $n(t)$, based on observation of the received signal $r(t) = s(t) + n(t)$. This problem arises in signal demodulation. Wiener determined the linear filter whose output is the best mean-square approximation to the desired signal $s(t)$. The resulting filter is called the *optimum linear (Wiener) filter.*

Hartley's and Nyquist's results on the maximum transmission rate of digital information were precursors to the work of Shannon (1948a,b), who established the mathematical foundations for information transmission and derived the fundamental limits for digital communication systems. In his pioneering work, Shannon formulated the basic problem of reliable transmission of information in statistical terms, using probabilistic models for information sources and communication channels. Based on such a statistical formulation, he adopted a logarithmic measure for the information content of a source. He also demonstrated that the effect of a transmitter power constraint, a bandwidth constraint, and additive noise can be associated with the channel and incorporated into a single parameter, called the *channel capacity*. For example, in the case of an additive white (spectrally flat) gaussian noise interference, an ideal bandlimited channel of bandwidth W has a capacity C given by

$$C = W \log_2 \left(1 + \frac{P}{WN_0}\right) \text{ bits/s} \tag{1-4-3}$$

where P is the average transmitted power and N_0 is the power spectral density of the additive noise. The significance of the channel capacity is as follows: If the information rate R from the source is less than C ($R < C$) then it is theoretically possible to achieve reliable (error-free) transmission through the channel by appropriate coding. On the other hand, if $R > C$, reliable transmission is not possible regardless of the amount of signal processing performed at the transmitter and receiver. Thus, Shannon established basic limits on communication of information, and gave birth to a new field that is now called *information theory*.

Another important contribution to the field of digital communication is the work of Kotelnikov (1947), who provided a coherent analysis of the various digital communication systems based on a geometrical approach. Kotelnikov's approach was later expanded by Wozencraft and Jacobs (1965).

Following Shannon's publications, came the classic work of Hamming (1950) on error-detecting and error-correcting codes to combat the detrimental effects of channel noise. Hamming's work stimulated many researchers in the years that followed, and a variety of new and powerful codes were discovered, many of which are used today in the implementation of modern communication systems.

The increase in demand for data transmission during the last three to four decades, coupled with the development of more sophisticated integrated circuits, has led to the development of very efficient and more reliable digital communication systems. In the course of these developments, Shannon's original results and the generalization of his results on maximum transmission limits over a channel and on bounds on the performance achieved have served as benchmarks for any given communication system design. The theoretical limits derived by Shannon and other researchers that contributed to the development of information theory serve as an ultimate goal in the continuing efforts to design and develop more efficient digital communication systems.

There have been many new advances in the area of digital communications following the early work of Shannon, Kotelnikov, and Hamming. Some of the most notable developments are the following:

- The development of new block codes by Muller (1954), Reed (1954), Reed and Solomon (1960), Bose and Ray-Chaudhuri (1960a,b), and Goppa (1970, 1971).
- The development of concatenated codes by Forney (1966).
- The development of computationally efficient decoding of BCH codes, e.g., the Berlekamp–Massey algorithm (see Chien, 1964; Berlekamp, 1968).
- The development of convolutional codes and decoding algorithms by Wozencraft and Reiffen (1961), Fano (1963), Zigangirov (1966), Jelinek (1969), Forney (1970, 1972), and Viterbi (1967, 1971).
- The development of trellis-coded modulation by Ungerboeck (1982), Forney *et al.* (1984), Wei (1987), and others.
- The development of efficient source encodings algorithms for data

compression, such as those devised by Ziv and Lempel (1977, 1978) and Linde *et al.* (1980).

1-5 OVERVIEW OF THE BOOK

Chapter 2 presents a brief review of the basic notions in the theory of probability and random processes. Our primary objectives in this chapter are to present results that are used throughout the book and to establish some necessary notation.

In Chapter 3, we provide an introduction to source coding for discrete and analog sources. Included in this chapter are the Huffman coding algorithm and the Lempel–Ziv algorithm for discrete sources, and scalar and vector quantization techniques for analog sources.

Chapter 4 treats the characterization of communication signals and systems from a mathematical viewpoint. Included in this chapter is a geometric representation of signal waveforms used for digital communications.

Chapters 5–8 are focused on modulation/demodulation and channel coding/decoding for the additive, white gaussian noise channel. The emphasis is on optimum demodulation and decoding techniques and their performance.

The design of efficient modulators and demodulators for linear filter channels with distortion is treated in Chapters 9–11. The focus is on signal design and on channel equalization methods to compensate for the channel distortion.

The final four chapters treat several more specialized topics. Chapter 12 treats multichannel and multicarrier communication systems. Chapter 13 is focused on spread spectrum signals for digital communications and their performance characteristics. Chapter 14 provides a in-depth treatment of communication through fading multipath channels. Included in this treatment is a description of channel characterization, signal design and demodulation techniques and their performance, and coding/decoding techniques and their performance. The last chapter of the book is focused on multiuser communication systems and multiple access methods.

1-6 BIBLIOGRAPHICAL NOTES AND REFERENCES

There are several historical treatments regarding the development of radio and telecommunications during the past century. These may be found in the books by McMahon (1984), Millman (1984), and Ryder and Fink (1984). We have already cited the classical works of Nyquist (1924), Hartley (1928), Kotelnikov (1947), Shannon (1948), and Hamming (1950), as well as some of the more important advances that have occurred in the field since 1950. The collected papers by Shannon have been published by IEEE Press in a book edited by Sloane and Wyner (1993). Other collected works published by the IEEE Press that might be of interest to the reader are *Key Papers in the Development of Coding Theory*, edited by Berlekamp (1974), and *Key Papers in the Development of Information Theory*, edited by Slepian (1974).

2

PROBABILITY AND STOCHASTIC PROCESSES

The theory of probability and stochastic processes is an essential mathematical tool in the design of digital communication systems. This subject is important in the statistical modeling of sources that generate the information, in the digitization of the source output, in the characterization of the channel through which the digital information is transmitted, in the design of the receiver that processes the information-bearing signal from the channel, and in the evaluation of the performance of the communication system. Our coverage of this rich and interesting subject is brief and limited in scope. We present a number of definitions and basic concepts in the theory of probability and stochastic processes and we derive several results that are important in the design of efficient digital communication systems and in the evaluation of their performance.

We anticipate that most readers have had some prior exposure to the theory of probability and stochastic processes, so that our treatment serves primarily as a review. Some readers, however, who have had no previous exposure may find the presentation in this chapter extremely brief. These readers will benefit from additional reading of engineering-level treatments of the subject found in the texts by Davenport and Root (1958), Davenport (1970), Papoulis (1984), Helstrom (1991), and Leon-Garcia (1994).

2-1 PROBABILITY

Let us consider an experiment, such as the rolling of a die, with a number of possible outcomes. The sample space S of the experiment consists of the set of all possible outcomes. In the case of the die,

$$S = \{1, 2, 3, 4, 5, 6\} \tag{2-1-1}$$

where the integers $1, \ldots, 6$ represent the number of dots on the six faces of the die. These six possible outcomes are the sample points of the experiment. An event is a subset of S, and may consist of any number of sample points. For example, the event A defined as

$$A = \{2, 4\} \tag{2-1-2}$$

consists of the outcomes 2 and 4. The complement of the event A, denoted by $\bar{A}$, consists of all the sample points in S that are not in A and, hence,

$$\bar{A} = \{1, 3, 5, 6\} \tag{2-1-3}$$

Two events are said to be mutally exclusive if they have no sample points in common—that is, if the occurrence of one event excludes the occurrence of the other. For example, if A is defined as in (2-1-2) and the event B is defined as

$$B = \{1, 3, 6\} \tag{2-1-4}$$

then A and B are mutually exclusive events. Similarly, A and $\bar{A}$ are mutually exclusive events.

The union (sum) of two events is an event that consists of all the sample points in the two events. For example, if B is the event defined in (2-1-4) and C is the event defined as

$$C = \{1, 2, 3\} \tag{2-1-5}$$

then, the union of B and C, denoted by $B \cup C$, is the event

$$\begin{aligned} D &= B \cup C \\ &= \{1, 2, 3, 6\} \end{aligned} \tag{2-1-6}$$

Similarly, $A \cup \bar{A} = S$, where S is the entire sample space or the certain event. On the other hand, the intersection of two events is an event that consists of the points that are common to the two events. Thus, if $E = B \cap C$ represents the intersection of the events B and C, defined by (2-1-4) and (2-1-5), respectively, then

$$E = \{1, 3\}$$

When the events are mutually exclusive, the intersection is the null event, denoted as $\varnothing$. For example, $A \cap B = \varnothing$, and $A \cap \bar{A} = \varnothing$. The definitions of union and intersection are extended to more than two events in a straightforward manner.

Associated with each event A contained in S is its probability $P(A)$. In the assignment of probabilities to events, we adopt an axiomatic viewpoint. That

is, we postulate that the probability of the event A satisfies the condition $P(A) \geq 0$. We also postulate that the probability of the sample space (certain event) is $P(S) = 1$. The third postulate deals with the probability of mutually exclusive events. Suppose that A_i, $i = 1, 2, \ldots$, are a (possibly infinite) number of events in the sample space S such that

$$A_i \cap A_j = \varnothing \quad i \neq j = 1, 2, \ldots$$

Then the probability of the union of these mutually exclusive events satisfies the condition

$$P\left(\bigcup_i A_i\right) = \sum_i P(A_i) \tag{2-1-7}$$

For example, in a roll of a fair die, each possible outcome is assigned the probability $\frac{1}{6}$. The event A defined by (2-1-2) consists of two mutually exclusive subevents or outcomes, and, hence, $P(A) = \frac{2}{6} = \frac{1}{3}$. Also, the probability of the event $A \cup B$, where A and B are the mutually exclusive events defined by (2-1-2) and (2-1-4), respectively, is $P(A) + P(B) = \frac{1}{3} + \frac{1}{2} = \frac{5}{6}$.

Joint Events and Joint Probabilities Instead of dealing with a single experiment, let us perform two experiments and consider their outcomes. For example, the two experiments may be two separate tosses of a single die or a single toss of two dice. In either case, the sample space S consists of the 36 two-tuples (i, j) where $i, j = 1, 2, \ldots, 6$. If the dice are fair, each point in the sample space is assigned the probability $\frac{1}{36}$. We may now consider joint events, such as $\{i \text{ is even}, j = 3\}$, and determine the associated probabilities of such events from knowledge of the probabilities of the sample points.

In general, if one experiment has the possible outcomes A_i, $i = 1, 2, \ldots, n$, and the second experiment has the possible outcomes B_j, $j = 1, 2, \ldots, m$, then the combined experiment has the possible joint outcomes (A_i, B_j), $i = 1, 2, \ldots, n$, $j = 1, 2, \ldots, m$. Associated with each joint outcome (A_i, B_j) is the joint probability $P(A_i, B_j)$ which satisfies the condition

$$0 \leq P(A_i, B_j) \leq 1$$

Assuming that the outcomes B_j, $j = 1, 2, \ldots, m$, are mutually exclusive, it follows that

$$\sum_{j=1}^{m} P(A_i, B_j) = P(A_i) \tag{2-1-8}$$

Similarly, if the outcomes A_i, $i = 1, 2, \ldots, n$, are mutually exclusive then

$$\sum_{i=1}^{n} P(A_i, B_j) = P(B_j) \tag{2-1-9}$$

Furthermore, if all the outcomes of the two experiments are mutually exclusive then

$$\sum_{i=1}^{n}\sum_{j=1}^{m} P(A_i, B_j) = 1 \tag{2-1-10}$$

The generalization of the above treatment to more than two experiments is straightforward.

Conditional Probabilities Consider a combined experiment in which a joint event occurs with probability $P(A, B)$. Suppose that the event B has occurred and we wish to determine the probability of occurrence of the event A. This is called the *conditional probability* of the event A given the occurrence of the event B and is defined as

$$P(A \mid B) = \frac{P(A, B)}{P(B)} \tag{2-1-11}$$

provided $P(B) > 0$. In a similar manner, the probability of the event B conditioned on the occurrence of the event A is defined as

$$P(B \mid A) = \frac{P(A, B)}{P(A)} \tag{2-1-12}$$

provided $P(A) > 0$. The relations in (2-1-11) and (2-1-12) may also be expressed as

$$P(A, B) = P(A \mid B)P(B) = P(B \mid A)P(A) \tag{2-1-13}$$

The relations in (2-1-11), (2-1-12), and (2-1-13) also apply to a single experiment in which A and B are any two events defined on the sample space S and $P(A, B)$ is interpreted as the probability of the $A \cap B$. That is, $P(A, B)$ denotes the simultaneous occurrence of A and B. For example, consider the events B and C given by (2-1-4) and (2-1-5), respectively, for the single toss of a die. The joint event consists of the sample points $\{1, 3\}$. The conditional probability of the event C given that B occurred is

$$P(C \mid B) = \frac{\frac{2}{6}}{\frac{3}{6}} = \frac{2}{3}$$

In a single experiment, we observe that when two events A and B are mutually exclusive, $A \cap B = \varnothing$ and, hence, $P(A \mid B) = 0$. Also, if A is a subset of B then $A \cap B = A$ and, hence,

$$P(A \mid B) = \frac{P(A)}{P(B)}$$

On the other hand, if B is a subset of A, we have $A \cap B = B$ and, hence,

$$P(A \mid B) = \frac{P(B)}{P(B)} = 1$$

An extremely useful relationship for conditional probabilities is Bayes' theorem, which states that if A_i, $i = 1, 2, \ldots, n$, are mutually exclusive events such that

$$\bigcup_{i=1}^{n} A_i = S$$

and B is an arbitrary event with nonzero probability then

$$\begin{aligned} P(A_i \mid B) &= \frac{P(A_i, B)}{P(B)} \\ &= \frac{P(B \mid A_i)P(A_i)}{\sum_{j=1}^{n} P(B \mid A_j)P(A_j)} \end{aligned} \tag{2-1-14}$$

We use this formula in Chapter 5 to derive the structure of the optimum receiver for a digital communication system in which the events A_i, $i = 1, 2, \ldots, n$, represent the possible transmitted messages in a given time interval, $P(A_i)$ represent their *a priori* probabilities, B represents the received signal, which consists of the transmitted message (one of the A_i) corrupted by noise, and $P(A_i \mid B)$ is the *a posteriori* probability of A_i conditioned on having observed the received signal B.

Statistical Independence The statistical independence of two or more events is another important concept in probability theory. It usually arises when we consider two or more experiments or repeated trials of a single experiment. To explain this concept, we consider two events A and B and their conditional probability $P(A \mid B)$, which is the probability of occurrence of A given that B has occurred. Suppose that the occurrence of A does not depend on the occurrence of B. That is,

$$P(A \mid B) = P(A) \tag{2-1-15}$$

Substitution of (2-1-15) into (2-1-13) yields the result

$$P(A, B) = P(A)P(B) \tag{2-1-16}$$

That is, the joint probability of the events A and B factors into the product of

the elementary or marginal probabilities $P(A)$ and $P(B)$. When the events A and B satisfy the relation in (2-1-16), they are said to be *statistically independent.*

For example, consider two successive experiments in tossing a die. Let A represent the even-numbered sample points $\{2, 4, 6\}$ in the first toss and B represent the even-numbered possible outcomes $\{2, 4, 6\}$ in the second toss. In a fair die, we assign the probabilities $P(A) = \frac{1}{2}$ and $P(B) = \frac{1}{2}$. Now, the joint probability of the joint event "even-numbered outcome on the first toss and even-numbered outcome on the second toss" is just the probability of the nine pairs of outcomes (i, j), $i = 2, 4, 6$, $j = 2, 4, 6$, which is $\frac{1}{4}$. Also,

$$P(A, B) = P(A)P(B) = \frac{1}{4}$$

Thus, the events A and B are statistically independent. Similarly, we may say that the outcomes of the two experiments are statistically independent.

The definition of statistical independence can be extended to three or more events. Three statistically independent events A_1, A_2, and A_3 must satisfy the following conditions:

$$\begin{aligned} P(A_1, A_2) &= P(A_1)P(A_2) \\ P(A_1, A_3) &= P(A_1)P(A_3) \\ P(A_2, A_3) &= P(A_2)P(A_3) \\ P(A_1, A_2, A_3) &= P(A_1)P(A_2)P(A_3) \end{aligned} \tag{2-1-17}$$

In the general case, the events A_i, $i = 1, 2, \ldots, n$, are statistically independent provided that the probabilities of the joint events taken $2, 3, 4, \ldots$, and n at a time factor into the product of the probabilities of the individual events.

2-1-1 Random Variables, Probability Distributions, and Probability Densities

Given an experiment having a sample space S and elements $s \in S$, we define a function $X(s)$ whose domain is S and whose range is a set of numbers on the real line. The function $X(s)$ is called a *random variable.* For example, if we flip a coin the possible outcomes are head (H) and tail (T), so S contains two points labeled H and T. Suppose we define a function $X(s)$ such that

$$X(s) = \begin{cases} 1 & (s = H) \\ -1 & (s = T) \end{cases} \tag{2-1-18}$$

Thus we have mapped the two possible outcomes of the coin-flipping

experiment into the two points (± 1) on the real line. Another experiment is the toss of a die with possible outcomes $S = \{1, 2, 3, 4, 5, 6\}$. A random variable defined on this sample space may be $X(s) = s$, in which case the outcomes of the experiment are mapped into the integers $1, \ldots, 6$, or, perhaps, $X(s) = s^2$, in which case the possible outcomes are mapped into the integers $\{1, 4, 9, 16, 25, 36\}$. These are examples of discrete random variables.

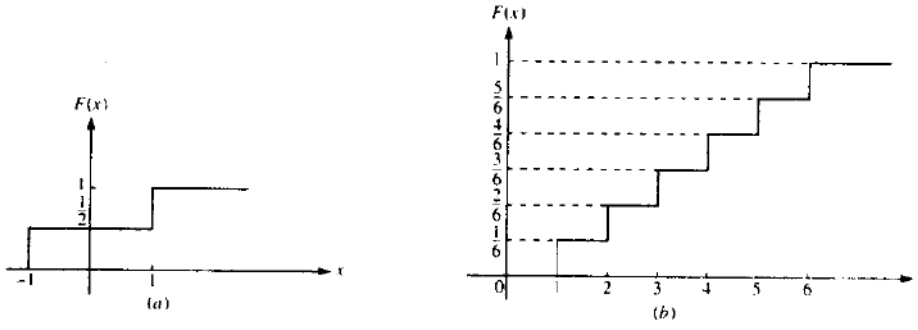
Although we have used as examples experiments that have a finite set of possible outcomes, there are many physical systems (experiments) that generate continuous outputs (outcomes). For example, the noise voltage generated by an electronic amplifier has a continuous amplitude. Consequently, the sample space S of voltage amplitudes $v \in S$ is continuous and so is the mapping $X(v) = v$. In such a case, the random variable† X is said to be a *continuous random variable.*

Given a random variable X, let us consider the event $\{X \leq x\}$ where x is any real number in the interval $(-\infty, \infty)$. We write the probability of this event as $P(X \leq x)$ and denote it simply by $F(x)$, i.e.,

$$F(x) = P(X \leq x) \quad (-\infty < x < \infty) \tag{2-1-19}$$

The function $F(x)$ is called the *probability distribution function* of the random variable X. It is also called the *cumulative distribution function* (cdf). Since $F(x)$ is a probability, its range is limited to the interval $0 \leq F(x) \leq 1$. In fact, $F(-\infty) = 0$ and $F(\infty) = 1$. For example, the discrete random variable generated by flipping a fair coin and defined by (2-1-18) has the cdf shown in Fig. 2-1-1(a). There are two discontinuities or jumps in $F(x)$, one at $x = -1$ and one at $x = 1$. Similarly, the random variable $X(s) = s$ generated by tossing a fair die has the cdf shown in Fig. 2-1-1(b). In this case $F(x)$ has six jumps, one at each of the points $x = 1, \ldots, 6$.

FIGURE 2-1-1 Examples of the cumulative distribution functions of two discrete random variables.



† The random variable $X(s)$ will be written simply as X.

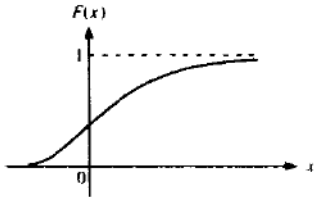


FIGURE 2-1-2 An example of the cumulative distribution function of a continuous random variable.

The cdf of a continuous random variable typically appears as shown in Fig. 2-1-2. This is a smooth, nondecreasing function of x. In some practical problems, we may also encounter a random variable of a mixed type. The cdf of such a random variable is a smooth, nondecreasing function in certain parts of the real line and contains jumps at a number of discrete values of x. An example of such a cdf is illustrated in Fig. 2-1-3.

The derivative of the cdf $F(x)$, denoted as $p(x)$, is called the *probability density function* (pdf) of the random variable X. Thus, we have

$$p(x) = \frac{dF(x)}{dx} \quad (-\infty < x \leq \infty) \tag{2-1-20}$$

or, equivalently

$$F(x) = \int_{-\infty}^{x} p(u)\, du \quad (-\infty < x < \infty) \tag{2-1-21}$$

Since $F(x)$ is a nondecreasing function, it follows that $p(x) \geq 0$. When the random variable is discrete or of a mixed type, the pdf contains impulses at the points of discontinuity of $F(x)$. In such cases, the discrete part of $p(x)$ may be expressed as

$$p(x) = \sum_{i=1}^{n} P(X = x_i)\, \delta(x - x_i) \tag{2-1-22}$$

where x_i, $i = 1, 2, \ldots, n$, are the possible discrete values of the random

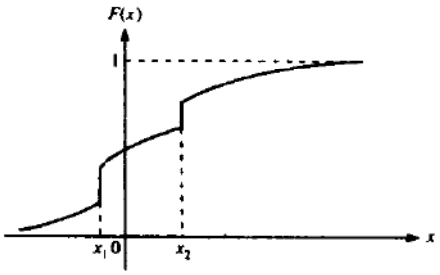


FIGURE 2-1-3 An example of the cumulative distribution function of a random variable of a mixed type.

variable; $P(X = x_i)$, $i = 1, 2, \ldots, n$, are the probabilities, and $\delta(x)$ denotes an impulse at $x = 0$.

Often we are faced with the problem of determining the probability that a random variable X falls in an interval (x_1, x_2), where $x_2 > x_1$. To determine the probability of this event, let us begin with the event $\{X \leq x_2\}$. The event can always be expressed as the union of two mutually exclusive events $\{X \leq x_1\}$ and $\{x_1 < X \leq x_2\}$. Hence the probability of the event $\{X \leq x_2\}$ can be expressed as the sum of the probabilities of the mutually exclusive events. Thus we have

$$P(X \leq x_2) = P(X \leq x_1) + P(x_1 < X \leq x_2)$$

$$F(x_2) = F(x_1) + P(x_1 < X \leq x_2)$$

or, equivalently,

$$P(x_1 < X \leq x_2) = F(x_2) - F(x_1)$$

$$= \int_{x_1}^{x_2} p(x)\, dx \tag{2-1-23}$$

In other words, the probability of the event $\{x_1 < X \leq x_2\}$ is simply the area under the pdf in the range $x_1 < X \leq x_2$.

Multiple Random Variables, Joint Probability Distributions, and Joint Probability Densities In dealing with combined experiments or repeated trials of a single experiment, we encounter multiple random variables and their cdfs and pdfs. Multiple random variables are basically multidimensional functions defined on a sample space of a combined experiment. Let us begin with two random variables X_1 and X_2, each of which may be continuous, discrete, or mixed. The joint cumulative distribution function (joint cdf) for the two random variables is defined as

$$F(x_1, x_2) = P(X_1 \leq x_1, X_2 \leq x_2)$$

$$= \int_{-\infty}^{x_1} \int_{-\infty}^{x_2} p(u_1, u_2)\, du_1\, du_2 \tag{2-1-24}$$

where $p(x_1, x_2)$ is the joint probability density function (joint pdf). The latter may also be expressed in the form

$$p(x_1, x_2) = \frac{\partial^2}{\partial x_1\, \partial x_2} F(x_1, x_2) \tag{2-1-25}$$

When the joint pdf $p(x_1, x_2)$ is integrated over one of the variables, we obtain the pdf of the other variable. That is,

$$\int_{-\infty}^{\infty} p(x_1, x_2)\, dx_1 = p(x_2)$$

$$\int_{-\infty}^{\infty} p(x_1, x_2)\, dx_2 = p(x_1) \tag{2-1-26}$$

The pdfs $p(x_1)$ and $p(x_2)$ obtained from integrating over one of the variables are called *marginal pdf*s. Furthermore, if $p(x_1, x_2)$ is integrated over both variables, we obtain

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} p(x_1, x_2)\, dx_1\, dx_2 = F(\infty, \infty) = 1 \tag{2-1-27}$$

We also note that $F(-\infty, -\infty) = F(-\infty, x_2) = F(x_1, -\infty) = 0$.

The generalization of the above expressions to multidimensional random variables is straightforward. Suppose that X_i, $i = 1, 2, \ldots, n$, are random variables with a joint cdf defined as

$$\begin{aligned} F(x_1, x_2, \ldots, x_n) &= P(X_1 \leq x_1, X_2 \leq x_2, \ldots, X_n \leq x_n) \\ &= \int_{-\infty}^{x_1}\int_{-\infty}^{x_2}\cdots\int_{-\infty}^{x_n} p(u_1, u_2, \ldots, u_n)\, du_1\, du_2 \cdots du_n \end{aligned} \tag{2-1-28}$$

where $p(x_1, x_2, \ldots, x_n)$ is the joint pdf. By taking the partial derivatives of $F(x_1, x_2, \ldots, x_n)$ given by (2-1-28), we obtain

$$p(x_1, x_2, \ldots, x_n) = \frac{\partial^n}{\partial x_1\, \partial x_2 \cdots \partial x_n} F(x_1, x_2, \ldots, x_n) \tag{2-1-29}$$

Any number of variables in $p(x_1, x_2, \ldots, x_n)$ can be eliminated by integrating over these variables. For example, integration over x_2 and x_3 yields

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} p(x_1, x_2, x_3, \ldots, x_n)\, dx_2\, dx_3 = p(x_1, x_4, \ldots, x_n) \tag{2-1-30}$$

It also follows that $F(x_1, \infty, \infty, x_4, \ldots, x_n) = F(x_1, x_4, x_5, \ldots, x_n)$ and

$$F(x_1, -\infty, -\infty, x_4, \ldots, x_n) = 0.$$

Conditional Probability Distribution Functions Let us consider two random variables X_1 and X_2 with joint pdf $p(x_1, x_2)$. Suppose that we wish to determine the probability that the random variable $X_1 \leq x_1$ conditioned on

$$x_2 - \Delta x_2 < X_2 \leq x_2$$

where Δx_2 is some positive increment. That is, we wish to determine the probability of the event $(X_1 \leq x_1 \,|\, x_2 - \Delta x_2 < X_2 \leq x_2)$. Using the relations established earlier for the conditional probability of an event, the probability of the event $(X_1 \leq x_1 \,|\, x_2 - \Delta x_2 < X_2 \leq x_2)$ can be expressed as the probability

of the joint event $(X_1 \leq x_1, x_2 - \Delta x_2 < X_2 \leq x_2)$ divided by the probability of the event $(x_2 - \Delta x_2 < X_2 \leq x_2)$. Thus

$$P(X_1 \leq x_1 \mid x_2 - \Delta x_2 < X_2 \leq x_2) = \frac{\int_{-\infty}^{x_1} \int_{x_2-\Delta x_2}^{x_2} p(u_1, u_2)\, du_1\, du_2}{\int_{x_2-\Delta x_2}^{x_2} p(u_2)\, du_2}$$

$$= \frac{F(x_1, x_2) - F(x_1, x_2 - \Delta x_2)}{F(x_2) - F(x_2 - \Delta x_2)} \tag{2-1-31}$$

Assuming that the pdfs $p(x_1, x_2)$ and $p(x_2)$ are continuous functions over the interval $(x_2 - \Delta x_2, x_2)$, we may divide both numerator and denominator in (2-1-31) by Δx_2 and take the limit as $\Delta x_2 \to 0$. Thus we obtain

$$P(X_1 \leq x_1 \mid X_2 = x_2) \equiv F(x_1 \mid x_2) = \frac{\partial F(x_1, x_2)/\partial x_2}{\partial F(x_2)/\partial x_2}$$

$$= \frac{\partial[\int_{-\infty}^{x_1} \int_{-\infty}^{x_2} p(u_1, u_2)\, du_1 du_2]/\partial x_2}{\partial[\int_{-\infty}^{x_2} p(u_2)\, du_2]/\partial x_2}$$

$$= \frac{\int_{-\infty}^{x_1} p(u_1, x_2)\, du_1}{p(x_2)} \tag{2-1-32}$$

which is the conditional cdf of the random variable X_1 given the random variable X_2. We observe that $F(-\infty \mid x_2) = 0$ and $F(\infty \mid x_2) = 1$. By differentiating (2-1-32) with respect to x_1, we obtain the corresponding pdf $p(x_1 \mid x_2)$ in the form

$$p(x_1 \mid x_2) = \frac{p(x_1, x_2)}{p(x_2)} \tag{2-1-33}$$

Alternatively, we may express the joint pdf $p(x_1, x_2)$ in terms of the conditional pdfs, $p(x_1 \mid x_2)$ or $p(x_2 \mid x_1)$, as

$$p(x_1, x_2) = p(x_1 \mid x_2)p(x_2)$$

$$= p(x_2 \mid x_1)p(x_1) \tag{2-1-34}$$

The extension of the relations given above to multidimensional random variables is also easily accomplished. Beginning with the joint pdf of the random variables X_i, $i = 1, 2, \ldots, n$, we may write

$$p(x_1, x_2, \ldots, x_n) = p(x_1, x_2, \ldots, x_k \mid x_{k+1}, \ldots, x_n)p(x_{k+1}, \ldots, x_n) \tag{2-1-35}$$

where k is any integer in the range $1 < k < n$. The joint conditional cdf corresponding to the pdf $p(x_1, x_2, \ldots, x_k \mid x_{k+1}, \ldots, x_n)$ is

$$F(x_1, x_2, \ldots, x_k \mid x_{k+1}, \ldots, x_n)$$

$$= \frac{\int_{-\infty}^{x_1} \cdots \int_{-\infty}^{x_k} p(u_1, u_2, \ldots, u_k, x_{k+1}, \ldots, x_n)\, du_1\, du_2 \cdots du_k}{p(x_{k+1}, \ldots, x_n)} \tag{2-1-36}$$

This conditional cdf satisfies the properties previously established for these functions, such as

$$F(\infty, x_2, \ldots, x_k \mid x_{k+1}, \ldots, x_n) = F(x_2, x_3, \ldots, x_k \mid x_{k+1}, \ldots, x_n)$$
$$F(-\infty, x_2, \ldots, x_k \mid x_{k+1}, \ldots, x_n) = 0$$

Statistically Independent Random Variables. We have already defined statistical independence of two or more events of a sample space S. The concept of statistical independence can be extended to random variables defined on a sample space generated by a combined experiment or by repeated trials of a single experiment. If the experiments result in mutually exclusive outcomes, the probability of an outcome in one experiment is independent of an outcome in any other experiment. That is, the joint probability of the outcomes factors into a product of the probabilities corresponding to each outcome. Consequently, the random variables corresponding to the outcomes in these experiments are independent in the sense that their joint pdf factors into a product of marginal pdfs. Hence the multidimensional random variables are statistically independent if and only if

$$F(x_1, x_2, \ldots, x_n) = F(x_1)F(x_2) \cdots F(x_n) \tag{2-1-37}$$

or, alternatively,

$$p(x_1, x_2, \ldots, x_n) = p(x_1)p(x_2) \cdots p(x_n) \tag{2-1-38}$$

2-1-2 Functions of Random Variables

A problem that arises frequently in practical applications of probability is the following. Given a random variable X, which is characterized by its pdf $p(x)$, determine the pdf of the random variable $Y = g(X)$, where $g(X)$ is some given function of X. When the mapping g from X to Y is one-to-one, the determination of $p(y)$ is relatively straightforward. However, when the mapping is not one-to-one, as is the case, for example, when $Y = X^2$, we must be very careful in our derivation of $p(y)$.

Example 2-1-1

Consider the random variable Y defined as

$$Y = aX + b \tag{2-1-39}$$

where a and b are constants. We assume that $a > 0$. If $a < 0$, the approach is similar (see Problem 2-3). We note that this mapping, illustrated in Fig. 2-1-4(a) is linear and monotonic. Let $F_X(x)$ and $F_Y(y)$ denote the cdfs for X and Y, respectively.† Then

$$F_Y(y) = P(Y \leq y) = P(aX + b \leq y) = P\left(X \leq \frac{y-b}{a}\right)$$
$$= \int_{-\infty}^{(y-b)/a} p_X(x)\,dx = F_X\left(\frac{y-b}{a}\right) \tag{2-1-40}$$

† To avoid confusion in changing variables, subscripts are used in the respective pdfs and cdfs.

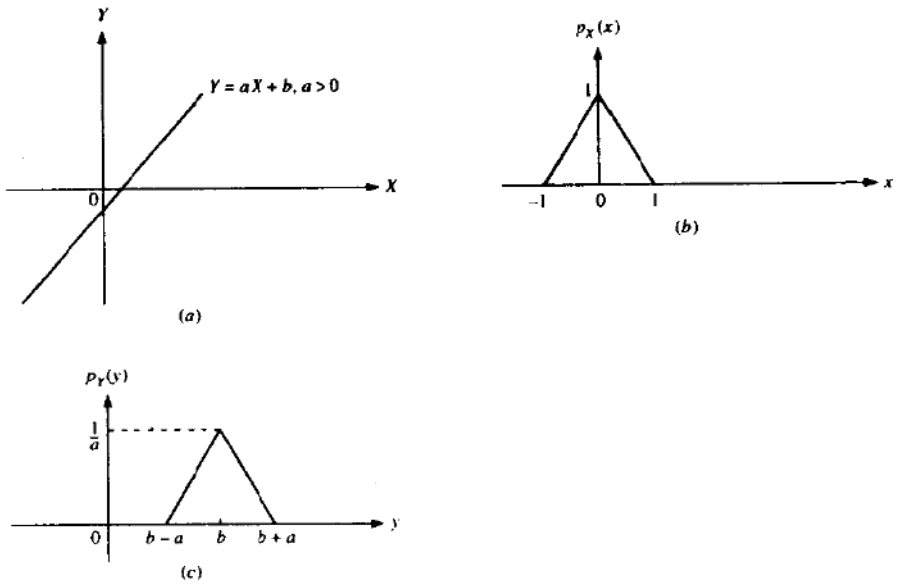


FIGURE 2-1-4 A linear transformation of a random variable X and an example of the corresponding pdfs of X and Y.

By differentiating (2-1-40) with respect to y, we obtain the relationship between the respective pdfs. It is

$$p_Y(y) = \frac{1}{a} p_X\left(\frac{y-b}{a}\right) \tag{2-1-41}$$

Thus (2-1-40) and (2-1-41) specify the cdf and pdf of the random variable Y in terms of the cdf and pdf of the random variable X for the linear transformation in (2-1-39). To illustrate this mapping for a specific pdf $p_X(x)$, consider the one shown in Fig. 2-1-4(b). The pdf $p_Y(y)$ that results from the mapping in (2-1-39) is shown in Fig. 2-1-4(c).

Example 2-1-2

Consider the random variable Y defined as

$$Y = aX^3 + b, \quad a > 0 \tag{2-1-42}$$

As in Example 2-1-1, the mapping between X and Y is one-to-one. Hence

$$\begin{aligned} F_Y(y) &= P(Y \leq y) = P(aX^3 + b \leq y) \\ &= P\left[X \leq \left(\frac{y-b}{a}\right)^{1/3}\right] = F_X\left[\left(\frac{y-b}{a}\right)^{1/3}\right] \end{aligned} \tag{2-1-43}$$

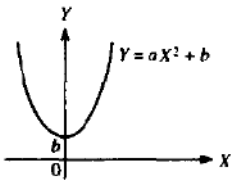


FIGURE 2-1-5 A quadratic transformation of the random variable X.

Differentiation of (2-1-43) with respect to y yields the desired relationship between the two pdfs as

$$p_Y(y) = \frac{1}{3a[(y-b)/a]^{2/3}} p_X\left[\left(\frac{y-b}{a}\right)^{1/3}\right] \tag{2-1-44}$$

Example 2-1-3

The random variable Y is defined as

$$Y = aX^2 + b, \quad a > 0 \tag{2-1-45}$$

In contrast to Examples 2-1-1 and 2-1-2, the mapping between X and Y, illustrated in Fig. 2-1-5, is not one-to-one. To determine the cdf of Y, we observe that

$$\begin{aligned} F_Y(y) &= P(Y \leq y) = P(aX^2 + b \leq y) \\ &= P\left(|X| \leq \sqrt{\frac{y-b}{a}}\right) \end{aligned}$$

Hence

$$F_Y(y) = F_X\left(\sqrt{\frac{y-b}{a}}\right) - F_X\left(-\sqrt{\frac{y-b}{a}}\right) \tag{2-1-46}$$

Differentiating (2-1-46) with respect to y, we obtain the pdf of Y in terms of the pdf of X in the form

$$p_Y(y) = \frac{p_X[\sqrt{(y-b)/a}]}{2a\sqrt{[(y-b)/a]}} + \frac{p_X[-\sqrt{(y-b)/a}]}{2a\sqrt{[(y-b)/a]}} \tag{2-1-47}$$

In Example 2-1-3, we observe that the equation $g(x) = ax^2 + b = y$ has two real solutions,

$$x_1 = \sqrt{\frac{y-b}{a}}$$

$$x_2 = -\sqrt{\frac{y-b}{a}}$$

and that $p_Y(y)$ consists of two terms corresponding to these two solutions. That is,

$$p_Y(y) = \frac{p_X[x_1 = \sqrt{(y-b)/a}]}{|g'[x_1 = \sqrt{(y-b)/a}]|} + \frac{p_X[x_2 = -\sqrt{(y-b)/a}]}{|g'[x_2 = -\sqrt{(y-b)/a}]|} \tag{2-1-48}$$

where $g'(x)$ denotes the first derivative of $g(x)$.

In the general case, suppose that $x_1, x_2, \ldots, x_n$ are the real roots of the equation $g(x) = y$. Then the pdf of the random variable $Y = g(X)$ may be expressed as

$$p_Y(y) = \sum_{i=1}^{n} \frac{p_X(x_i)}{|g'(x_i)|} \tag{2-1-49}$$

where the roots x_i, $i = 1, 2, \ldots, n$, are functions of y.

Now let us consider functions of multidimensional random variables. Suppose that X_i, $i = 1, 2, \ldots, n$, are random variables with joint pdf $p_X(x_1, x_2, \ldots, x_n)$, and let Y_i, $i = 1, 2, \ldots, n$, be another set of n random variables related to the X_i by the functions

$$Y_i = g_i(X_1, X_2, \ldots, X_n), \quad i = 1, 2, \ldots, n \tag{2-1-50}$$

We assume that the $g_i(X_1, X_2, \ldots, X_n)$, $i = 1, 2, \ldots, n$, are single-valued functions with continuous partial derivatives and invertible. By "invertible" we mean that the X_i, $i = 1, 2, \ldots, n$, can be expressed as functions of Y_i, $i = 1, 2, \ldots, n$, in the form

$$X_i = g_i^{-1}(Y_1, Y_2, \ldots, Y_n), \quad i = 1, 2, \ldots, n \tag{2-1-51}$$

where the inverse functions are also assumed to be single-valued with continuous partial derivatives. The problem is to determine the joint pdf of Y_i, $i = 1, 2, \ldots, n$, denoted by $p_Y(y_1, y_2, \ldots, y_n)$, given the joint pdf $p_X(x_1, x_2, \ldots, x_n)$.

To determine the desired relation, let R_X be the region in the n-dimensional space of the random variables X_i, $i = 1, 2, \ldots, n$, and let R_Y be the (one-to-one) mapping of R_X defined by the functions $Y_i = g_i(X_1, X_2, \ldots, X_n)$. Clearly,

$$\idotsint_{R_Y} p_Y(y_1, y_2, \ldots, y_n)\, dy_1\, dy_2 \cdots dy_n = \idotsint_{R_X} p_X(x_1, x_2, \ldots, x_n)\, dx_1\, dx_2 \cdots dx_n \tag{2-1-52}$$

By making a change in variables in the multiple integral on the right-hand side of (2-1-52) with the substitution

$$x_i = g_i^{-1}(y_1, y_2, \ldots, y_n) \equiv g_i^{-1}, \quad i = 1, 2, \ldots, n$$

we obtain

$$\int\int_{R_Y} \cdots \int p_Y(y_1, y_2, \ldots, y_n)\, dy_1\, dy_2 \cdots dy_n$$

$$= \int\int_{R_X} \cdots \int p_X(x_1 = g_1^{-1}, x_2 = g_2^{-1}, \ldots, x_n = g_n^{-1})\, |J|\, dy_1\, dy_2 \cdots dy_n \tag{2-1-53}$$

where J denotes the jacobian of the transformation, defined by the determinant

$$J = \begin{vmatrix} \dfrac{\partial g_1^{-1}}{\partial y_1} & \dfrac{\partial g_2^{-1}}{\partial y_1} & \cdots & \dfrac{\partial g_n^{-1}}{\partial y_1} \\ \vdots & \vdots & & \vdots \\ \dfrac{\partial g_1^{-1}}{\partial y_n} & \dfrac{\partial g_2^{-1}}{\partial y_n} & \cdots & \dfrac{\partial g_n^{-1}}{\partial y_n} \end{vmatrix} \tag{2-1-54}$$

Consequently, the desired relation for the joint pdf of the Y_i, $i = 1, 2, \ldots, n$, is

$$p_Y(y_1, y_2, \ldots, y_n) = p_X(x_1 = g_1^{-1}, x_2 = g_2^{-1}, \ldots, x_n = g_n^{-1})\, |J| \tag{2-1-55}$$

Example 2-1-4

An important functional relation between two sets of n-dimensional random variables that frequently arises in practice is the linear transformation

$$Y_i = \sum_{j=1}^{n} a_{ij} X_j, \qquad i = 1, 2, \ldots, n \tag{2-1-56}$$

where the $\{a_{ij}\}$ are constants. It is convenient to employ the matrix form for the transformation, which is

$$\mathbf{Y} = \mathbf{AX} \tag{2-1-57}$$

where $\mathbf{X}$ and $\mathbf{Y}$ are n-dimensional vectors and $\mathbf{A}$ is an $n \times n$ matrix. We assume that $\mathbf{A}$ is nonsingular. Then $\mathbf{A}$ is invertible and, hence,

$$\mathbf{X} = \mathbf{A}^{-1}\mathbf{Y} \tag{2-1-58}$$

Equivalently, we have

$$X_i = \sum_{j=1}^{n} b_{ij} Y_j, \qquad i = 1, 2, \ldots, n \tag{2-1-59}$$

where $\{b_{ij}\}$ are the elements of the inverse matrix $\mathbf{A}^{-1}$. The jacobian of this transformation is $J = 1/\det \mathbf{A}$. Hence

$$p_Y(y_1, y_2, \ldots, y_n)$$

$$= p_X\left(x_1 = \sum_{j=1}^{n} b_{1j} y_j, x_2 = \sum_{j=1}^{n} b_{2j} y_j, \ldots, x_n = \sum_{j=1}^{n} b_{nj} y_j\right) \frac{1}{|\det \mathbf{A}|} \tag{2-1-60}$$

2-1-3 Statistical Averages of Random Variables

Averages play an important role in the characterization of the outcomes of experiments and the random variables defined on the sample space of the experiments. Of particular interest are the first and second moments of a single random variable and the joint moments, such as the correlation and covariance, between any pair of random variables in a multidimensional set of random variables. Also of great importance are the characteristic function for a single random variable and the joint characteristic function for a multidimensional set of random variables. This section is devoted to the definition of these important statistical averages.

First we consider a single random variable X characterized by its pdf $p(x)$. The *mean* or *expected value* of X is defined as

$$E(X) \equiv m_x = \int_{-\infty}^{\infty} xp(x)\,dx \tag{2-1-61}$$

where $E(\ )$ denotes expectation (statistical averaging). This is the first moment of the random variable X. In general, the nth moment is defined as

$$E(X^n) = \int_{-\infty}^{\infty} x^n p(x)\,dx \tag{2-1-62}$$

Now, suppose that we define a random variable $Y = g(X)$, where $g(X)$ is some arbitrary function of the random variable X. The expected value of Y is

$$E(Y) = E[g(X)] = \int_{-\infty}^{\infty} g(x)p(x)\,dx \tag{2-1-63}$$

In particular, if $Y = (X - m_x)^n$ where m_x is the mean value of X, then

$$E(Y) = E[(X - m_x)^n] = \int_{-\infty}^{\infty} (x - m_x)^n p(x)\,dx \tag{2-1-64}$$

This expected value is called the nth *central moment* of the random variable X, because it is a moment taken relative to the mean. When $n = 2$, the central moment is called the *variance* of the random variable and denoted as σ_x^2. That is,

$$\sigma_x^2 = \int_{-\infty}^{\infty} (x - m_x)^2 p(x)\,dx \tag{2-1-65}$$

This parameter provides a measure of the dispersion of the random variable X. By expanding the term $(x - m_x)^2$ in the integral of (2-1-65) and noting that the expected value of a constant is equal to the constant, we obtain the expression that relates the variance to the first and second moments, namely,

$$\begin{aligned}\sigma_x^2 &= E(X^2) - [E(X)]^2 \\ &= E(X^2) - m_x^2\end{aligned} \tag{2-1-66}$$

In the case of two random variables, X_1 and X_2, with joint pdf $p(x_1, x_2)$, we define the *joint moment* as

$$E(X_1^k X_2^n) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x_1^k x_2^n p(x_1, x_2)\, dx_1\, dx_2 \tag{2-1-67}$$

and the *joint central moment* as

$$E[(X_1 - m_1)^k (X_2 - m_2)^n] = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x_1 - m_1)^k (x_2 - m_2)^n p(x_1, x_2)\, dx_1\, dx_2 \tag{2-1-68}$$

where $m_i = E(X_i)$. Of particular importance to us are the joint moment and joint central moment corresponding to $k = n = 1$. These joint moments are called the *correlation* and the *covariance* of the random variables X_1 and X_2, respectively.

In considering multidimensional random variables, we can define joint moments of any order. However, the moments that are most useful in practical applications are the correlations and covariances between pairs of random variables. To elaborate, suppose that X_i, $i = 1, 2, \ldots, n$, are random variables with joint pdf $p(x_1, x_2, \ldots, x_n)$. Let $p(x_i, x_j)$ be the joint pdf of the random variables X_i and X_j. Then the correlation between X_i and X_j is given by the joint moment

$$E(X_i X_j) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x_i x_j p(x_i, x_j)\, dx_i\, dx_j \tag{2-1-69}$$

and the covariance of X_i and X_j is

$$\begin{aligned} \mu_{ij} &\equiv E[(X_i - m_i)(X_j - m_j)] \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x_i - m_i)(x_j - m_j) p(x_i, x_j)\, dx_i\, dx_j \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x_i x_j p(x_i, x_j)\, dx_i\, dx_j - m_i m_j \\ &= E(X_i X_j) - m_i m_j \end{aligned} \tag{2-1-70}$$

The $n \times n$ matrix with elements μ_{ij} is called the *covariance matrix* of the random variables X_i, $i = 1, 2, \ldots, n$. We shall encounter the covariance matrix in our discussion of jointly gaussian random variables in Section 2-1-4.

Two random variables are said to be *uncorrelated* if $E(X_i X_j) = E(X_i)E(X_j) = m_i m_j$. In that case, the covariance $\mu_{ij} = 0$. We note that when X_i and X_j are statistically independent, they are also uncorrelated. However, if X_i and X_j are uncorrelated, they are not necessarily statistically independent.

Two random variables are said to be *orthogonal* if $E(X_iX_j) = 0$. We note that this condition holds when X_i and X_j are uncorrelated and either one or both of the random variables have zero mean.

Characteristic Functions The *characteristic function* of a random variable X is defined as the statistical average

$$E(e^{jvX}) \equiv \psi(jv) = \int_{-\infty}^{\infty} e^{jvx} p(x)\, dx \tag{2-1-71}$$

where the variable v is real and $j = \sqrt{-1}$. We note that $\psi(jv)$ may be described as the Fourier transform† of the pdf $p(x)$. Hence the inverse Fourier transform is

$$p(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \psi(jv) e^{-jvx}\, dv \tag{2-1-72}$$

One useful property of the characteristic function is its relation to the moments of the random variable. We note that the first derivative of (2-1-71) with respect to v yields

$$\frac{d\psi(jv)}{dv} = j \int_{-\infty}^{\infty} x e^{jvx} p(x)\, dx$$

By evaluating the derivative at $v = 0$, we obtain the first moment (mean)

$$E(X) = m_x = -j \frac{d\psi(jv)}{dv}\bigg|_{v=0} \tag{2-1-73}$$

The differentiation process can be repeated, so that the nth derivative of $\psi(jv)$ evaluated at $v = 0$ yields the nth moment

$$E(X^n) = (-j)^n \frac{d^n \psi(jv)}{dv^n}\bigg|_{v=0} \tag{2-1-74}$$

Thus the moments of a random variable can be determined from the characteristic function. On the other hand, suppose that the characteristic function can be expanded in a Taylor series about the point $v = 0$. That is,

$$\psi(jv) = \sum_{n=0}^{\infty} \left[\frac{d^n \psi(jv)}{dv^n}\right]_{v=0} \frac{v^n}{n!} \tag{2-1-75}$$

Using the relation in (2-1-74) to eliminate the derivative in (2-1-75), we obtain

† Usually the Fourier transform of a function $g(u)$ is defined as $G(v) = \int_{-\infty}^{\infty} g(u) e^{-juv}\, du$, which differs from (2-1-71) by the negative sign in the exponential. This is a trivial difference, however, so we call the integral in (2-1-71) a Fourier transform.

an expression for the characteristic function in terms of its moments in the form

$$\psi(jv) = \sum_{n=0}^{\infty} E(X^n)\frac{(jv)^n}{n!} \tag{2-1-76}$$

The characteristic function provides a simple method for determining the pdf of a sum of statistically independent random variables. To illustrate this point, let X_i, $i = 1, 2, \ldots, n$, be a set of n statistically independent random variables and let

$$Y = \sum_{i=1}^{n} X_i \tag{2-1-77}$$

The problem is to determine the pdf of Y. We shall determine the pdf of Y by first finding its characteristic function and then computing the inverse Fourier transform. Thus

$$\begin{aligned}\psi_Y(jv) &= E(e^{jvY}) \\ &= E\left[\exp\left(jv\sum_{i=1}^{n} X_i\right)\right] \\ &= E\left[\prod_{i=1}^{n}(e^{jvX_i})\right] \\ &= \int_{-\infty}^{\infty}\cdots\int_{-\infty}^{\infty}\left(\prod_{i=1}^{n} e^{jvx_i}\right)p(x_1, x_2, \ldots, x_n)\,dx_1\,dx_2\cdots dx_n\end{aligned} \tag{2-1-78}$$

Since the random variables are statistically independent, $p(x_1, x_2, \ldots, x_n) = p(x_1)p(x_2)\cdots p(x_n)$, and, hence, the nth-order integral in (2-1-78) reduces to a product of n single integrals, each corresponding to the characteristic function of one of the X_i. Hence,

$$\psi_Y(jv) = \prod_{i=1}^{n}\psi_{X_i}(jv) \tag{2-1-79}$$

If, in addition to their statistical independence, the X_i are identically distributed then all the $\psi_X(jv)$ are identical. Consequently,

$$\psi_Y(jv) = [\psi_X(jv)]^n \tag{2-1-80}$$

Finally, the pdf of Y is determined from the inverse Fourier transform of $\psi_Y(jv)$, given by (2-1-72).

Since the characteristic function of the sum of n statistically independent random variables is equal to the product of the characteristic functions of the individual random variables X_i, $i = 1, 2, \ldots, n$, it follows that, in the transform domain, the pdf of Y is the n-fold convolution of the pdfs of the X_i. Usually the n-fold convolution is more difficult to perform than the characteristic function method described above in determining the pdf of Y.

When working with n-dimensional random variables, it is appropriate to define and n-dimensional Fourier transform of the joint pdf. In particular, if

X_i, $i = 1, 2, \ldots, n$, are random variables with pdf $p(x_1, x_2, \ldots, x_n)$, the *n-dimensional characteristic function* is defined as

$$\begin{aligned}\psi(jv_1, jv_2, \ldots, jv_n) &= E\left[\exp\left(j\sum_{i=1}^{n} v_i X_i\right)\right] \\ &= \int_{-\infty}^{\infty}\cdots\int_{-\infty}^{\infty}\exp\left(j\sum_{i=1}^{n} v_i x_i\right)p(x_1, x_2, \ldots, x_n)\,dx_1\,dx_2\cdots dx_n \end{aligned} \quad (2\text{-}1\text{-}81)$$

Of special interest is the two-dimensional characteristic function

$$\psi(jv_1, jv_2) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{j(v_1x_1 + v_2x_2)}p(x_1, x_2)\,dx_1\,dx_2 \quad (2\text{-}1\text{-}82)$$

We observe that the partial derivatives of $\psi(jv_1, jv_2)$ with respect to v_1 and v_2 can be used to generate the joint moments. For example, it is easy to show that

$$E(X_1X_2) = -\left.\frac{\partial^2\psi(jv_1, jv_2)}{\partial v_1\,\partial v_2}\right|_{v_1=v_2=0} \quad (2\text{-}1\text{-}83)$$

Higher-order moments are generated in a straightforward manner.

2-1-4 Some Useful Probability Distributions

In subsequent chapters, we shall encounter several different types of random variables. In this section we list these frequently encountered random variables, their pdfs, their cdfs, and their moments. We begin with the binomial distribution, which is the distribution of a discrete random variable, and then we present the distributions of several continuous random variables.

Binomial Distribution Let X be a discrete random variable that has two possible values, say $X = 1$ or $X = 0$, with probabilities p and $1 - p$, respectively. The pdf of X is shown in Fig. 2-1-6. Now, suppose that

$$Y = \sum_{i=1}^{n} X_i$$

where the X_i, $i = 1, 2, \ldots, n$, are statistically independent and identically

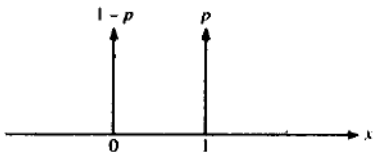


FIGURE 2-1-6 The probability distribution function of X.

distributed random variables with the pdf shown in Fig. 2-1-6. What is the probability distribution function of Y?

To answer this question, we observe that the range of Y is the set of integers from 0 to n. The probability that $Y = 0$ is simply the probability that all the $X_i = 0$. Since the X_i are statistically independent,

$$P(Y = 0) = (1 - p)^n$$

The probability that $Y = 1$ is simply the probability that one $X_i = 1$ and the rest of the $X_i = 0$. Since this event can occur in n different ways,

$$P(Y = 1) = np(1 - p)^{n-1}$$

To generalize, the probability that $Y = k$ is the probability that k of the X_i are equal to one and $n - k$ are equal to zero. Since there are

$$\binom{n}{k} \equiv \frac{n!}{k!\,(n-k)!} \tag{2-1-84}$$

different combinations that result in the event $\{Y = k\}$, it follows that

$$P(Y = k) = \binom{n}{k} p^k (1 - p)^{n-k} \tag{2-1-85}$$

where $\binom{n}{k}$ is the binomial coefficient. Consequently, the pdf of Y may be expressed as

$$\begin{aligned} p(y) &= \sum_{k=0}^{n} P(Y = k)\,\delta(y - k) \\ &= \sum_{k=0}^{n} \binom{n}{k} p^k (1 - p)^{n-k}\,\delta(y - k) \end{aligned} \tag{2-1-86}$$

The cdf of Y is

$$\begin{aligned} F(y) &= P(Y \leq y) \\ &= \sum_{k=0}^{[y]} \binom{n}{k} p^k (1 - p)^{n-k} \end{aligned} \tag{2-1-87}$$

where $[y]$ denotes the largest integer m such that $m \leq y$. The cdf in (2-1-87) characterizes a binomially distributed random variable.

The first two moments of Y are

$$\begin{aligned} E(Y) &= np \\ E(Y^2) &= np(1 - p) + n^2p^2 \\ \sigma^2 &= np(1 - p) \end{aligned} \tag{2-1-88}$$

and the characteristic function is

$$\psi(jv) = (1 - p + pe^{jv})^n \tag{2-1-89}$$

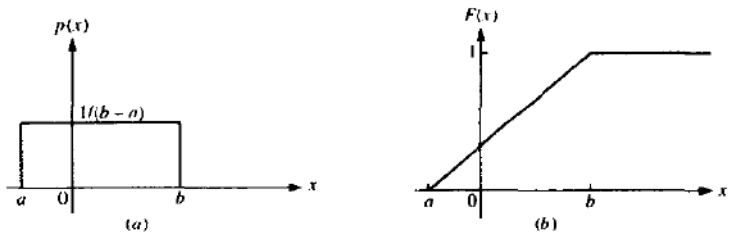


FIGURE 2-1-7 The pdf and cdf of a uniformly distributed random variable.

Uniform Distribution The pdf and cdf of a uniformly distributed random variable X are shown in Fig. 2-1-7. The first two moments of X are

$$\begin{aligned} E(X) &= \tfrac{1}{2}(a+b) \\ E(X^2) &= \tfrac{1}{3}(a^2+b^2+ab) \\ \sigma^2 &= \tfrac{1}{12}(a-b)^2 \end{aligned} \tag{2-1-90}$$

and the characteristic function is

$$\psi(jv) = \frac{e^{jvb} - e^{jva}}{jv(b-a)} \tag{2-1-91}$$

Gaussian (Normal) Distribution The pdf of a gaussian or normally distributed random variable is

$$p(x) = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-(x-m_x)^2/2\sigma^2} \tag{2-1-92}$$

where m_x is the mean and σ^2 is the variance of the random variable. The cdf is

$$\begin{aligned} F(x) &= \int_{-\infty}^{x} p(u)\,du \\ &= \frac{1}{\sqrt{2\pi}\,\sigma} \int_{-\infty}^{x} e^{-(u-m_x)^2/2\sigma^2}\,du \\ &= \frac{1}{2}\frac{2}{\sqrt{\pi}} \int_{-\infty}^{(x-m_x)/\sqrt{2}\sigma} e^{-t^2}\,dt \\ &= \tfrac{1}{2} + \tfrac{1}{2}\,\mathrm{erf}\left(\frac{x-m_x}{\sqrt{2}\,\sigma}\right) \end{aligned} \tag{2-1-93}$$

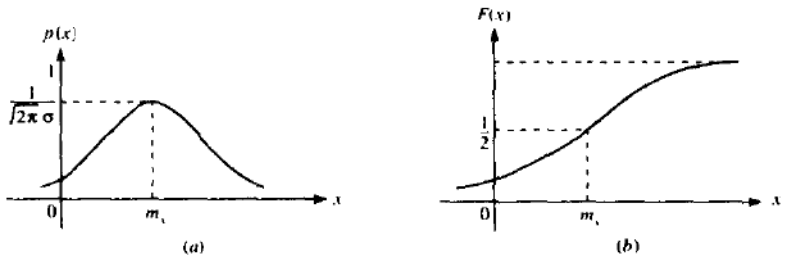


FIGURE 2-1-8 The pdf and cdf of a gaussian-distributed random variable.

where erf (x) denotes the error function, defined as

$$\text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2}\, dt \tag{2-1-94}$$

The pdf and cdf are illustrated in Fig. 2-1-8.

The cdf $F(x)$ may also be expressed in terms of the complementary error function. That is,

$$F(x) = 1 - \tfrac{1}{2}\,\text{erfc}\left(\frac{x - m_x}{\sqrt{2}\,\sigma}\right)$$

where

$$\begin{aligned}\text{erfc}(x) &= \frac{2}{\sqrt{\pi}} \int_x^\infty e^{-t^2}\, dt \\ &= 1 - \text{erf}(x)\end{aligned} \tag{2-1-95}$$

We note that erf $(-x) = -\text{erf}(x)$, erfc $(-x) = 2 - \text{erfc}(x)$, erf $(0) = \text{erfc}(\infty) = 0$, and erf $(\infty) = \text{erfc}(0) = 1$. For $x > m_x$, the complementary error function is proportional to the area under the tail of the gaussian pdf. For large values of x, the complementary error function erfc (x) may be approximated by the asymptotic series

$$\text{erfc}(x) = \frac{e^{-x^2}}{x\sqrt{\pi}}\left(1 - \frac{1}{2x^2} + \frac{1 \cdot 3}{2^2 x^4} - \frac{1 \cdot 3 \cdot 5}{2^3 x^6} + \cdots\right) \tag{2-1-96}$$

where the approximation error is less than the last term used.

The function that is frequently used for the area under the tail of the gaussian pdf is denoted by $Q(x)$ and defined as

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-t^2/2}\, dt, \qquad x \geq 0 \tag{2-1-97}$$

By comparing (2-1-95) with (2-1-97), we find

$$Q(x) = \tfrac{1}{2}\,\text{erfc}\left(\frac{x}{\sqrt{2}}\right) \tag{2-1-98}$$

The characteristic function of a gaussian random variable with mean m_x and variance σ^2 is

$$\psi(jv) = \int_{-\infty}^{\infty} e^{jvx}\left[\frac{1}{\sqrt{2\pi}\,\sigma} e^{-(x-m_x)^2/2\sigma^2}\right] dx$$
$$= e^{jvm_x - (1/2)v^2\sigma^2} \tag{2-1-99}$$

The central moments of a gaussian random variable are

$$E[(X-m_x)^k] \equiv \mu_k = \begin{cases} 1\cdot 3\cdots(k-1)\sigma^k & (\text{even } k) \\ 0 & (\text{odd } k) \end{cases} \tag{2-1-100}$$

and the ordinary moments may be expressed in terms of the central moments as

$$E(X^k) = \sum_{i=0}^{k} \binom{k}{i} m_x^i \mu_{k-i} \tag{2-1-101}$$

The sum of n statistically independent gaussian random variables is also a gaussian random variable. To demonstrate this point, let

$$Y = \sum_{i=1}^{n} X_i \tag{2-1-102}$$

where the X_i, $i = 1, 2, \ldots, n$, are statistically independent gaussian random variables with means m_i and variances σ_i^2. Using the result in (2-1-79), we find that the characteristic function of Y is

$$\psi_Y(jv) = \prod_{i=1}^{n} \psi_{X_i}(jv)$$
$$= \prod_{i=1}^{n} e^{jvm_i - v^2\sigma_i^2/2}$$
$$= e^{jvm_y - v^2\sigma_y^2/2} \tag{2-1-103}$$

where

$$m_y = \sum_{i=1}^{n} m_i$$
$$\sigma_y^2 = \sum_{i=1}^{n} \sigma_i^2 \tag{2-1-104}$$

Therefore, Y is gaussian-distributed with mean m_y and variance σ_y^2.

Chi-Square Distribution A chi-square-distributed random variable is related to a gaussian-distributed random variable in the sense that the former can be viewed as a transformation of the latter. To be specific, let $Y = X^2$, where X is a gaussian random variable. Then Y has a chi-square distribution. We distinguish between two types of chi-square distributions. The first is called a

central chi-square distribution and is obtained when X has zero mean. The second is called a *non-central chi-square distribution*, and is obtained when X has a nonzero mean.

First we consider the central chi-square distribution. Let X be gaussian-distributed with zero mean and variance σ^2. Since $Y = X^2$, the result given in (2-1-47) applies directly with $a = 1$ and $b = 0$. Thus we obtain the pdf of Y in the form

$$p_Y(y) = \frac{1}{\sqrt{2\pi y}\,\sigma} e^{-y/2\sigma^2}, \qquad y \geq 0 \tag{2-1-105}$$

The cdf of Y is

$$\begin{aligned} F_Y(y) &= \int_0^y p_Y(u)\,du \\ &= \frac{1}{\sqrt{2\pi}\,\sigma} \int_0^y \frac{1}{\sqrt{u}} e^{-u/2\sigma^2}\,du \end{aligned} \tag{2-1-106}$$

which cannot be expressed in closed form. The characteristic function, however, can be determined in closed form. It is

$$\psi(jv) = \frac{1}{(1 - j2v\sigma^2)^{1/2}} \tag{2-1-107}$$

Now, suppose that the random variable Y is defined as

$$Y = \sum_{i=1}^{n} X_i^2 \tag{2-1-108}$$

where the X_i, $i = 1, 2, \ldots, n$, are statistically independent and identically distributed gaussian random variables with zero mean and variance σ^2. As a consequence of the statistical independence of the X_i, the characteristic function of Y is

$$\psi_Y(jv) = \frac{1}{(1 - j2v\sigma^2)^{n/2}} \tag{2-1-109}$$

The inverse transform of this characteristic function yields the pdf

$$p_Y(y) = \frac{1}{\sigma^n 2^{n/2}\Gamma(\frac{1}{2}n)} y^{n/2-1} e^{-y/2\sigma^2}, \qquad y \geq 0 \tag{2-1-110}$$

where $\Gamma(p)$ is the gamma function, defined as

$$\begin{aligned} &\Gamma(p) = \int_0^\infty t^{p-1} e^{-t}\,dt, \qquad p > 0 \\ &\Gamma(p) = (p-1)!, \quad p \text{ an integer}, p > 0 \\ &\Gamma(\tfrac{1}{2}) = \sqrt{\pi}, \qquad \Gamma(\tfrac{3}{2}) = \tfrac{1}{2}\sqrt{\pi} \end{aligned} \tag{2-1-111}$$

This pdf, which is a generalization of (2-1-105), is called a *chi-square* (or

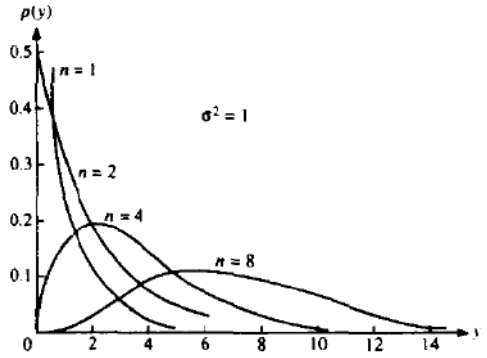


FIGURE 2-1-9 The pdf of a chi-square-distributed random variable for several degrees of freedom.

gamma) *pdf with n degrees of freedom.* It is illustrated in Fig. 2-1-9. The case $n = 2$ yields the exponential distribution.

The first two moments of Y are

$$\begin{aligned} E(Y) &= n\sigma^2 \\ E(Y^2) &= 2n\sigma^4 + n^2\sigma^4 \\ \sigma_y^2 &= 2n\sigma^4 \end{aligned} \tag{2-1-112}$$

The cdf of Y is

$$F_Y(y) = \int_0^y \frac{1}{\sigma^n 2^{n/2}\Gamma(\frac{1}{2}n)} u^{n/2-1} e^{-u/2\sigma^2}\, du, \qquad y \geq 0 \tag{2-1-113}$$

This integral can be easily manipulated into the form of the incomplete gamma function, which is tabulated by Pearson (1965). When n is even, the integral in (2-1-113) can be expressed in closed form. Specifically, let $m = \frac{1}{2}n$, where m is an integer. Then, by repeated integration by parts, we obtain

$$F_Y(y) = 1 - e^{-y/2\sigma^2} \sum_{k=0}^{m-1} \frac{1}{k!}\left(\frac{y}{2\sigma^2}\right)^k, \qquad y \geq 0 \tag{2-1-114}$$

Let us now consider a noncentral chi-square distribution, which results from squaring a gaussian random variable having a nonzero mean. If X is gaussian with mean m_x and variance σ^2, the random variable $Y = X^2$ has the pdf

$$p_Y(y) = \frac{1}{\sqrt{2\pi y}\,\sigma} e^{-(y+m_x^2)/2\sigma^2} \cosh\left(\frac{\sqrt{y}\, m_x}{\sigma^2}\right), \qquad y \geq 0 \tag{2-1-115}$$

which is obtained by applying the result in (2-1-47) to the gaussian pdf given by (2-1-92). The characteristic function corresponding to this pdf is

$$\psi_Y(jv) = \frac{1}{(1 - j2v\sigma^2)^{1/2}} e^{jm_x^2 v/(1 - j2v\sigma^2)} \tag{2-1-116}$$

To generalize these results, let Y be the sum of squares of gaussian random variables as defined by (2-1-108). The X_i, $i = 1, 2, \ldots, n$, are assumed to be statistically independent with means m_i, $i = 1, 2, \ldots, n$, and identical variances equal to σ^2. Then the characteristic function of Y, obtained from (2-1-116) by applying the relation in (2-1-79), is

$$\psi_Y(jv) = \frac{1}{(1 - j2v\sigma^2)^{n/2}} \exp\left(\frac{jv \sum_{i=1}^{n} m_i^2}{1 - j2v\sigma^2}\right) \tag{2-1-117}$$

This characteristic function can be inverse-Fourier-transformed to yield the pdf

$$p_Y(y) = \frac{1}{2\sigma^2}\left(\frac{y}{s^2}\right)^{(n-2)/4} e^{-(s^2+y)/2\sigma^2} I_{n/2-1}\left(\sqrt{y}\,\frac{s}{\sigma^2}\right), \quad y \geq 0 \tag{2-1-118}$$

where, by definition,

$$s^2 = \sum_{i=1}^{n} m_i^2 \tag{2-1-119}$$

and $I_\alpha(x)$ is the αth-order modified Bessel function of the first kind, which may be represented by the infinite series

$$I_\alpha(x) = \sum_{k=0}^{\infty} \frac{(x/2)^{\alpha+2k}}{k!\,\Gamma(\alpha + k + 1)}, \quad x \geq 0 \tag{2-1-120}$$

The pdf given by (2-1-118) is called the *noncentral chi-square pdf with n degrees of freedom*. The parameter s^2 is called the *noncentrality parameter of the distribution*.

The cdf of the noncentral chi square with n degrees of freedom is

$$F_Y(y) = \int_0^y \frac{1}{2\sigma^2}\left(\frac{u}{s^2}\right)^{(n-2)/4} e^{-(s^2+u)/2\sigma^2} I_{n/2-1}\left(\sqrt{u}\,\frac{s}{\sigma^2}\right) du \tag{2-1-121}$$

There is no closed-form expression for this integral. However, when $m = \frac{1}{2}n$ is an integer, the cdf can be expressed in terms of the generalized Marcum's Q function, which is defined as

$$\begin{aligned} Q_m(a, b) &= \int_b^\infty x\left(\frac{x}{a}\right)^{m-1} e^{-(x^2+a^2)/2} I_{m-1}(ax)\, dx \\ &= Q_1(a, b) + e^{(a^2+b^2)/2} \sum_{k=1}^{m-1} \left(\frac{b}{a}\right)^k I_k(ab) \end{aligned} \tag{2-1-122}$$

where

$$Q_1(a, b) = e^{-(a^2+b^2)/2} \sum_{k=0}^{\infty} \left(\frac{a}{b}\right)^k I_k(ab), \quad b > a > 0 \tag{2-1-123}$$

If we change the variable of integration in (2-1-121) from u to x, where

$$x^2 = u/\sigma^2$$

and let $a^2 = s^2/\sigma^2$, then it is easily shown that

$$F_Y(y) = 1 - Q_m\left(\frac{s}{\sigma}, \frac{\sqrt{y}}{\sigma}\right) \tag{2-1-124}$$

Finally, we state that the first two moments of a noncentral chi-square-distributed random variable are

$$\begin{aligned} E(Y) &= n\sigma^2 + s^2 \\ E(Y^2) &= 2n\sigma^4 + 4\sigma^2 s^2 + (n\sigma^2 + s^2)^2 \\ \sigma_y^2 &= 2n\sigma^4 + 4\sigma^2 s^2 \end{aligned} \tag{2-1-125}$$

Rayleigh Distribution The Rayleigh distribution is frequently used to model the statistics of signals transmitted through radio channels such as cellular radio. This distribution is closely related to the central chi-square distribution. To illustrate this point, let $Y = X_1^2 + X_2^2$ where X_1 and X_2 are zero-mean statistically independent gaussian random variables, each having a variance σ^2. From the discussion above, it follows that Y is chi-square-distributed with two degrees of freedom. Hence, the pdf of Y is

$$p_Y(y) = \frac{1}{2\sigma^2} e^{-y/2\sigma^2}, \quad y \geq 0 \tag{2-1-126}$$

Now, suppose we define a new random variable

$$R = \sqrt{X_1^2 + X_2^2} = \sqrt{Y} \tag{2-1-127}$$

Making a simple change of variable in the pdf of (2-1-126), we obtain the pdf of R in the form

$$p_R(r) = \frac{r}{\sigma^2} e^{-r^2/2\sigma^2}, \quad r \geq 0 \tag{2-1-128}$$

This is the pdf of a Rayleigh-distributed random variable. The corresponding cdf is

$$\begin{aligned} F_R(r) &= \int_0^r \frac{u}{\sigma^2} e^{-u^2/2\sigma^2}\, du \\ &= 1 - e^{-r^2/2\sigma^2}, \quad r \geq 0 \end{aligned} \tag{2-1-129}$$

The moments of R are

$$E(R^k) = (2\sigma^2)^{k/2}\Gamma(1 + \tfrac{1}{2}k) \tag{2-1-130}$$

and the variance is

$$\sigma_r^2 = (2 - \tfrac{1}{2}\pi)\sigma^2 \tag{2-1-131}$$

The characteristic function of the Rayleigh-distributed random variable is

$$\psi_R(jv) = \int_0^\infty \frac{r}{\sigma^2} e^{-r^2/2\sigma^2} e^{jvr}\, dr \tag{2-1-132}$$

This integral may be expressed as

$$\begin{aligned}\psi_R(jv) &= \int_0^\infty \frac{r}{\sigma^2} e^{-r^2/2\sigma^2} \cos vr\, dr + j \int_0^\infty \frac{r}{\sigma^2} e^{-r^2/2\sigma^2} \sin vr\, dr \\ &= {}_1F_1(1, \tfrac{1}{2}; -\tfrac{1}{2}v^2\sigma^2) + j\sqrt{\tfrac{1}{2}\pi}\, v\sigma^2 e^{-v^2\sigma^2/2}\end{aligned} \tag{2-1-133}$$

where ${}_1F_1(1, \frac{1}{2}; -a)$ is the confluent hypergeometric function, which is defined as

$${}_1F_1(\alpha, \beta; x) = \sum_{k=0}^{\infty} \frac{\Gamma(\alpha + k)\Gamma(\beta)x^k}{\Gamma(\alpha)\Gamma(\beta + k)k!}, \qquad \beta \neq 0, -1, -2, \ldots \tag{2-1-134}$$

Beaulieu (1990) has shown that ${}_1F_1(1, \frac{1}{2}; -a)$ may be expressed as

$${}_1F_1(1, \tfrac{1}{2}; -a) = -e^{-a} \sum_{k=0}^{\infty} \frac{a^k}{(2k-1)k!} \tag{2-1-135}$$

As a generalization of the above expression, consider the random variable

$$R = \sqrt{\sum_{i=1}^{n} X_i^2} \tag{2-1-136}$$

where the X_i, $i = 1, 2, \ldots, n$, are statistically independent, identically distributed zero mean gaussian random variables. The random variable R has a generalized Rayleigh distribution. Clearly, $Y = R^2$ is chi-square-distributed with n degrees of freedom. Its pdf is given by (2-1-110). A simple change in variable in (2-1-110) yields the pdf of R in the form

$$p_R(r) = \frac{r^{n-1}}{2^{(n-2)/2}\sigma^n \Gamma(\frac{1}{2}n)} e^{-r^2/2\sigma^2}, \qquad r \geq 0 \tag{2-1-137}$$

As a consequence of the functional relationship between the central chi-square and the Rayleigh distributions, the corresponding cdfs are similar. Thus, for any n, the cdf of R can be put in the form of the incomplete gamma function. In the special case when n is even, i.e., $n = 2m$, the cdf of R can be expressed in the closed form

$$F_R(r) = 1 - e^{-r^2/2\sigma^2} \sum_{k=0}^{m-1} \frac{1}{k!} \left(\frac{r^2}{2\sigma^2}\right)^k, \qquad r \geq 0 \tag{2-1-138}$$

Finally, we state that the kth moment of R is

$$E(R^k) = (2\sigma^2)^{k/2} \frac{\Gamma(\frac{1}{2}(n+k))}{\Gamma(\frac{1}{2}n)}, \quad k \geq 0 \tag{2-1-139}$$

which holds for any integer n.

Rice Distribution Just as the Rayleigh distribution is related to the central chi-square distribution, the Rice distribution is related to the noncentral chi-square distribution. To illustrate this relation, let $Y = X_1^2 + X_2^2$, where X_1 and X_2 are statistically independent gaussian random variables with means m_i, $i = 1, 2$, and common variance σ^2. From the previous discussion, we know that Y has a noncentral chi-square distribution with noncentrality parameter $s^2 = m_1^2 + m_2^2$. The pdf of Y, obtained from (2-1-118) for $n = 2$, is

$$p_Y(y) = \frac{1}{2\sigma^2} e^{-(s^2+y)/2\sigma^2} I_0\left(\sqrt{y}\,\frac{s}{\sigma^2}\right), \quad y \geq 0 \tag{2-1-140}$$

Now, we define a new random variable $R = \sqrt{Y}$. The pdf of R, obtained from (2-1-140) by a simple change of variable, is

$$p_R(r) = \frac{r}{\sigma^2} e^{-(r^2+s^2)/2\sigma^2} I_0\left(\frac{rs}{\sigma^2}\right), \quad r \geq 0 \tag{2-1-141}$$

This is the pdf of a Ricean-distributed random variable. As will be shown in Chapter 5, this pdf characterizes the statistics of the envelope of a signal corrupted by additive narrowband gaussian noise. It is also used to model the signal statistics of signals transmitted through some radio channels. The cdf of R is easily obtained by specializing the results in (2-1-124) to the case $m = 1$. This yields

$$F_R(r) = 1 - Q_1\left(\frac{s}{\sigma}, \frac{r}{\sigma}\right), \quad r \geq 0 \tag{2-1-142}$$

where $Q_1(a, b)$ is defined by (2-1-123).

As a generalization of the expressions given above, let R be defined as in (2-1-136) where the X_i, $i = 1, 2, \ldots, n$ are statistically independent gaussian random variables with means m_i, $i = 1, 2, \ldots, n$, and identical variances equal to σ^2. The random variable $R^2 = Y$ has a noncentral chi-square distribution with n degrees of freedom and noncentrality parameter s^2 given by (2-1-119). Its pdf is given by (2-1-118). Hence the pdf of R is

$$p_R(r) = \frac{r^{n/2}}{\sigma^2 s^{(n-2)/2}} e^{-(r^2+s^2)/2\sigma^2} I_{n/2-1}\left(\frac{rs}{\sigma^2}\right), \quad r \geq 0 \tag{2-1-143}$$

and the corresponding cdf is

$$F_R(r) = P(R \leq r) = P(\sqrt{Y} \leq r) = P(Y \leq r^2) = F_Y(r^2) \qquad (2\text{-}1\text{-}144)$$

where $F_Y(r^2)$ is given by (2-1-121). In the special case where $m = \frac{1}{2}n$ is an integer, we have

$$F_R(r) = 1 - Q_m\left(\frac{s}{\sigma}, \frac{r}{\sigma}\right), \quad r \geq 0 \qquad (2\text{-}1\text{-}145)$$

which follows from (2-1-124). Finally, we state that the kth moment of R is

$$E(R^k) = (2\sigma^2)^{k/2} e^{-s^2/2\sigma^2} \frac{\Gamma(\frac{1}{2}(n+k))}{\Gamma(\frac{1}{2}n)} {}_1F_1\left(\frac{n+k}{2}, \frac{n}{2}; \frac{s^2}{2\sigma^2}\right), \quad k \geq 0 \qquad (2\text{-}1\text{-}146)$$

where ${}_1F_1(\alpha, \beta; x)$ is the confluent hypergeometric function.

Nakagami m-Distribution Both the Rayleigh distribution and the Rice distribution are frequently used to describe the statistical fluctuations of signals received from a multipath fading channel. These channel models are considered in Chapter 14. Another distribution that is frequently used to characterize the statistics of signals transmitted through multipath fading channels is the Nakagami m-distribution. The pdf for this distribution is given by Nakagami (1960) as

$$p_R(r) = \frac{2}{\Gamma(m)} \left(\frac{m}{\Omega}\right)^m r^{2m-1} e^{-mr^2/\Omega} \qquad (2\text{-}1\text{-}147)$$

where Ω is defined as

$$\Omega = E(R^2) \qquad (2\text{-}1\text{-}148)$$

and the parameter m is defined as the ratio of moments, called the *fading figure*,

$$m = \frac{\Omega^2}{E[(R^2 - \Omega)^2]}, \quad m \geq \tfrac{1}{2} \qquad (2\text{-}1\text{-}149)$$

A normalized version of (2-1-147) may be obtained by defining another random variable $X = R/\sqrt{\Omega}$ (see Problem 2-15). The nth moment of R is

$$E(R^n) = \frac{\Gamma(m + \frac{1}{2}n)}{\Gamma(m)} \left(\frac{\Omega}{m}\right)^{n/2}$$

By setting $m = 1$, we observe that (2-1-147) reduces to a Rayleigh pdf. For

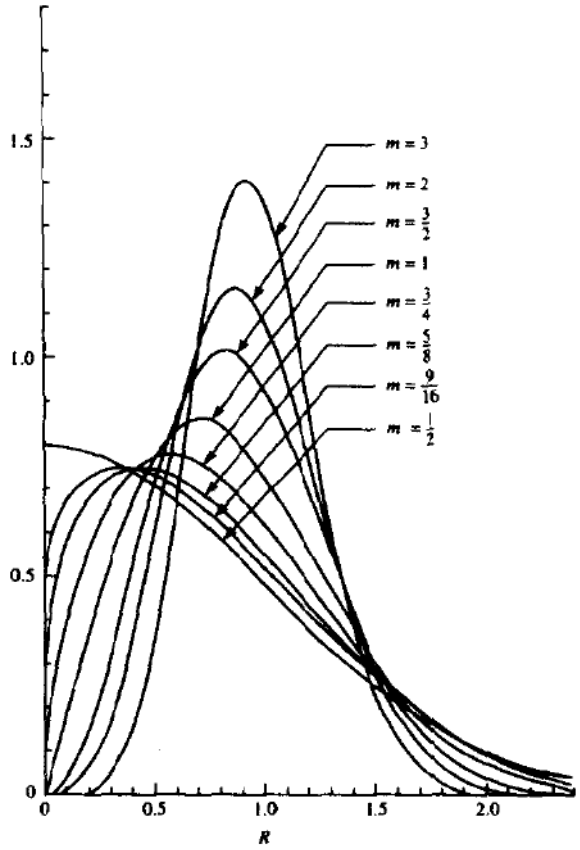


FIGURE 2-1-10 The m-distributed pdf, shown with $\Omega = 1$. m is the fading figure. (*Miyagaki* et al. *1978.*)

values of m in the range $\frac{1}{2} \leq m \leq 1$, we obtain pdfs that have larger tails than a Rayleigh-distributed random variable. For values of $m > 1$, the tail of the pdf decays faster than that of the Rayleigh. Figure 2-1-10 illustrates the pdfs for different values of m.

Multivariate Gaussian Distribution Of the many multivariate or multidimensional distributions that can be defined, the multivariate gaussian distribution is the most important and the one most likely to be encountered in practice. We shall briefly introduce this distribution and state its basic properties.

Let us assume that X_i, $i = 1, 2, \ldots, n$, are gaussian random variables with means m_i, $i = 1, 2, \ldots, n$, variances σ_i^2, $i = 1, 2, \ldots, n$, and covariances μ_{ij}, $i, j = 1, 2, \ldots, n$. Clearly, $\mu_{ii} = \sigma_i^2$, $i = 1, 2, \ldots, n$. Let $\mathbf{M}$ denote the $n \times n$

covariance matrix with elements $\{\mu_{ij}\}$, let $\mathbf{X}$ denote the $n \times 1$ column vector of random variables, and let $\mathbf{m}_x$ denote the $n \times 1$ column vector of mean values m_i, $i = 1, 2, \ldots, n$. The joint pdf of the gaussian random variables X_i, $i = 1, 2, \ldots, n$, is defined as

$$p(x_1, x_2, \ldots, x_n) = \frac{1}{(2\pi)^{n/2}(\det \mathbf{M})^{1/2}} \exp\left[-\tfrac{1}{2}(\mathbf{x} - \mathbf{m}_x)'\mathbf{M}^{-1}(\mathbf{x} - \mathbf{m}_x)\right] \tag{2-1-150}$$

where $\mathbf{M}^{-1}$ denotes the inverse of $\mathbf{M}$ and $\mathbf{x}'$ denotes the transpose of $\mathbf{x}$.

The characteristic function corresponding to this n-dimensional joint pdf is

$$\psi(j\mathbf{v}) = E(e^{j\mathbf{v}'\mathbf{x}})$$

where $\mathbf{v}$ is an n-dimensional vector with elements v_i, $i = 1, 2, \ldots, n$. Evaluation of this n-dimensional Fourier transform yields the result

$$\psi(j\mathbf{v}) = \exp\left(j\mathbf{m}_x'\mathbf{v} - \tfrac{1}{2}\mathbf{v}'\mathbf{M}\mathbf{v}\right) \tag{2-1-151}$$

An important special case of (2-1-150) is the bivariate or two-dimensional gaussian pdf. The mean $\mathbf{m}_x$ and the covariance matrix $\mathbf{M}$ for this case are

$$\mathbf{m}_x = \begin{bmatrix} m_1 \\ m_2 \end{bmatrix}, \qquad \mathbf{M} = \begin{bmatrix} \sigma_1^2 & \mu_{12} \\ \mu_{12} & \sigma_2^2 \end{bmatrix} \tag{2-1-152}$$

where the joint central moment μ_{12} is defined as

$$\mu_{12} = E[(X_1 - m_1)(X_2 - m_2)]$$

It is convenient to define a normalized covariance

$$\rho_{ij} = \frac{\mu_{ij}}{\sigma_i \sigma_j}, \qquad i \neq j \tag{2-1-153}$$

where ρ_{ij} satisfies the condition $0 \leq |\rho_{ij}| \leq 1$. When dealing with the two-dimensional case, it is customary to drop the subscripts on μ_{12} and ρ_{12}. Hence the covariance matrix is expressed as

$$\mathbf{M} = \begin{bmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{bmatrix} \tag{2-1-154}$$

Its inverse is

$$\mathbf{M}^{-1} = \frac{1}{\sigma_1^2\sigma_2^2(1 - \rho^2)} \begin{bmatrix} \sigma_2^2 & -\rho\sigma_1\sigma_2 \\ -\rho\sigma_1\sigma_2 & \sigma_1^2 \end{bmatrix} \tag{2-1-155}$$

and $\det \mathbf{M} = \sigma_1^2\sigma_2^2(1-\rho^2)$. Substitution for $\mathbf{M}^{-1}$ into (2-1-150) yields the desired bivariate gaussian pdf in the form

$$p(x_1, x_2) = \frac{1}{2\pi\sigma_1\sigma_2\sqrt{1-\rho^2}} \times \exp\left[-\frac{\sigma_2^2(x_1-m_1)^2 - 2\rho\sigma_1\sigma_2(x_1-m_1)(x_2-m_2) + \sigma_1^2(x_2-m_2)^2}{2\sigma_1^2\sigma_2^2(1-\rho^2)}\right] \tag{2-1-156}$$

We note that when $\rho = 0$, the joint pdf $p(x_1, x_2)$ in (2-1-156) factors into the product $p(x_1)p(x_2)$, where $p(x_i)$, $i = 1, 2$, are the marginal pdfs. Since ρ is a measure of the correlation between X_1 and X_2, we have shown that when the gaussian random variables X_1 and X_2 are uncorrelated, they are also statistically independent. This is an important property of gaussian random variables, which does not hold in general for other distributions. It extends to n-dimensional gaussian random variables in a straightforward manner. That is, if $\rho_{ij} = 0$ for $i \neq j$ then the random variables X_i, $i = 1, 2, \ldots, n$ are uncorrelated and, hence, statistically independent.

Now, let us consider a linear transformation of n gaussian random variables X_i, $i = 1, 2, \ldots, n$, with mean vector $\mathbf{m}_x$ and covariance matrix $\mathbf{M}$. Let

$$\mathbf{Y} = \mathbf{AX} \tag{2-1-157}$$

where $\mathbf{A}$ is a nonsingular matrix. As shown previously, the jacobian of this transformation is $J = 1/\det \mathbf{A}$. Since $\mathbf{X} = \mathbf{A}^{-1}\mathbf{Y}$, we may substitute for $\mathbf{X}$ in (2-1-150) and, thus, we obtain the joint pdf of $\mathbf{Y}$ in the form

$$p(\mathbf{y}) = \frac{1}{(2\pi)^{n/2}(\det \mathbf{M})^{1/2} \det \mathbf{A}} \exp\left[-\tfrac{1}{2}(\mathbf{A}^{-1}\mathbf{y} - \mathbf{m}_x)'\mathbf{M}^{-1}(\mathbf{A}^{-1}\mathbf{y} - \mathbf{m}_x)\right]$$

$$= \frac{1}{(2\pi)^{n/2}(\det \mathbf{Q})^{1/2}} \exp\left[-\tfrac{1}{2}(\mathbf{y} - \mathbf{m}_y)'\mathbf{Q}^{-1}(\mathbf{y} - \mathbf{m}_y)\right] \tag{2-1-158}$$

where the vector $\mathbf{m}_y$ and the matrix $\mathbf{Q}$ are defined as

$$\begin{aligned} \mathbf{m}_y &= \mathbf{A}\mathbf{m}_x \\ \mathbf{Q} &= \mathbf{AMA} \end{aligned} \tag{2-1-159}$$

Thus we have shown that a linear transformation of a set of jointly gaussian random variables results in another set of jointly gaussian random variables.

Suppose that we wish to perform a linear transformation that results in n statistically independent gaussian random variables. How should the matrix $\mathbf{A}$ be selected? From our previous discussion, we know that the gaussian random

variables are statistically independent if they are pairwise-uncorrelated, i.e., if the covariance matrix **Q** is diagonal. Therefore, we must have

$$\mathbf{AMA}' = \mathbf{D} \tag{2-1-160}$$

where **D** is a diagonal matrix. The matrix **M** is a covariance matrix; hence, it is positive definite. One solution is to select **A** to be an orthogonal matrix ($\mathbf{A}' = \mathbf{A}^{-1}$) consisting of columns that are the eigenvectors of the covariance matrix **M**. Then **D** is a diagonal matrix with diagonal elements equal to the eigenvalues of **M**.

Example 2-1-5

Consider the bivariate gaussian pdf with covariance matrix

$$\mathbf{M} = \begin{bmatrix} 1 & \frac{1}{2} \\ \frac{1}{2} & 1 \end{bmatrix}$$

Let us determine the transformation **A** that will result in uncorrelated random variables. First, we solve for the eigenvalues of **M**. The characteristic equation is

$$\det(\mathbf{M} - \lambda\mathbf{I}) = 0$$

$$(1 - \lambda)^2 - \tfrac{1}{4} = 0$$

$$\lambda = \tfrac{3}{2}, \tfrac{1}{2}$$

Next we determine the two eigenvectors. If **a** denotes an eigenvector, we have

$$(\mathbf{M} - \lambda\mathbf{I})\mathbf{a} = 0$$

With $\lambda_1 = \frac{3}{2}$ and $\lambda_2 = \frac{1}{2}$, we obtain the eigenvectors

$$\mathbf{a}_1 = \begin{bmatrix} \sqrt{\frac{1}{2}} \\ \sqrt{\frac{1}{2}} \end{bmatrix}, \quad \mathbf{a}_2 = \begin{bmatrix} \sqrt{\frac{1}{2}} \\ -\sqrt{\frac{1}{2}} \end{bmatrix}$$

Therefore,

$$\mathbf{A} = \sqrt{\tfrac{1}{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

It is easily verified that $\mathbf{A}^{-1} = \mathbf{A}'$ and that

$$\mathbf{AMA}' = \mathbf{D}$$

where the diagonal elements of **D** are $\frac{3}{2}$ and $\frac{1}{2}$.

2-1-5 Upper Bounds on the Tail Probability

In evaluating the performance of a digital communication system, it is often necessary to determine the area under the tail of the pdf. We refer to this area as the *tail probability*. In this section, we present two upper bounds on the tail probability. The first, obtained from the Chebyshev inequality, is rather loose. The second, called the *Chernoff bound*, is much tighter.

Chebyshev Inequality Suppose that X is an arbitrary random variable with finite mean m_x and finite variance σ_x^2. For any positive number δ,

$$P(|X - m_x| \geq \delta) \leq \frac{\sigma_x^2}{\delta^2} \tag{2-1-161}$$

This relation is called the *Chebyshev inequality*. The proof of this bound is relatively simple. We have

$$\sigma_x^2 = \int_{-\infty}^{\infty} (x - m_x)^2 p(x)\, dx \geq \int_{|x - m_x| \geq \delta} (x - m_x)^2 p(x)\, dx$$

$$\geq \delta^2 \int_{|x - m_x| \geq \delta} p(x)\, dx = \delta^2 P(|X - m_x| \geq \delta)$$

Thus the validity of the inequality is established.

It is apparent that the Chebyshev inequality is simply an upper bound on the area under the tails of the pdf $p(y)$, where $Y = X - m_x$, i.e., the area of $p(y)$ in the intervals $(-\infty, -\delta)$ and (δ, ∞). Hence, the Chebyshev inequality may be expressed as

$$1 - [F_Y(\delta) - F_Y(-\delta)] \leq \frac{\sigma_x^2}{\delta^2} \tag{2-1-162}$$

or, equivalently, as

$$1 - [F_X(m_x + \delta) - F_X(m_x - \delta)] \leq \frac{\sigma_x^2}{\delta^2} \tag{2-1-163}$$

There is another way to view the Chebyshev bound. Working with the zero mean random variable $Y = X - m_x$, for convenience, suppose we define a function $g(Y)$ as

$$g(Y) = \begin{cases} 1 & (|Y| \geq \delta) \\ 0 & (|Y| < \delta) \end{cases} \tag{2-1-164}$$

Since $g(Y)$ is either 0 or 1 with probabilities $P(|Y| < \delta)$ and $P(|Y| \geq \delta)$, respectively, its mean value is

$$E[g(Y)] = P(|Y| \geq \delta) \tag{2-1-165}$$

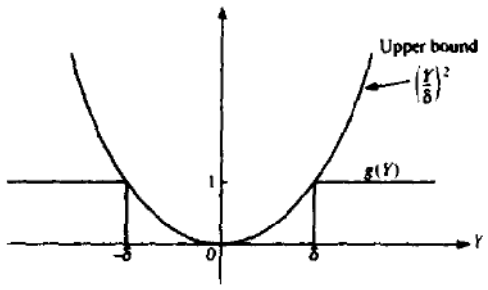


FIGURE 2-1-11 A quadratic upper bound on $g(Y)$ used in obtaining the tail probability (Chebyshev bound).

Now suppose that we upper-bound $g(Y)$ by the quadratic $(Y/\delta)^2$, i.e.,

$$g(Y) \leq \left(\frac{Y}{\delta}\right)^2 \tag{2-1-166}$$

The graph of $g(Y)$ and the upper bound are shown in Fig. 2-1-11. It follows that

$$E[g(Y)] \leq E\left(\frac{Y^2}{\delta^2}\right) = \frac{E(Y^2)}{\delta^2} = \frac{\sigma_y^2}{\delta^2} = \frac{\sigma_x^2}{\delta^2}$$

Since $E[g(Y)]$ is the tail probability, as seen from (2-1-165), we have obtained the Chebyshev bound.

For many practical applications, the Chebyshev bound is extremely loose. The reason for this may be attributed to the looseness of the quadratic $(Y/\delta)^2$ in overbounding $g(Y)$. There are certainly many other functions that can be used to overbound $g(Y)$. Below, we use an exponential bound to derive an upper bound on the tail probability that is extremely tight.

Chernoff Bound The Chebyshev bound given above involves the area under the two tails of the pdf. In some applications we are interested only in the area under one tail, either in the interval (δ, ∞) or in the interval $(-\infty, \delta)$. In such a case we can obtain an extremely tight upper bound by overbounding the function $g(Y)$ by an exponential having a parameter that can be optimized to yield as tight an upper bound as possible. Specifically, we consider the tail probability in the interval (δ, ∞). The function $g(Y)$ is overbounded as

$$g(Y) \leq e^{\nu(Y-\delta)} \tag{2-1-167}$$

where $g(Y)$ is now defined as

$$g(Y) = \begin{cases} 1 & (Y \geq \delta) \\ 0 & (Y < \delta) \end{cases} \tag{2-1-168}$$

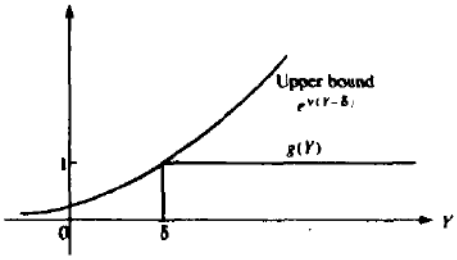


FIGURE 2-1-12 An exponential upper bound on $g(Y)$ used in obtaining the tail probability (Chernoff bound).

and $v \geq 0$ is the parameter to be optimized. The graph of $g(Y)$ and the exponential upper bound are shown in Fig. 2-1-12.

The expected value of $g(Y)$ is

$$E[g(Y)] = P(Y \geq \delta) \leq E(e^{v(Y-\delta)}) \tag{2-1-169}$$

This bound is valid for any $v \geq 0$. The tightest upper bound is obtained by selecting the value of v that minimizes $E(e^{v(Y-\delta)})$. A necessary condition for a minimum is

$$\frac{d}{dv} E(e^{v(Y-\delta)}) = 0 \tag{2-1-170}$$

But the order of differentiation and expectation can be interchanged, so that

$$\begin{aligned} \frac{d}{dv} E(e^{v(Y-\delta)}) &= E\left(\frac{d}{dv} e^{v(Y-\delta)}\right) \\ &= E[(Y-\delta)e^{v(Y-\delta)}] \\ &= e^{-v\delta}[E(Ye^{vY}) - \delta E(e^{vY})] = 0 \end{aligned}$$

Therefore the value of v that gives the tightest upper bound is the solution to the equation

$$E(Ye^{vY}) - \delta E(e^{vY}) = 0 \tag{2-1-171}$$

Let $\hat{v}$ be the solution of (2-1-171). Then, from (2-1-169), the upper bound on the one-sided tail probability is

$$P(Y \geq \delta) \leq e^{-\hat{v}\delta} E(e^{\hat{v}Y}) \tag{2-1-172}$$

This is the Chernoff bound for the upper tail probability for a discrete or a continuous random variable having a zero mean.† This bound may be used to show that $Q(x) \leq e^{-x^2/2}$, where $Q(x)$ is the area in the tail of the gaussian pdf (see Problem 2-18).

† Note that $E(e^{vY})$ for real v is not the characteristic function of Y. It is called the *moment generating function of Y*.

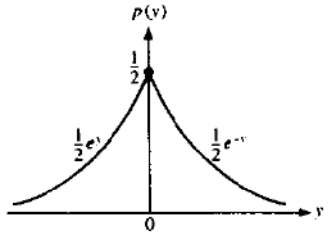


FIGURE 2-1-13 The pdf of a Laplace-distributed random variable.

An upper bound on the lower tail probability can be obtained in a similar manner, with the result that

$$P(Y \leq \delta) \leq e^{-\hat{\nu}\delta} E(e^{\hat{\nu}Y}) \tag{2-1-173}$$

where $\hat{\nu}$ is the solution to (2-1-171) and $\delta < 0$.

Example 2-1-6

Consider the (Laplace) pdf

$$p(y) = \tfrac{1}{2} e^{-|y|} \tag{2-1-174}$$

which is illustrated in Fig. 2-1-13. Let us evaluate the upper tail probability from the Chernoff bound and compare it with the true tail probability, which is

$$P(Y \geq \delta) = \int_{\delta}^{\infty} \tfrac{1}{2} e^{-y}\, dy = \tfrac{1}{2} e^{-\delta} \tag{2-1-175}$$

To solve (2-1-171) for $\hat{\nu}$, we must determine the moments $E(Ye^{\nu Y})$ and $E(e^{\nu Y})$. For the pdf in (2-1-174), we find that

$$\begin{aligned} E(Ye^{\nu Y}) &= \frac{2\nu}{(\nu+1)^2(\nu-1)^2} \\ E(e^{\nu Y}) &= \frac{1}{(1+\nu)(1-\nu)} \end{aligned} \tag{2-1-176}$$

Substituting these moments into (2-1-171), we obtain the quadratic equation

$$\nu^2\delta + 2\nu - \delta = 0$$

which has the solutions

$$\hat{\nu} = \frac{-1 \pm \sqrt{1+\delta^2}}{\delta} \tag{2-1-177}$$

Since $\hat{\nu}$ must be positive, one of the two solutions is discarded. Thus

$$\hat{\nu} = \frac{-1 + \sqrt{1+\delta^2}}{\delta} \tag{2-1-178}$$

Finally, we evaluate the upper bound in (2-1-172) by eliminating $E(e^{\nu Y})$ using the second relation in (2-1-176) and by substituting for ν from (2-1-178). The result is

$$P(Y \geq \delta) \leq \frac{\delta^2}{2(-1+\sqrt{1+\delta^2})} e^{1-\sqrt{1+\delta^2}} \tag{2-1-179}$$

For $\delta \gg 1$, (2-1-179) reduces to

$$P(Y \geq \delta) \leq \frac{\delta}{2} e^{-\delta} \tag{2-1-180}$$

We note that the Chernoff bound decreases exponentially as δ increases. Consequently, it approximates closely the exact tail probability given by (2-1-175). In contrast, the Chebyshev upper bound for the upper tail probability obtained by taking one-half of the probability in the two tails (due to symmetry in the pdf) is

$$P(Y \geq \delta) \leq \frac{1}{\delta^2}$$

Hence, this bound is extremely loose.

When the random variable has a nonzero mean, the Chernoff bound can be extended as we now demonstrate. If $Y = X - m_x$, we have

$$P(Y \geq \delta) = P(X - m_x \geq \delta) = P(X \geq m_x + \delta) = P(X \geq \delta_m)$$

where, by definition, $\delta_m = m_x + \delta$. Since $\delta > 0$, it follows that $\delta_m > m_x$. Let $g(X)$ be defined as

$$g(X) = \begin{cases} 1 & (X \geq \delta_m) \\ 0 & (X < \delta_m) \end{cases} \tag{2-1-181}$$

and upper-bounded as

$$g(X) \leq e^{\nu(X-\delta_m)} \tag{2-1-182}$$

From this point, the derivation parallels the steps contained in (2-1-169)–(2-1-172). The final result is

$$P(X \geq \delta_m) \leq e^{-\nu\delta_m} E(e^{\nu X}) \tag{2-1-183}$$

where $\delta_m > m_x$ and ν is the solution to the equation

$$E(Xe^{\nu X}) - \delta_m E(e^{\nu X}) = 0 \tag{2-1-184}$$

In a similar manner, we can obtain the Chernoff bound for the lower tail probability. For $\delta < 0$, we have

$$P(X - m_x \leq \delta) = P(X \leq m_x + \delta) = P(X \leq \delta_m) \leq E(e^{\nu(X-\delta_m)}) \tag{2-1-185}$$

From our previous development, it is apparent that (2-1-185) results in the bound

$$P(X \leq \delta_m) \leq e^{-\nu\delta_m} E(e^{\nu X}) \tag{2-1-186}$$

where $\delta_m < m_x$ and ν is the solution to (2-1-184).

2-1-6 Sums of Random Variables and the Central Limit Theorem

We have previously considered the problem of determining the pdf of a sum of n statistically independent random variables. In this section, we again consider the sum of statistically independent random variables, but our approach is different and is independent of the particular pdf of the random variables in the sum. To be specific, suppose that X_i, $i = 1, 2, \ldots, n$, are statistically independent and identically distributed random variables, each having a finite mean m_x and a finite variance σ_x^2. Let Y be defined as the normalized sum, called the *sample mean*:

$$Y = \frac{1}{n}\sum_{i=1}^{n} X_i \tag{2-1-187}$$

First we shall determine upper bounds on the tail probabilities of Y and then we shall prove a very important theorem regarding the pdf of Y in the limit as $n \to \infty$.

The random variable Y defined in (2-1-187) is frequently encountered in estimating the mean of a random variable X from a number of observations X_i, $i = 1, 2, \ldots, n$. In other words, the X_i, $i = 1, 2, \ldots, n$, may be considered as independent samples drawn from a distribution $F_X(x)$, and Y is the estimate of the mean m_x.

The mean of Y is

$$E(Y) = m_y = \frac{1}{n}\sum_{i=1}^{n} E(X_i)$$

$$= m_x$$

The variance of Y is

$$\sigma_y^2 = E(Y^2) - m_y^2 = E(Y^2) - m_x^2$$

$$= \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n} E(X_i X_j) - m_x^2$$

$$= \frac{1}{n^2}\sum_{i=1}^{n} E(X_i^2) + \frac{1}{n^2}\sum_{\substack{i=1 \\ i\neq j}}^{n}\sum_{j=1}^{n} E(X_i)E(X_j) - m_x^2$$

$$= \frac{1}{n}(\sigma_x^2 + m_x^2) + \frac{1}{n^2}n(n-1)m_x^2 - m_x^2$$

$$= \frac{\sigma_x^2}{n}$$

When Y is viewed as an estimate for the mean m_x, we note that its expected value is equal to m_x and its variance decreases inversely with the number of

samples n. As n approaches infinity, the variance σ_y^2 approaches zero. An estimate of a parameter (in this case the mean m_x) that satisfies the conditions that its expected value converges to the true value of the parameter and the variance converges to zero as $n \to \infty$ is said to be a *consistent estimate*.

The tail probability of the random variable Y can be upper-bounded by use of the bounds presented in Section 2-1-5. The Chebyshev inequality applied to Y is

$$P(|Y - m_y| \geq \delta) \leq \frac{\sigma_y^2}{\delta^2}$$
$$P\left(\left|\frac{1}{n}\sum_{i=1}^{n} X_i - m_x\right| \geq \delta\right) \leq \frac{\sigma_x^2}{n\delta^2} \tag{2-1-188}$$

In the limit as $n \to \infty$, (2-1-188) becomes

$$\lim_{n\to\infty} P\left(\left|\frac{1}{n}\sum_{i=1}^{n} X_i - m_x\right| \geq \delta\right) = 0 \tag{2-1-189}$$

Therefore, the probability that the estimate of the mean differs from the true mean m_x by more than δ ($\delta > 0$) approaches zero as n approaches infinity. This statement is a form of the law of large numbers. Since the upper bound converges to zero relatively slowly, i.e., inversely with n, the expression in (2-1-188) is called the *weak law of large numbers*.

The Chernoff bound applied to the random variable Y yields an exponential dependence of n, and thus provides a tighter upper bound on the one-sided tail probability. Following the procedure developed in Section 2-1-5, we can determine that the tail probability for y is

$$P(Y - m_y \geq \delta) = P\left(\frac{1}{n}\sum_{i=1}^{n} X_i - m_x \geq \delta\right)$$
$$= P\left(\sum_{i=1}^{n} X_i \geq n\delta_m\right) \leq E\left\{\exp\left[v\left(\sum_{i=1}^{n} X_i - n\delta_m\right)\right]\right\} \tag{2-1-190}$$

where $\delta_m = m_x + \delta$ and $\delta > 0$. But the X_i, $i = 1, 2, \ldots, n$, are statistically independent and identically distributed. Hence,

$$E\left\{\exp\left[v\left(\sum_{i=1}^{n} X_i - n\delta_m\right)\right]\right\} = e^{-vn\delta_m} E\left[\exp\left(v\sum_{i=1}^{n} X_i\right)\right]$$
$$= e^{-vn\delta_m} \prod_{i=1}^{n} E(e^{vX_i})$$
$$= [e^{-v\delta_m} E(e^{vX})]^n \tag{2-1-191}$$

where X denotes any one of the X_i. The parameter v that yields the tightest upper bound is obtained by differentiating (2-1-191) and setting the derivative equal to zero. This yields the equation

$$E(Xe^{vX}) - \delta_m E(e^{vX}) = 0 \tag{2-1-192}$$

Let the solution of (2-1-192) be denoted by $\hat{\nu}$. Then, the bound on the upper tail probability is

$$P\left(\frac{1}{n}\sum_{i=1}^{n} X_i \geq \delta_m\right) \leq [e^{-\hat{\nu}\delta_m} E(e^{\hat{\nu}X})]^n, \qquad \delta_m > m_x \tag{2-1-193}$$

In a similar manner, we find that the lower tail probability is upper-bounded as

$$P(Y \leq \delta_m) \leq [e^{-\hat{\nu}\delta_m} E(e^{\hat{\nu}X})]^n, \qquad \delta_m < m_x \tag{2-1-194}$$

where $\hat{\nu}$ is the solution to (2-1-192).

Example 2-1-7

Let X_i, $i = 1, 2, \ldots, n$, be a set of statistically independent random variables defined as

$$X_i = \begin{cases} 1 & \text{with probability } p < \frac{1}{2} \\ -1 & \text{with probability } 1 - p \end{cases}$$

We wish to determine a tight upper bound on the probability that the sum of the X_i is greater than zero. Since $p < \frac{1}{2}$, we note that the sum will have a negative value for the mean; hence we seek the upper tail probability. With $\delta_m = 0$ in (2-1-193), we have

$$P\left(\sum_{i=1}^{n} X_i \geq 0\right) \leq [E(e^{\hat{\nu}X})]^n \tag{2-1-195}$$

where $\hat{\nu}$ is the solution to the equation

$$E(Xe^{\nu X}) = 0 \tag{2-1-196}$$

Now

$$E(Xe^{\nu X}) = -(1-p)e^{-\nu} + pe^{\nu} = 0$$

Hence

$$\hat{\nu} = \ln\left(\sqrt{\frac{1-p}{p}}\right) \tag{2-1-197}$$

Furthermore,

$$E(e^{\hat{\nu}X}) = pe^{\hat{\nu}} + (1-p)e^{-\hat{\nu}}$$

Therefore the bound in (2-1-195) becomes

$$\begin{aligned} P\left(\sum_{i=1}^{n} X_i \geq 0\right) &\leq [pe^{\hat{\nu}} + (1-p)e^{-\hat{\nu}}]^n \\ &\leq \left[p\sqrt{\frac{1-p}{p}} + (1-p)\sqrt{\frac{p}{1-p}}\right]^n \\ &\leq [4p(1-p)]^{n/2} \end{aligned} \tag{2-1-198}$$

We observe that the upper bound decays exponentially with n, as expected.

In contrast, if the Chebyshev bound were evaluated, the tail probability would decrease inversely with n.

Central Limit Theorem We conclude this section with an extremely useful theorem concerning the cdf of a sum of random variables in the limit as the number of terms in the sum approaches infinity. There are several versions of this theorem. We shall prove the theorem for the case in which the random variables X_i, $i = 1, 2, \ldots, n$, being summed are statistically independent and identically distributed, each having a finite mean m_x and a finite variance σ_x^2. For convenience, we define the normalized random variable

$$U_i = \frac{X_i - m_x}{\sigma_x}, \qquad i = 1, 2, \ldots, n$$

Thus U_i has a zero mean and unit variance. Now, let

$$Y = \frac{1}{\sqrt{n}} \sum_{i=1}^{n} U_i \tag{2-1-199}$$

Since each term in the sum has a zero mean and unit variance, it follows that the normalized (by $1/\sqrt{n}$) random variable Y has zero mean and unit variance. We wish to determine the cdf of Y in the limit as $n \to \infty$.

The characteristic function of Y is

$$\begin{aligned} \psi_Y(jv) = E(e^{jvY}) &= E\left[\exp\left(\frac{jv \sum_{i=1}^{n} U_i}{\sqrt{n}}\right)\right] \\ &= \prod_{i=1}^{n} \psi_{U_i}\left(\frac{jv}{\sqrt{n}}\right) \\ &= \left[\psi_U\left(\frac{jv}{\sqrt{n}}\right)\right]^n \end{aligned} \tag{2-1-200}$$

where U denotes any of the U_i, which are identically distributed. Now, let us expand the characteristic function of U in a Taylor series. The expansion yields

$$\psi_U\left(j\frac{v}{\sqrt{n}}\right) = 1 + j\frac{v}{\sqrt{n}}E(U) - \frac{v^2}{n2!}E(U^2) + \frac{(jv)^3}{(\sqrt{n})^3 3!}E(U^3) - \ldots \tag{2-1-201}$$

Since $E(U) = 0$ and $E(U^2) = 1$, (2-1-201) simplifies to

$$\psi_U\left(\frac{jv}{\sqrt{n}}\right) = 1 - \frac{v^2}{2n} + \frac{1}{n}R(v, n) \tag{2-1-202}$$

where $R(v, n)/n$ denotes the remainder. We note that $R(v, n)$ approaches

zero as $n \to \infty$. Substitution of (2-1-202) into (2-1-200) yields the characteristic function of Y in the form

$$\psi_Y(jv) = \left[1 - \frac{v^2}{2n} + \frac{R(v, n)}{n}\right]^n \tag{2-1-203}$$

Taking the natural logarithm of (2-1-203), we obtain

$$\ln \psi_Y(jv) = n \ln \left[1 - \frac{v^2}{2n} + \frac{R(v, n)}{n}\right] \tag{2-1-204}$$

For small values of x, $\ln (1 + x)$ can be expanded in the power series

$$\ln (1 + x) = x - \tfrac{1}{2}x^2 + \tfrac{1}{3}x^3 - \ldots$$

This expansion applied to (2-1-204) yields

$$\ln \psi_Y(jv) = n\left[-\frac{v^2}{2n} + \frac{R(v, n)}{n} - \frac{1}{2}\left(-\frac{v^2}{2n} + \frac{R(v, n)}{n}\right)^2 + \ldots\right] \tag{2-1-205}$$

Finally, when we take the limit as $n \to \infty$, (2-1-205) reduces to $\lim_{n\to\infty} \ln \psi_Y(jv) = -\frac{1}{2}v^2$, or, equivalently,

$$\lim_{n\to\infty} \psi_Y(jv) = e^{-v^2/2} \tag{2-1-206}$$

But, this is just the characteristic function of a gaussian random variable with zero mean and unit variance. Thus we have the important result that the sum of statistically independent and identically distributed random variables with finite mean and variance approaches a gaussian cdf as $n \to \infty$. This result is known as the *central limit theorem.*

Although we assumed that the random variables in the sum are identically distributed, the assumption can be relaxed provided that additional restrictions are imposed on the properties of the random variables. There is one variation of the theorem, for example, in which the assumption of identically distributed random variables is abandoned in favor of a condition on the third absolute moment of the random variables in the sum. For a discussion of this and other variations of the central limit theorem, the reader is referred to the book by Cramer (1946).

2-2 STOCHASTIC PROCESSES

Many of the random phenomena that occur in nature are functions of time. For example, the meteorological phenomena such as the random fluctuations in air temperature and air pressure are functions of time. The thermal noise voltages generated in the resistors of an electronic device such as a radio receiver are also a function of time. Similarly, the signal at the output of a source that generates information is characterized as a random signal that

varies with time. An audio signal that is transmitted over a telephone channel is an example of such a signal. All these are examples of stochastic (random) processes. In our study of digital communications, we encounter stochastic processes in the characterization and modeling of signals generated by information sources, in the characterization of communication channels used to transmit the information, in the characterization of noise generated in a receiver, and in the design of the optimum receiver for processing the received random signal.

At any given time instant, the value of a stochastic process, whether it is the value of the noise voltage generated by a resistor or the amplitude of the signal generated by an audio source, is a random variable. Thus, we may view a stochastic process as a random variable indexed by the parameter t. We shall denote such a process by $X(t)$. In general, the parameter t is continuous, whereas X may be either continuous or discrete, depending on the characteristics of the source that generates the stochastic process.

The noise voltage generated by a single resistor or a single information source represents a single realization of the stochastic process. Hence, it is called a *sample function* of the stochastic process. The set of all possible sample functions, e.g., the set of all noise voltage waveforms generated by resistors, constitute an ensemble of sample functions or, equivalently, the stochastic process $X(t)$. In general, the number of sample functions in the ensemble is assumed to be extremely large; often it is infinite.

Having defined a stochastic process $X(t)$ as an ensemble of sample functions, we may consider the values of the process at any set of time instants $t_1 > t_2 > t_3 > \ldots > t_n$ where n is any positive integer. In general, the random variables $X_{t_i} \equiv X(t_i)$, $i = 1, 2, \ldots, n$, are characterized statistically by their joint pdf $p(x_{t_1}, x_{t_2}, \ldots, x_{t_n})$. Furthermore, all the probabilistic relations defined in Section 2-1 for multidimensional random variables carry over to the random variables X_{t_i}, $i = 1, 2, \ldots, n$.

Stationary Stochastic Processes As indicated above, the random variables X_{t_i}, $i = 1, 2, \ldots, n$, obtained from the stochastic process $X(t)$ for any set of time instants $t_1 > t_2 > t_3 > \ldots > t_n$ and any n are characterized statistically by the joint pdf $p(x_{t_1}, x_{t_2}, \ldots, x_{t_n})$. Let us consider another set of n random variables $X_{t_i+t} \equiv X(t_i + t)$, $i = 1, 2, \ldots, n$, where t is an arbitrary time shift. These random variables are characterized by the joint pdf $p(x_{t_1+t}, x_{t_2+t}, \ldots, x_{t_n+t})$. The joint pdfs of the random variables X_{t_i} and X_{t_i+t}, $i = 1, 2, \ldots, n$, may or may not be identical. When they are identical, i.e., when

$$p(x_{t_1}, x_{t_2}, \ldots, x_{t_n}) = p(x_{t_1+t}, x_{t_2+t}, \ldots, x_{t_n+t}) \tag{2-2-1}$$

for all t and all n, the stochastic process is said to be *stationary in the strict sense*. That is, the statistics of a stationary stochastic process are invariant to any translation of the time axis. On the other hand, when the joint pdfs are different, the stochastic process is *nonstationary*.

2-2-1 Statistical Averages

Just as we have defined statistical averages for random variables, we may similarly define statistical averages for a stochastic process. Such averages are also called *ensemble averages*. Let $X(t)$ denote a random process and let $X_{t_i} \equiv X(t_i)$. The *nth moment* of the random variable X_{t_i} is defined as

$$E(X_{t_i}^n) = \int_{-\infty}^{\infty} x_{t_i}^n p(x_{t_i})\, dx_{t_i} \tag{2-2-2}$$

In general, the value of the nth moment will depend on the time instant t_i if the pdf of X_{t_i} depends on t_i. When the process is stationary, however, $p(x_{t_i+t}) = p(x_{t_i})$ for all t. Hence, the pdf is independent of time, and, as a consequence, *the nth moment is independent of time.*

Next we consider the two random variables $X_{t_i} \equiv X(t_i)$, $i = 1, 2$. The correlation between X_{t_1} and X_{t_2} is measured by the joint moment

$$E(X_{t_1}X_{t_2}) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x_{t_1}x_{t_2} p(x_{t_1}, x_{t_2})\, dx_{t_1}\, dx_{t_2} \tag{2-2-3}$$

Since this joint moment depends on the time instants t_1 and t_2, it is denoted by $\phi(t_1, t_2)$. The function $\phi(t_1, t_2)$ is called the *autocorrelation function* of the stochastic process. When the process $X(t)$ is stationary, the joint pdf of the pair (X_{t_1}, X_{t_2}) is identical to the joint pdf of the pair (X_{t_1+t}, X_{t_2+t}) for any arbitrary t. This implies that the autocorrelation function of $X(t)$ does not depend on the specific time instants t_1 and t_2, but, instead, it depends on the time difference $t_1 - t_2$. Thus, for a stationary stochastic process, the joint moment in (2-2-3) is

$$E(X_{t_1}X_{t_2}) = \phi(t_1, t_2) = \phi(t_1 - t_2) = \phi(\tau) \tag{2-2-4}$$

where $\tau = t_1 - t_2$ or, equivalently, $t_2 = t_1 - \tau$. If we let $t_2 = t_1 + \tau$, we have

$$\phi(-\tau) = E(X_{t_1}X_{t_1+\tau}) = E(X_{t_1}X_{t_1-\tau}) = \phi(\tau)$$

Therefore, $\phi(\tau)$ is an even function. We also note that $\phi(0) = E(X_t^2)$ denotes the average power in the process $X(t)$.

There exist nonstationary processes with the property that the mean value of the process is independent of time (a constant) and where the autocorrelation function satisfies the condition that $\phi(t_1, t_2) = \phi(t_1 - t_2)$. Such a process is called *wide-sense stationary*. Consequently, wide-sense stationarity is a less stringent condition than strict-sense stationarity. When reference is made to a stationary stochastic process in any subsequent discussion in which correlation functions are involved, the less stringent condition (wide-sense stationarity) is implied.

Related to the autocorrelation function is the autocovariance function of a stochastic process, which is defined as

$$\begin{aligned}\mu(t_1, t_2) &= E\{[X_{t_1} - m(t_1)][X_{t_2} - m(t_2)]\} \\ &= \phi(t_1, t_2) - m(t_1)m(t_2)\end{aligned} \tag{2-2-5}$$

where $m(t_1)$ and $m(t_2)$ are the means of X_{t_1} and X_{t_2}, respectively. When the process is stationary, the autocovariance function simplifies to

$$\mu(t_1, t_2) = \mu(t_1 - t_2) = \mu(\tau) = \phi(\tau) - m^2 \tag{2-2-6}$$

where $\tau = t_1 - t_2$.

Higher-order joint moments of two or more random variables derived from a stochastic process $X(t)$ are defined in an obvious manner. With the possible exception of the gaussian random process, for which higher-order moments can be expressed in terms of first and second moments, high-order moments are encountered very infrequently in practice.

Averages for a Gaussian Process Suppose that $X(t)$ is a gaussian random process. Hence, at time instants $t = t_i$, $i = 1, 2, \ldots, n$, the random variables X_{t_i}, $i = 1, 2, \ldots, n$, are jointly gaussian with mean values $m(t_i)$, $i = 1, 2, \ldots, n$, and autocovariances

$$\mu(t_i, t_j) = E[(X_{t_i} - m(t_i))(X_{t_j} - m(t_j))], \quad i, j = 1, 2, \ldots, n \tag{2-2-7}$$

If we denote the $n \times n$ covariance matrix with elements $\mu(t_i, t_j)$ by $\mathbf{M}$ and the vector of mean values by $\mathbf{m}_x$, then the joint pdf of the random variables X_{t_i}, $i = 1, 2, \ldots, n$ is given by (2-1-150).

If the gaussian process is stationary then $m(t_i) = m$ for all t_i and $\mu(t_i, t_j) = \mu(t_i - t_j)$. We observe that the gaussian random process is completely specified by the mean and autocovariance functions. Since the joint gaussian pdf depends only on these two moments, it follows that if the gaussian process is wide-sense stationary, it is also strict-sense stationary. Of course, the converse is always true for any stochastic process.

Averages for Joint Stochastic Processes Let $X(t)$ and $Y(t)$ denote two stochastic processes and let $X_{t_i} \equiv X(t_i)$, $i = 1, 2, \ldots, n$, and $Y_{t'_j} \equiv Y(t'_j)$, $j = 1, 2, \ldots, m$, represent the random variables at times $t_1 > t_2 > t_3 > \ldots > t_n$ and $t'_1 > t'_2 > \ldots > t'_m$, respectively. The two processes are characterized statistically by their joint pdf

$$p(x_{t_1}, x_{t_2}, \ldots, x_{t_n}, y_{t'_1}, y_{t'_2}, \ldots, y_{t'_m})$$

for any set of time instants $t_1, t_2, \ldots, t_n$, $t'_1, t'_2, \ldots, t'_m$ and for any positive integer values of n and m.

The *cross-correlation function* of $X(t)$ and $Y(t)$, denoted by $\phi_{xy}(t_1, t_2)$, is defined as the joint moment

$$\phi_{xy}(t_1, t_2) = E(X_{t_1} Y_{t_2}) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x_{t_1} y_{t_2} p(x_{t_1}, y_{t_2})\, dx_{t_1}\, dy_{t_2} \tag{2-2-8}$$

and the *cross-covariance* is

$$\mu_{xy}(t_1, t_2) = \phi_{xy}(t_1, t_2) - m_x(t_1) m_y(t_2) \tag{2-2-9}$$

When the processes are jointly and individually stationary, we have $\phi_{xy}(t_1, t_2) = \phi_{xy}(t_1 - t_2)$ and $\mu_{xy}(t_1, t_2) = \mu_{xy}(t_1 - t_2)$. In this case, we note that

$$\phi_{xy}(-\tau) = E(X_{t_1} Y_{t_1+\tau}) = E(X_{t_1-\tau} Y_{t_1}) = \phi_{yx}(\tau) \tag{2-2-10}$$

The stochastic processes $X(t)$ and $Y(t)$ are said to be *statistically independent* if and only if

$$p(x_{t_1}, x_{t_2}, \ldots, x_{t_n}, y_{t_1'}, y_{t_2'}, \ldots, y_{t_m'}) = p(x_{t_1}, x_{t_2}, \ldots, x_{t_n}) p(y_{t_1'}, y_{t_2'}, \ldots, y_{t_m'})$$

for all choices of t_i and t_i' and for all positive integers n and m. The processes are said to be *uncorrelated* if

$$\phi_{xy}(t_1, t_2) = E(X_{t_1}) E(Y_{t_2})$$

Hence,

$$\mu_{xy}(t_1, t_2) = 0$$

A *complex-valued stochastic process* $Z(t)$ is defined as

$$Z(t) = X(t) + jY(t) \tag{2-2-11}$$

where $X(t)$ and $Y(t)$ are stochastic processes. The joint pdf of the random variables $Z_{t_i} \equiv Z(t_i)$, $i = 1, 2, \ldots$, is given by the joint pdf of the components (X_{t_i}, Y_{t_i}), $i = 1, 2, \ldots, n$. Thus, the pdf that characterizes Z_{t_i}, $i = 1, 2, \ldots, n$, is

$$p(x_{t_1}, x_{t_2}, \ldots, x_{t_n}, y_{t_1}, y_{t_2}, \ldots, y_{t_n})$$

The complex-valued stochastic process $Z(t)$ is encountered in the representation of narrowband bandpass noise in terms of its equivalent lowpass components. An important characteristic of such a process is its autocorrelation function. The function is defined as

$$\begin{aligned} \phi_{zz}(t_1, t_2) &= \tfrac{1}{2} E(Z_{t_1} Z_{t_2}^*) \\ &= \tfrac{1}{2} E[(X_{t_1} + jY_{t_1})(X_{t_2} - jY_{t_2})] \\ &= \tfrac{1}{2}\{\phi_{xx}(t_1, t_2) + \phi_{yy}(t_1, t_2) + j[\phi_{yx}(t_1, t_2) - \phi_{xy}(t_1, t_2)]\} \end{aligned} \tag{2-2-12}$$

where $\phi_{xx}(t_1, t_2)$ and $\phi_{yy}(t_1, t_2)$ are the autocorrelation functions of $X(t)$ and $Y(t)$, respectively, and $\phi_{yx}(t_1, t_2)$ and $\phi_{xy}(t_1, t_2)$ are the cross-correlation functions. The factor of $\frac{1}{2}$ in the definition of the autocorrelation function of a complex-valued stochastic process is an arbitrary but mathematically convenient normalization factor, as we will demonstrate in our treatment of such processes in Chapter 4.

When the processes $X(t)$ and $Y(t)$ are jointly and individually stationary, the autocorrelation function of $Z(t)$ becomes

$$\phi_{zz}(t_1, t_2) = \phi_{zz}(t_1 - t_2) = \phi_{zz}(\tau)$$

where $t_2 = t_1 - \tau$. Also, the complex conjugate of (2-2-12) is

$$\phi_{zz}^*(\tau) = \tfrac{1}{2} E(Z_{t_1}^* Z_{t_1-\tau}) = \tfrac{1}{2} E(Z_{t_1+\tau}^* Z_{t_1}) = \phi_{zz}(-\tau) \tag{2-2-13}$$

Hence, $\phi_{zz}(\tau) = \phi_{zz}^*(-\tau)$.

Now, suppose that $Z(t) = X(t) + jY(t)$ and $W(t) = U(t) + jV(t)$ are two complex-valued stochastic processes. The cross-correlation function of $Z(t)$ and $W(t)$ is defined as

$$\begin{aligned}\phi_{zw}(t_1, t_2) &= \tfrac{1}{2}E(Z_{t_1}W^*_{t_2}) \\ &= \tfrac{1}{2}E[(X_{t_1} + jY_{t_1})(U_{t_2} - jV_{t_2})] \\ &= \tfrac{1}{2}\{\phi_{xu}(t_1, t_2) + \phi_{yv}(t_1, t_2) + j[\phi_{yu}(t_1, t_2) - \phi_{xv}(t_1, t_2)]\} \quad (2\text{-}2\text{-}14)\end{aligned}$$

When $X(t)$, $Y(t)$, $U(t)$, and $V(t)$ are pairwise-stationary, the cross-correlation functions in (2-2-14) become functions of the time difference $\tau = t_1 - t_2$. Furthermore,

$$\phi^*_{zw}(\tau) = \tfrac{1}{2}E(Z^*_{t_1}W_{t_1-\tau}) = \tfrac{1}{2}E(Z^*_{t_1+\tau}W_{t_1}) = \phi_{wz}(-\tau) \quad (2\text{-}2\text{-}15)$$

2-2-2 Power Density Spectrum

The frequency content of a signal is a very basic characteristic that distinguishes one signal from another. In general, a signal can be classified as having either a finite (nonzero) average power (infinite energy) or finite energy. The frequency content of a finite energy signal is obtained as the Fourier transform of the corresponding time function. If the signal is periodic, its energy is infinite and, consequently, its Fourier transform does not exist. The mechanism for dealing with periodic signals is to represent them in a Fourier series. With such a representation, the Fourier coefficients determine the distribution of power at the various discrete frequency components.

A stationary stochastic process is an infinite energy signal, and, hence, its Fourier transform does not exist. The spectral characteristic of a stochastic signal is obtained by computing the Fourier transform of the autocorrelation function. That is, the distribution of power with frequency is given by the function

$$\Phi(f) = \int_{-\infty}^{\infty} \phi(\tau)e^{-j2\pi f\tau}\,d\tau \quad (2\text{-}2\text{-}16)$$

The inverse Fourier transform relationship is

$$\phi(\tau) = \int_{-\infty}^{\infty} \Phi(f)e^{j2\pi f\tau}\,df \quad (2\text{-}2\text{-}17)$$

We observe that

$$\begin{aligned}\phi(0) &= \int_{-\infty}^{\infty} \Phi(f)\,df \\ &= E(|X_i|^2) \geq 0 \quad (2\text{-}2\text{-}18)\end{aligned}$$

Since $\phi(0)$ represents the average power of the stochastic signal, which is the area under $\Phi(f)$, $\Phi(f)$ is the distribution of power as a function of frequency. Therefore, $\Phi(f)$ is called the *power density spectrum* of the stochastic process.

If the stochastic process is real, $\phi(\tau)$ is real and even, and, hence $\Phi(f)$ is real and even. On the other hand, if the process is complex, $\phi(\tau) = \phi^*(-\tau)$ and, hence

$$\Phi^*(f) = \int_{-\infty}^{\infty} \phi^*(\tau) e^{j2\pi f\tau}\, d\tau = \int_{-\infty}^{\infty} \phi^*(-\tau) e^{-j2\pi f\tau}\, d\tau$$

$$= \int_{-\infty}^{\infty} \phi(\tau) e^{-j2\pi f\tau}\, d\tau = \Phi(f) \tag{2-2-19}$$

Therefore, $\Phi(f)$ is real.

The definition of a power density spectrum can be extended to two jointly stationary stochastic processes $X(t)$ and $Y(t)$, which have a cross-correlation function $\phi_{xy}(\tau)$. The Fourier transform of $\phi_{xy}(\tau)$, i.e.,

$$\Phi_{xy}(f) = \int_{-\infty}^{\infty} \phi_{xy}(\tau) e^{-j2\pi f\tau}\, d\tau \tag{2-2-20}$$

is called the *cross-power density spectrum*. If we conjugate both sides of (2-2-20), we have

$$\Phi_{xy}^*(f) = \int_{-\infty}^{\infty} \phi_{xy}^*(\tau) e^{j2\pi f\tau}\, d\tau = \int_{-\infty}^{\infty} \phi_{xy}^*(-\tau) e^{-j2\pi f\tau}\, d\tau$$

$$= \int_{-\infty}^{\infty} \phi_{yx}(\tau) e^{-j2\pi f\tau}\, d\tau = \Phi_{yx}(f) \tag{2-2-21}$$

This relation holds in general. However, if $X(t)$ and $Y(t)$ are real stochastic processes,

$$\Phi_{xy}^*(f) = \int_{-\infty}^{\infty} \phi_{xy}(\tau) e^{j2\pi f\tau}\, d\tau = \Phi_{xy}(-f) \tag{2-2-22}$$

By combining the result in (2-2-21) with the result in (2-2-22), we find that the cross-power density spectrum of two real processes satisfies the condition

$$\Phi_{yx}(f) = \Phi_{xy}(-f) \tag{2-2-23}$$

2-2-3 Response of a Linear Time-Invariant System to a Random Input Signal

Consider a linear time-invariant system (filter) that is characterized by its impulse response $h(t)$ or, equivalently, by its frequency response $H(f)$, where $h(t)$ and $H(f)$ are a Fourier transform pair. Let $x(t)$ be the input signal to the system and let $y(t)$ denote the output signal. The output of the system may be expressed in terms of the convolution integral as

$$y(t) = \int_{-\infty}^{\infty} h(\tau) x(t-\tau)\, d\tau \tag{2-2-24}$$

Now, suppose that $x(t)$ is a sample function of a stationary stochastic process $X(t)$. Then, the output $y(t)$ is a sample function of a stochastic process $Y(t)$. We wish to determine the mean and autocorrelation functions of the output.

Since convolution is a linear operation performed on the input signal $x(t)$, the expected value of the integral is equal to the integral of the expected value. Thus, the mean value of $Y(t)$ is

$$m_y = E[Y(t)] = \int_{-\infty}^{\infty} h(\tau)E[X(t-\tau)]\,d\tau$$

$$= m_x \int_{-\infty}^{\infty} h(\tau)\,d\tau = m_x H(0) \tag{2-2-25}$$

where $H(0)$ is the frequency response of the linear system at $f = 0$. Hence, the mean value of the output process is a constant.

The autocorrelation function of the output is

$$\phi_{yy}(t_1, t_2) = \tfrac{1}{2}E(Y_{t_1}Y_{t_2}^*)$$

$$= \frac{1}{2}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h(\beta)h^*(\alpha)E[X(t_1-\beta)X^*(t_2-\alpha)]\,d\alpha\,d\beta$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h(\beta)h^*(\alpha)\phi_{xx}(t_1 - t_2 + \alpha - \beta)\,d\alpha\,d\beta$$

The last step indicates that the double integral is a function of the time difference $t_1 - t_2$. In other words, if the input process is stationary, the output is also stationary. Hence

$$\phi_{yy}(\tau) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h^*(\alpha)h(\beta)\phi_{xx}(\tau + \alpha - \beta)\,d\alpha\,d\beta \tag{2-2-26}$$

By evaluating the Fourier transform of both sides of (2-2-26), we obtain the power density spectrum of the output process in the form

$$\Phi_{yy}(f) = \int_{-\infty}^{\infty} \phi_{yy}(\tau)e^{-j2\pi f\tau}\,d\tau$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h^*(\alpha)h(\beta)\phi_{xx}(\tau + \alpha - \beta)e^{-j2\pi f\tau}\,d\tau\,d\alpha\,d\beta$$

$$= \Phi_{xx}(f)\,|H(f)|^2 \tag{2-2-27}$$

Thus, we have the important result that the power density spectrum of the output signal is the product of the power density spectrum of the input multiplied by the magnitude squared of the frequency response of the system.

When the autocorrelation function $\phi_{yy}(\tau)$ is desired, it is usually easier to determine the power density spectrum $\Phi_{yy}(f)$ and then to compute the inverse transform. Thus, we have

$$\phi_{yy}(\tau) = \int_{-\infty}^{\infty} \Phi_{yy}(f) e^{j2\pi f\tau}\, df$$

$$= \int_{-\infty}^{\infty} \Phi_{xx}(f)\, |H(f)|^2\, e^{j2\pi f\tau}\, df \tag{2-2-28}$$

We observe that the average power in the output signal is

$$\phi_{yy}(0) = \int_{-\infty}^{\infty} \Phi_{xx}(f)\, |H(f)|^2\, df \tag{2-2-29}$$

Since $\phi_{yy}(0) = E(|Y_t|^2)$, it follows that

$$\int_{-\infty}^{\infty} \Phi_{xx}(f)\, |H(f)|^2\, df \geq 0$$

Suppose we let $|H(f)|^2 = 1$ for any arbitrarily small interval $f_1 \leq f \leq f_2$, and $H(f) = 0$ outside this interval. Then,

$$\int_{f_1}^{f_2} \Phi_{xx}(f)\, df \geq 0$$

But this is possible if and only if $\Phi_{xx}(f) \geq 0$ for all f.

Example 2-2-1

Suppose that the lowpass filter illustrated in Fig. 2-2-1 is excited by a stochastic process $x(t)$ having a power density spectrum

$$\Phi_{xx}(f) = \tfrac{1}{2}N_0 \quad \text{for all } f$$

A stochastic process having a flat power density spectrum is called *white*

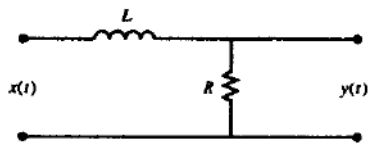


FIGURE 2-2-1 An example of a lowpass filter.

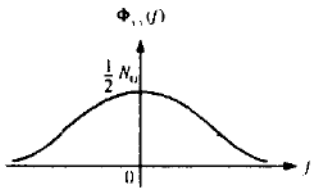


FIGURE 2-2-2 The power density spectrum of the lowpass filter output when the input is white noise.

noise. Let us determine the power density spectrum of the output process. The transfer function of the lowpass filter is

$$H(f) = \frac{R}{R + j2\pi fL} = \frac{1}{1 + j2\pi fL/R}$$

and, hence,

$$|H(f)|^2 = \frac{1}{1 + (2\pi L/R)^2 f^2} \tag{2-2-30}$$

The power density spectrum of the output process is

$$\Phi_{yy}(f) = \frac{N_0}{2} \frac{1}{1 + (2\pi L/R)^2 f^2} \tag{2-2-31}$$

This power density spectrum is illustrated in Fig. 2-2-2. Its inverse Fourier transform yields the autocorrelation function

$$\begin{aligned} \phi_{yy}(\tau) &= \int_{-\infty}^{\infty} \frac{N_0}{2} \frac{1}{1 + (2\pi L/R)^2 f^2} e^{j2\pi f\tau} df \\ &= \frac{RN_0}{4L} e^{-(R/L)|\tau|} \end{aligned} \tag{2-2-32}$$

The autocorrelation function $\phi_{yy}(\tau)$ is shown in Fig. 2-2-3. We observe that the second moment of the process $Y(t)$ is $\phi_{yy}(0) = RN_0/4L$.

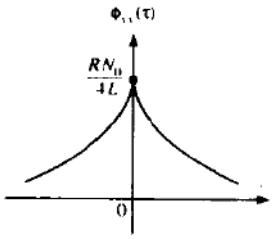


FIGURE 2-2-3 The autocorrelation function of the output of the lowpass filter for a white-noise input.

As a final exercise, we determine the cross-correlation function between $y(t)$ and $x(t)$, where $x(t)$ denotes the input and $y(t)$ denotes the output of the linear system. We have

$$\phi_{yx}(t_1, t_2) = \tfrac{1}{2}E(Y_{t_1}X^*_{t_2}) = \frac{1}{2}\int_{-\infty}^{\infty} h(\alpha)E[X(t_1 - \alpha)X^*(t_2)]\,d\alpha$$

$$= \int_{-\infty}^{\infty} h(\alpha)\phi_{xx}(t_1 - t_2 - \alpha)\,d\alpha = \phi_{yx}(t_1 - t_2)$$

Hence, the stochastic processes $X(t)$ and $Y(t)$ are jointly stationary. With $t_1 - t_2 = \tau$, we have

$$\phi_{yx}(\tau) = \int_{-\infty}^{\infty} h(\alpha)\phi_{xx}(\tau - \alpha)\,d\alpha \tag{2-2-33}$$

Note that the integral in (2-2-33) is a convolution integral. Hence in the frequency domain the relation (2-2-33) becomes

$$\Phi_{yx}(f) = \Phi_{xx}(f)H(f) \tag{2-2-34}$$

We observe that if the input process is white noise, the cross correlation of the input with the output of the system yields the impulse response $h(t)$ to within a scale factor.

2-2-4 Sampling Theorem for Band-Limited Stochastic Processes

Recall that a deterministic signal $s(t)$ that has a Fourier transform $S(f)$ is called band-limited if $S(f) = 0$ for $|f| > W$, where W is the highest frequency contained in $s(t)$. Such a signal is uniquely represented by samples of $s(t)$ taken at a rate of $f_s \geq 2W$ samples/s. The minimum rate $f_N = 2W$ samples/s is called the *Nyquist rate*. Sampling below the Nyquist rate results in frequency aliasing.

The band-limited signal sampled at the Nyquist rate can be reconstructed from its samples by use of the interpolation formula

$$s(t) = \sum_{n=-\infty}^{\infty} s\left(\frac{n}{2W}\right)\frac{\sin\left[2\pi W\left(t - \frac{n}{2W}\right)\right]}{2\pi W\left(t - \frac{n}{2W}\right)} \tag{2-2-35}$$

where $\{s(n/2W)\}$ are the samples of $s(t)$ taken at $t = n/2W$, $n = 0, \pm 1, \pm 2, \ldots$. Equivalently, $s(t)$ can be reconstructed by passing the sampled signal through

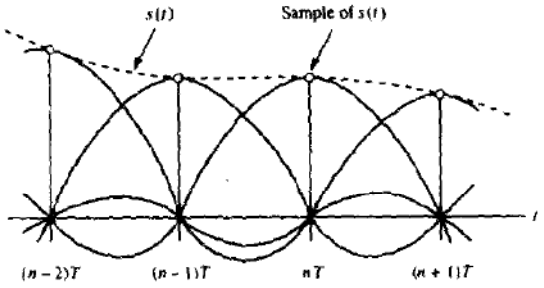


FIGURE 2-2-4 Signal reconstruction based on ideal interpolation.

an ideal low-pass filter with impulse response $h(t) = (\sin 2\pi Wt)/2\pi Wt$. Figure 2-2-4 illustrates the signal reconstruction process based on ideal interpolation.

A stationary stochastic process $X(t)$ is said to be *band-limited* if its power density spectrum $\Phi(f) = 0$ for $|f| > W$. Since $\Phi(f)$ is the Fourier transform of the autocorrelation function $\phi(\tau)$, it follows that $\phi(\tau)$ can be represented as

$$\phi(\tau) = \sum_{n=-\infty}^{\infty} \phi\left(\frac{n}{2W}\right) \frac{\sin\left[2\pi W\left(\tau - \frac{n}{2W}\right)\right]}{2\pi W\left(\tau - \frac{n}{2W}\right)} \tag{2-2-36}$$

where $\{\phi(n/2W)\}$ are samples of $\phi(\tau)$ taken at $\tau = n/2W$, $n = 0, \pm 1, \pm 2, \ldots$.

Now, if $X(t)$ is a band-limited stationary stochastic process then $X(t)$ can be represented as

$$X(t) = \sum_{n=-\infty}^{\infty} X\left(\frac{n}{2W}\right) \frac{\sin\left[2\pi W\left(t - \frac{n}{2W}\right)\right]}{2\pi W\left(t - \frac{n}{2W}\right)} \tag{2-2-37}$$

where $\{X(n/2W)\}$ are samples of $X(t)$ taken at $t = n/2W$, $n = 0, \pm 1, \pm 2, \ldots$. This is the sampling representation for a stationary stochastic process. The samples are random variables that are described statistically by appropriate joint probability density functions. The signal representation in (2-2-37) is easily established by showing that (Problem 2-17)

$$E\left\{\left|X(t) - \sum_{n=-\infty}^{\infty} X\left(\frac{n}{2W}\right) \frac{\sin\left[2\pi W\left(t - \frac{n}{2W}\right)\right]}{2\pi W\left(t - \frac{n}{2W}\right)}\right|^2\right\} = 0 \tag{2-2-38}$$

Hence, equality between the sampling representation and the stochastic process $X(t)$ holds in the sense that the mean square error is zero.

2-2-5 Discrete-Time Stochastic Signals and Systems

The characterization of continuous-time stochastic signals given above can be easily carried over to discrete-time stochastic signals. Such signals are usually obtained by uniformly sampling a continuous-time stochastic process.

A discrete-time stochastic process $X(n)$ consists of an ensemble of sample sequences $\{x(n)\}$. The statistical properties of $X(n)$ are similar to the characterization of $X(t)$ with the restriction that n is now an integer (time) variable. Hence, the *mth moment* of $X(n)$ is defined as

$$E[X_n^m] = \int_{-\infty}^{\infty} X_n^m p(X_n)\, dX_n \tag{2-2-39}$$

and the *autocorrelation sequence* is

$$\phi(n, k) = \tfrac{1}{2}E(X_n X_k^*) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} X_n X_k^* p(X_n, X_k)\, dX_n\, dX_k \tag{2-2-40}$$

Similarly, the autocovariance sequence is

$$\mu(n, k) = \phi(n, k) - E(X_n)E(X_k^*) \tag{2-2-41}$$

For a stationary process, we have $\phi(n, k) \equiv \phi(n-k)$, $\mu(n, k) \equiv \mu(n-k)$, and

$$\mu(n-k) = \phi(n-k) - |m_x|^2 \tag{2-2-42}$$

where $m_x = E(X_n)$ is the mean value.

As in the case of continuous-time stochastic processes, a discrete-time stationary process has infinite energy but a finite average power, which is given as

$$E(|X_n|^2) = \phi(0) \tag{2-2-43}$$

The power density spectrum for the discrete-time process is obtained by computing the Fourier transform of $\phi(n)$. Since $\phi(n)$ is a discrete-time sequence, the Fourier transform is defined as

$$\Phi(f) = \sum_{n=-\infty}^{\infty} \phi(n) e^{-j2\pi fn} \tag{2-2-44}$$

and the inverse transform relationship is

$$\phi(n) = \int_{-1/2}^{1/2} \Phi(f) e^{j2\pi fn}\, df \tag{2-2-45}$$

We make the observation that the power density spectrum $\Phi(f)$ is periodic with a period $f_p = 1$. In other words, $\Phi(f + k) = \Phi(f)$ for $k = \pm 1, \pm 2, \ldots$. This is a characteristic of the Fourier transform of any discrete-time sequence such as $\phi(n)$.

Finally, let us consider the response of a discrete-time, linear time-invariant system to a stationary stochastic input signal. The system is characterized in

the time domain by its unit sample response $h(n)$ and in the frequency domain by the frequency response $H(f)$, where

$$H(f) = \sum_{n=-\infty}^{\infty} h(n)e^{-j2\pi fn} \tag{2-2-46}$$

The response of the system to the stationary stochastic input signal $X(n)$ is given by the convolution sum

$$y(n) = \sum_{k=-\infty}^{\infty} h(k)x(n-k) \tag{2-2-47}$$

The mean value of the output of the system is

$$\begin{aligned} m_y &= E[y(n)] = \sum_{k=-\infty}^{\infty} h(k)E[x(n-k)] \\ m_y &= m_x \sum_{k=-\infty}^{\infty} h(k) = m_x H(0) \end{aligned} \tag{2-2-48}$$

where $H(0)$ is the zero frequency (dc) gain of the system.

The autocorrelation sequence for the output process is

$$\begin{aligned} \phi_{yy}(k) &= \tfrac{1}{2}E[y^*(n)y(n+k)] \\ &= \tfrac{1}{2}\sum_{i=-\infty}^{\infty}\sum_{j=-\infty}^{\infty} h^*(i)h(j)E[x^*(n-i)x(n+k-j)] \\ &= \sum_{i=-\infty}^{\infty}\sum_{j=-\infty}^{\infty} h^*(i)h(j)\phi_{xx}(k-j+i) \end{aligned} \tag{2-2-49}$$

This is the general form for the autocorrelation sequence of the system output in terms of the autocorrelation of the system input and the unit sample response of the system. By taking the Fourier transform of $\phi_{yy}(k)$ and substituting the relation in (2-2-49), we obtain the corresponding frequency domain relationship

$$\Phi_{yy}(f) = \Phi_{xx}(f)\,|H(f)|^2 \tag{2-2-50}$$

which is identical to (2-2-27) except that in (2-2-50) the power density spectra $\Phi_{yy}(f)$ and $\Phi_{xx}(f)$ and the frequency response $H(f)$ are periodic functions of frequency with period $f_p = 1$.

2-2-6 Cyclostationary Processes

In dealing with signals that carry digital information we encounter stochastic processes that have statistical averages that are periodic. To be specific, let us consider a stochastic process of the form

$$X(t) = \sum_{n=-\infty}^{\infty} a_n g(t - nT) \tag{2-2-51}$$

where $\{a_n\}$ is a (discrete-time) sequence of random variables with mean $m_a = E(a_n)$ for all n and autocorrelation sequence $\phi_{aa}(k) = \frac{1}{2}E(a_n^* a_{n+k})$. The signal $g(t)$ is deterministic. The stochastic process $X(t)$ represents the signal for several different types of linear modulation techniques which are introduced in Chapter 4. The sequence $\{a_n\}$ represents the digital information sequence (of symbols) that is transmitted over the communication channel and $1/T$ represents the rate of transmission of the information symbols.

Let us determine the mean and autocorrelation function of $X(t)$. First, the mean value is

$$E[X(t)] = \sum_{n=-\infty}^{\infty} E(a_n)g(t - nT)$$

$$= m_a \sum_{n=-\infty}^{\infty} g(t - nT) \tag{2-5-52}$$

We observe that the mean is time-varying. In fact, it is periodic with period T.

The autocorrelation function of $X(t)$ is

$$\phi_{xx}(t + \tau, t) = \tfrac{1}{2}E[X(t + \tau)X^*(t)]$$

$$= \tfrac{1}{2} \sum_{n=-\infty}^{\infty} \sum_{m=-\infty}^{\infty} E(a_n^* a_m)g^*(t - nT)g(t + \tau - mT)$$

$$= \sum_{n=-\infty}^{\infty} \sum_{m=-\infty}^{\infty} \phi_{aa}(m - n)g^*(t - nT)g(t + \tau - mT) \tag{2-2-53}$$

Again, we observe that

$$\phi_{xx}(t + \tau + kT, t + kT) = \phi_{xx}(t + \tau, t) \tag{2-2-54}$$

for $k = \pm 1, \pm 2, \ldots$. Hence, the autocorrelation function of $X(t)$ is also periodic with period T.

Such a stochastic process is called *cyclostationary* or *periodically stationary*. Since the autocorrelation function depends on both the variables t and τ, its frequency domain representation requires the use of a two-dimensional Fourier transform.

Since it is highly desirable to characterize such signals by their power density spectrum, an alternative approach is to compute the *time-average autocorrelation function* over a single period, defined as

$$\bar{\phi}_{xx}(\tau) = \frac{1}{T} \int_{-T/2}^{T/2} \phi_{xx}(t + \tau, t)\, dt \tag{2-2-55}$$

Thus, we eliminate the time dependence by dealing with the average autocorrelation function. Now, the fourier transform of $\bar{\phi}_{xx}(\tau)$ yields the

average power density spectrum of the cyclostationary stochastic process. This approach allows us to simply characterize cyclostationary processes in the frequency domain in terms of the power spectrum. That is, the power density spectrum is

$$\Phi_{xx}(f) = \int_{-\infty}^{\infty} \bar{\phi}_{xx}(\tau) e^{-j2\pi f\tau}\, d\tau \tag{2-2-56}$$

2-3 BIBLIOGRAPHICAL NOTES AND REFERENCES

In this chapter we have provided a review of basic concepts and definitions in the theory of probability and stochastic processes. As stated in the opening paragraph, this theory is an important mathematical tool in the statistical modeling of information sources, communication channels, and in the design of digital communication systems. Of particular importance in the evaluation of communication system performance is the Chernoff bound. This bound is frequently used in bounding the probability of error of digital communication systems that employ coding in the transmission of information. Our coverage also highlighted a number of probability distributions and their properties, which are frequently encountered in the design of digital communication systems.

The texts by Davenport and Root (1958), Davenport (1970), Papoulis (1984) Pebbles (1987), Helstrom (1991) and Leon-Garcia (1994) provide engineering-oriented treatments of probability and stochastic processes. A more mathematical treatment of probability theory may be found in the text by Loève (1955). Finally, we cite the book by Miller (1964), which treats multidimensional gaussian distributions.

PROBLEMS

2-1 One experiment has four mutually exclusive outcomes A_i, $i = 1, 2, 3, 4$, and a second experiment has three mutually exclusive outcomes B_j, $j = 1, 2, 3$. The joint probabilities $P(A_i, B_j)$ are

$$\begin{aligned}
&P(A_1, B_1) = 0.10, \quad P(A_1, B_2) = 0.08, \quad P(A_1, B_3) = 0.13\\
&P(A_2, B_1) = 0.05, \quad P(A_2, B_2) = 0.03, \quad P(A_2, B_3) = 0.09\\
&P(A_3, B_1) = 0.05, \quad P(A_3, B_2) = 0.12, \quad P(A_3, B_3) = 0.14\\
&P(A_4, B_1) = 0.11, \quad P(A_4, B_2) = 0.04, \quad P(A_4, B_3) = 0.06
\end{aligned}$$

Determine the probabilities $P(A_i)$, $i = 1, 2, 3, 4$, and $P(B_j)$, $j = 1, 2, 3$.

2-2 The random variables X_i, $i = 1, 2, \ldots, n$, have the joint pdf $p(x_1, x_2, \ldots, x_n)$. Prove that

$$\begin{aligned}
&p(x_1, x_2, x_3, \ldots, x_n)\\
&\quad = p(x_n \mid x_{n-1}, \ldots, x_1) p(x_{n-1} \mid x_{n-2}, \ldots, x_1) \cdots p(x_3 \mid x_2, x_1) p(x_2 \mid x_1) p(x_1)
\end{aligned}$$

2-3 The pdf of a random variable X is $p(x)$. A random variable Y is defined as

$$Y = aX + b$$

where $a < 0$. Determine the pdf of Y in terms of the pdf of X.

2-4 Suppose that X is a gaussian random variable with zero mean and unit variance. Let

$$Y = aX^3 + b, \quad a > 0$$

Determine and plot the pdf of Y.

2-5 **a** Let X_r and X_i be statistically independent zero-mean gaussian random variables with identical variance. Show that a (rotational) transformation of the form

$$Y_r + jY_i = (X_r + jX_i)e^{j\phi}$$

results in another pair (Y_r, Y_i) of gaussian random variables that have the same joint pdf as the pair (X_r, X_i).

b Note that

$$\begin{bmatrix} Y_r \\ Y_i \end{bmatrix} = \mathbf{A}\begin{bmatrix} X_r \\ X_i \end{bmatrix}$$

where $\mathbf{A}$ is a 2×2 matrix. As a generalization of the two-dimensional transformation of the gaussian random variables considered in (a), what property must the linear transformation $\mathbf{A}$ satisfy if the pdfs for $\mathbf{X}$ and $\mathbf{Y}$, where $\mathbf{Y} = \mathbf{AX}$, $\mathbf{X} = (X_1 X_2 \cdots X_n)$ and $\mathbf{Y} = (Y_1 Y_2 \cdots Y_n)$, are identical?

2-6 The random variable Y is defined as

$$Y = \sum_{i=1}^{n} X_i$$

where the X_i, $i = 1, 2, \ldots, n$, are statistically independent random variables with

$$X_i = \begin{cases} 1 & \text{with probability } p \\ 0 & \text{with probability } 1 - p \end{cases}$$

a Determine the characteristic function of Y.

b From the characteristic function, determine the moments $E(Y)$ and $E(Y^2)$.

2-7 The four random variables X_1, X_2, X_3, X_4 are zero-mean jointly gaussian random variables with covariance $\mu_{ij} = E(X_i X_j)$ and characteristic function $\psi(jv_1, jv_2, jv_3, jv_4)$. Show that

$$E(X_1 X_2 X_3 X_4) = \mu_{12}\mu_{34} + \mu_{13}\mu_{24} + \mu_{14}\mu_{23}$$

2-8 From the characteristic functions for the central chi-square and noncentral chi-square random variables given by (2-1-109) and (2-1-117), respectively,

determine the corresponding first and second moments given by (2-1-112) and (2-1-125)

2-9 The pdf of a Cauchy distributed random variable X is

$$p(x) = \frac{a/\pi}{x^2 + a^2}, \qquad -\infty < x < \infty$$

a Determine the mean and variance of X.
b Determine the characteristic function of X.

2-10 The random variable Y is defined as

$$Y = \frac{1}{n}\sum_{i=1}^{n} X_i$$

where X_i, $i = 1, 2, \ldots, n$, are statistically independent and identically distributed random variables each of which has the Cauchy pdf given in Problem 2-9
a Determine the characteristic function of Y.
b Determine the pdf of Y.
c Consider the pdf of Y in the limit as $n \to \infty$. Does the central limit hold? Explain your answer.

2-11 Assume that random processes $x(t)$ and $y(t)$ are individually and jointly stationary.
a Determine the autocorrelation function of $z(t) = x(t) + y(t)$.
b Determine the autocorrelation function of $z(t)$ when $x(t)$ and $y(t)$ are uncorrelated.
c Determine the autocorrelation function of $z(t)$ when $x(t)$ and $y(t)$ are uncorrelated and have zero means.

2-12 The autocorrelation function of a stochastic process $X(t)$ is

$$\phi_{xx}(\tau) = \tfrac{1}{2}N_0\delta(\tau)$$

Such a process is called *white noise*. Suppose $x(t)$ is the input to an ideal bandpass filter having the frequency response characteristic shown in Fig. P2-12. Determine the total noise power at the output of the filter.

2-13 The covariance matrix of three random variables X_1, X_2 and X_3 is

$$\begin{bmatrix} \mu_{11} & 0 & \mu_{13} \\ 0 & \mu_{22} & 0 \\ \mu_{31} & 0 & \mu_{33} \end{bmatrix}$$

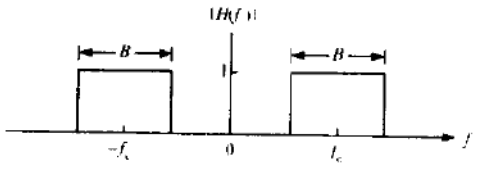


FIGURE P.2-12

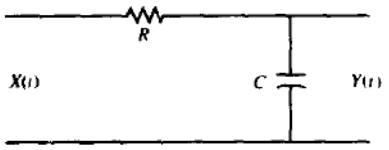


FIGURE P2-16

The linear transformation $\mathbf{Y} = \mathbf{AX}$ is made where

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 1 & 0 & 1 \end{bmatrix}$$

Determine the covariance matrix of $\mathbf{Y}$.

2-14 Let $X(t)$ be a stationary real normal process with zero mean. Let a new process $Y(t)$ be defined by

$$Y(t) = X^2(t)$$

Determine the autocorrelation function of $Y(t)$ in terms of the autocorrelation function of $X(t)$. *Hint*: Use the result on gaussian variables derived in Problem 2-7.

2-15 For the Nakagami pdf, given by (2-1-147), define the normalized random variable $X = R/\sqrt{\Omega}$. Determine the pdf of X.

2-16 The input $X(t)$ in the circuit shown in Fig. P2-16 is a stochastic process with $E[X(t)] = 0$ and $\phi_{xx}(\tau) = \sigma^2\delta(\tau)$, i.e., $X(t)$ is a white noise process.

a Determine the spectral density $\Phi_{yy}(f)$.

b Determine $\phi_{yy}(\tau)$ and $E[Y^2(t)]$.

2-17 Demonstrate the validity of (2-2-38).

2-18 Use the Chernoff bound to show that $Q(x) \leq e^{-x^2/2}$ where $Q(x)$ is defined by (2-1-97).

2-19 Determine the mean, the autocorrelation sequence, and the power density spectrum of the output of a system with unit sample response

$$h(n) = \begin{cases} 1 & (n = 0) \\ -2 & (n = 1) \\ 1 & (n = 2) \\ 0 & (\text{otherwise}) \end{cases}$$

when the input $x(n)$ is a white-noise process with variance σ_x^2.

2-20 The autocorrelation sequence of a discrete-time stochastic process is $\phi(k) = (\frac{1}{2})^{|k|}$. Determine its power density spectrum.

2-21 A discrete-time stochastic process $X(n) \equiv X(nT)$ is obtained by periodic sampling of a continuous-time zero-mean stationary process $X(t)$ where T is the sampling interval, i.e., $f_s = 1/T$ is the sampling rate.

a Determine the relationship between the autocorrelation function of $X(t)$ and the autocorrelation sequence of $X(n)$.

b Express the power density spectrum of $X(n)$ in terms of the power density spectrum of the process $X(t)$.

c Determine the conditions under which the power density spectrum of $X(n)$ is equal to the power density spectrum of $X(t)$.

2-22 Consider a band-limited zero-mean stationary stochastic $X(t)$ with power density spectrum

$$\Phi(f) = \begin{cases} 1 & (|f| \leq W) \\ 0 & (|f| > W) \end{cases}$$

$X(t)$ is sampled at a rate $f_s = 1/T$ to yield a discrete-time process $X(n) = X(nT)$.

a Determine the expression for the autocorrelation sequence of $X(n)$.

b Determine the minimum value of T that results in a white (spectrally flat) sequence.

c Repeat (b) if the power density spectrum of $X(t)$ is

$$\Phi(f) = \begin{cases} 1 - |f|/W & (|f| \leq W) \\ 0 & (|f| > W) \end{cases}$$

2-23 Show that the functions

$$f_k(t) = \frac{\sin\left[2\pi W\left(t - \frac{k}{2W}\right)\right]}{2\pi W\left(t - \frac{k}{2W}\right)}, \quad k = 0, \pm 1, \pm 2, \ldots$$

are orthogonal over the interval $[-\infty, \infty]$, i.e.,

$$\int_{-\infty}^{\infty} f_k(t) f_j(t)\, dt = \begin{cases} 1/2W & (k = j) \\ 0 & (k \neq j) \end{cases}$$

Therefore, the sampling theorem reconstruction formula may be viewed as a series expansion of the band-limited signal $s(t)$, where the weights are samples of $s(t)$ and the $\{f_k(t)\}$ are the set of orthogonal functions used in the series expansion.

2-24 The noise equivalent bandwidth of a system is defined as

$$B_{eq} = \frac{1}{G} \int_0^{\infty} |H(f)|^2\, df$$

where $G = \max |H(f)|^2$. Using this definition, determine the noise equivalent bandwidth of the ideal bandpass filter shown in Fig. P2-12 and the lowpass system shown in Fig. P2-16.

3

SOURCE CODING

Communication systems are designed to transmit the information generated by a source to some destination. Information sources may take a variety of different forms. For example, in radio broadcasting, the source is generally an audio source (voice or music). In TV broadcasting, the information source is a video source whose output is a moving image. The outputs of these sources are analog signals and, hence, the sources are called *analog sources*. In contrast, computers and storage devices, such as magnetic or optical disks, produce discrete outputs (usually binary or ASCII characters) and, hence, they are called *discrete sources*.

Whether a source is analog or discrete, a digital communication system is designed to transmit information in digital form. Consequently, the output of the source must be converted to a format that can be transmitted digitally. This conversion of the source output to a digital form is generally performed by the source encoder, whose output may be assumed to be a sequence of binary digits.

In this chapter, we treat source encoding based on mathematical models of information sources and a quantitative measure of the information emitted by a source. We consider the encoding of discrete sources first and then we discuss the encoding of analog sources. We begin by developing mathematical models for information sources.

3-1 MATHEMATICAL MODELS FOR INFORMATION SOURCES

Any information source produces an output that is random, i.e., the source output is characterized in statistical terms. Otherwise, if the source output

were known exactly, there would be no need to transmit it. In this section, we consider both discrete and analog information sources, and we postulate mathematical models for each type of source.

The simplest type of discrete source is one that emits a sequence of letters selected from a finite alphabet. For example, a *binary source* emits a binary sequence of the form 100101110..., where the alphabet consists of the two letters {0, 1}. More generally, a discrete information source with an alphabet of L possible letters, say $\{x_1, x_2, \ldots, x_L\}$, emits a sequence of letters selected from the alphabet.

To construct a mathematical model for a discrete source, we assume that each letter in the alphabet $\{x_1, x_2, \ldots, x_L\}$ has a given probability p_k of occurrence. That is,

$$p_k = P(X = x_k), \quad 1 \leq k \leq L$$

where

$$\sum_{k=1}^{L} p_k = 1$$

We consider two mathematical models of discrete sources. In the first, we assume that the output sequence from the source is statistically independent. That is, the current output letter is statistically independent from all past and future outputs. A source whose output satisfies the condition of statistical independence among output letters in the sequence is said to be *memoryless*. Such a source is called a *discrete memoryless source* (DMS).

If the discrete source output is statistically dependent, as, for example, English text, we may construct a mathematical model based on statistical stationarity. By definition, a discrete source is said to be *stationary* if the joint probabilities of two sequences of length n, say $a_1, a_2, \ldots, a_n$ and $a_{1+m}, a_{2+m}, \ldots, a_{n+m}$ are identical for all $n \geq 1$ and for all shifts m. In other words, the joint probabilities for any arbitrary length sequence of source outputs are invariant under a shift in the time origin.

An *analog* source has an output waveform $x(t)$ that is a sample function of a stochastic process $X(t)$. We assume that $X(t)$ is a stationary stochastic process with autocorrelation function $\phi_{xx}(\tau)$ and power spectral density $\Phi_{xx}(f)$. When $X(t)$ is a bandlimited stochastic process, i.e., $\Phi_{xx}(f) = 0$ for $|f| \geq W$, the sampling theorem may be used to represent $X(t)$ as

$$X(t) = \sum_{n=-\infty}^{\infty} X\left(\frac{n}{2W}\right) \frac{\sin\left[2\pi W\left(t - \frac{n}{2W}\right)\right]}{2\pi W\left(t - \frac{n}{2W}\right)} \tag{3-1-1}$$

where $\{X(n/2W)\}$ denote the samples of the process $X(t)$ taken at the sampling (Nyquist) rate of $f_s = 2W$ samples/s. Thus, by applying the sampling theorem, we may convert the output of an analog source into an equivalent

discrete-time source. Then, the source output is characterized statistically by the joint pdf $p(x_1, x_2, \ldots, x_m)$ for all $m \geq 1$, where $X_n = X(n/2W)$, $1 \leq n \leq m$, are the random variables corresponding to the samples of $X(t)$.

We note that the output samples $\{X(n/2W)\}$ from the stationary sources are generally continuous, and, hence, they cannot be represented in digital form without some loss in precision. For example, we may quantize each sample to a set of discrete values, but the quantization process results in loss of precision, and, consequently, the original signal cannot be reconstructed exactly from the quantized sample values. Later in this chapter, we shall consider the distortion resulting from quantization of the samples from an analog source.

3-2 A LOGARITHMIC MEASURE OF INFORMATION

To develop an appropriate measure of information, let us consider two discrete random variables with possible outcomes x_i, $i = 1, 2, \ldots, n$, and y_i, $i = 1, 2, \ldots, m$, respectively. Suppose we observe some outcome $Y = y_j$ and we wish to determine, quantitatively, the amount of information that the occurrence of the event $Y = y_j$ provides about the event $X = x_i$, $i = 1, 2, \ldots, n$. We observe that when X and Y are statistically independent, the occurrence of $Y = y_j$ provides no information about the occurrence of the event $X = x_i$. On the other hand, when X and Y are fully dependent such that the occurrence of $Y = y_j$ determines the occurrence of $X = x_i$, the information content is simply that provided by the event $X = x_i$. A suitable measure that satisfies these conditions is the logarithm of the ratio of the conditional probability

$$P(X = x_i \mid Y = y_j) = P(x_i \mid y_j)$$

divided by the probability

$$P(X = x_i) = P(x_i)$$

That is, the information content provided by the occurrence of the event $Y = y_j$ about the event $X = x_i$ is defined as

$$I(x_i; y_j) = \log \frac{P(x_i \mid y_j)}{P(x_i)} \tag{3-2-1}$$

$I(x_i; y_j)$ is called the *mutual information* between x_i and y_j.

The units of $I(x_i; y_j)$ are determined by the base of the logarithm, which is usually selected as either 2 or e. When the base of the logarithm is 2, the units of $I(x_i; y_j)$ are bits, and when the base is e, the units of $I(x_i; y_j)$ are called *nats* (natural units). (The standard abbreviation for $\log_e$ is ln.) Since

$$\ln a = \ln 2 \log_2 a = 0.693\,15 \log_2 a$$

the information measured in nats is equal to ln 2 times the information measured in bits.

When the random variables X and Y are statistically independent,

$P(x_i \mid y_j) = P(x_i)$ and, hence, $I(x_i; y_j) = 0$. On the other hand, when the occurrence of the event $Y = y_j$ uniquely determines the occurrence of the event $X = x_i$, the conditional probability in the numerator of (3-2-1) is unity and, hence,

$$I(x_i; y_j) = \log \frac{1}{P(x_i)} = -\log P(x_i) \tag{3-2-2}$$

But (3-2-2) is just the information of the event $X = x_i$. For this reason, it is called the *self-information* of the event $X = x_i$ and it is denoted as

$$I(x_i) = \log \frac{1}{P(x_i)} = -\log P(x_i) \tag{3-2-3}$$

We note that a high-probability event conveys less information than a low-probability event. In fact, if there is only a single event x with probability $P(x) = 1$ then $I(x) = 0$. To demonstrate further that the logarithmic measure of information content is the appropriate one for digital communications, let us consider the following example.

Example 3-2-1

Suppose we have a discrete information source that emits a binary digit, either 0 or 1, with equal probability every τ_s seconds. The information content of each output from source is

$$\begin{aligned} I(x_i) &= -\log_2 P(x_i), \quad x_i = 0, 1 \\ &= -\log_2 \tfrac{1}{2} = 1 \text{ bit} \end{aligned}$$

Now suppose that successive outputs from the source are statistically independent, i.e., the source is memoryless. Let us consider a block of k binary digits from the source that occurs in a time interval $k\tau_s$. There are $M = 2^k$ possible k-bit blocks, each of which is equally probable with probability $1/M = 2^{-k}$. The self-information of a k-bit block is

$$I(x_i') = -\log_2 2^{-k} = k \text{ bits}$$

emitted in a time interval $k\tau_s$. Thus the logarithmic measure of information content possesses the desired additivity property when a number of source outputs is considered as a block.

Now let us return to the definition of mutual information given in (3-2-1) and multiply the numerator and denominator of the ratio of probabilities by $P(y_j)$. Since

$$\frac{P(x_i \mid y_j)}{P(x_i)} = \frac{P(x_i \mid y_j)P(y_j)}{P(x_i)P(y_j)} = \frac{P(x_i, y_j)}{P(x_i)P(y_j)} = \frac{P(y_j \mid x_i)}{P(y_j)}$$

we conclude that

$$I(x_i; y_j) = I(y_j; x_i) \tag{3-2-4}$$

Therefore the information provided by the occurrence of the event $Y = y_j$ about the event $X = x_i$ is identical to the information provided by the occurrence of the event $X = x_i$ about the event $Y = y_j$.

Example 3-2-2

Suppose that X and Y are binary-valued $\{0, 1\}$ random variables that represent the input and output of a binary-input, binary-output channel. The input symbols are equally likely and the output symbols depend on the input according to the conditional probabilities

$$P(Y = 0 \mid X = 0) = 1 - p_0$$
$$P(Y = 1 \mid X = 0) = p_0$$
$$P(Y = 1 \mid X = 1) = 1 - p_1$$
$$P(Y = 0 \mid X = 1) = p_1$$

Let us determine the mutual information about the occurrence of the events $X = 0$ and $X = 1$, given that $Y = 0$.

From the probabilities given above, we obtain

$$\begin{aligned} P(Y = 0) &= P(Y = 0 \mid X = 0)P(X = 0) + P(Y = 0 \mid X = 1)P(X = 1) \\ &= \tfrac{1}{2}(1 - p_0 + p_1) \\ P(Y = 1) &= P(Y = 1 \mid X = 0)P(X = 0) + P(Y = 1 \mid X = 1)P(X = 1) \\ &= \tfrac{1}{2}(1 - p_1 + p_0) \end{aligned}$$

Then, the mutual information about the occurrence of the event $X = 0$, given that $Y = 0$ is observed, is

$$I(x_1; y_1) = I(0; 0) = \log_2 \frac{P(Y = 0 \mid X = 0)}{P(Y = 0)} = \log_2 \frac{2(1 - p_0)}{1 - p_0 + p_1}$$

Similarly, given that $Y = 0$ is observed, the mutual information about the occurrence of the event $X = 1$ is

$$I(x_2; y_1) = I(1; 0) = \log_2 \frac{2p_1}{1 - p_0 + p_1}$$

Let us consider some special cases: First, if $p_0 = p_1 = 0$, the channel is called *noiseless* and

$$I(0; 0) = \log_2 2 = 1 \text{ bit}$$

Hence, the output specifies the input with certainty. On the other hand, if $p_0 = p_1 = \frac{1}{2}$, the channel is *useless* because

$$I(0; 0) = \log_2 1 = 0$$

However, if $p_0 = p_1 = \frac{1}{4}$, then

$$I(0; 0) = \log_2 \tfrac{3}{2} = 0.587$$

$$I(0; 1) = \log_2 \tfrac{1}{2} = -1 \text{ bit}$$

. In addition to the definition of mutual information and self-information, it is useful to define the *conditional self-information* as

$$I(x_i \mid y_j) = \log \frac{1}{P(x_i \mid y_j)} = -\log P(x_i \mid y_j) \tag{3-2-5}$$

Then, by combining (3-2-1), (3-2-3), and (3-2-5), we obtain the relationship

$$I(x_i; y_j) = I(x_i) - I(x_i \mid y_j) \tag{3-2-6}$$

We interpret $I(x_i \mid y_j)$ as the self-information about the event $X = x_i$ after having observed the event $Y = y_j$. Since both $I(x_i) \geq 0$ and $I(x_i \mid y_j) \geq 0$, it follows that $I(x_i; y_j) < 0$ when $I(x_i \mid y_j) > I(x_i)$, and $I(x_i; y_j) > 0$ when $I(x_i \mid y_j) < I(x_i)$. Hence, the mutual information between a pair of events can be either positive, or negative, or zero.

3-2-1 Average Mutual Information and Entropy

Having defined the mutual information associated with the pair of events (x_i, y_j), which are possible outcomes of the two random variables X and Y, we can obtain the average value of the mutual information by simply weighting $I(x_i; y_j)$ by the probability of occurrence of the joint event and summing over all possible joint events. Thus, we obtain

$$\begin{aligned} I(X; Y) &= \sum_{i=1}^{n} \sum_{j=1}^{m} P(x_i, y_j) I(x_i; y_j) \\ &= \sum_{i=1}^{n} \sum_{j=1}^{m} P(x_i, y_j) \log \frac{P(x_i, y_j)}{P(x_i) P(y_j)} \end{aligned} \tag{3-2-7}$$

as the average mutual information between X and Y. We observe that

$I(X;Y) = 0$ when X and Y are statistically independent. An important characteristic of the average mutual information is that $I(X;Y) \geq 0$ (see Problem 3-4).

Similarly, we define the average self-information, denoted by $H(X)$, as

$$H(X) = \sum_{i=1}^{n} P(x_i) I(x_i)$$

$$= -\sum_{i=1}^{n} P(x_i) \log P(x_i) \tag{3-2-8}$$

When X represents the alphabet of possible output letters from a source, $H(X)$ represents the average self-information per source letter, and it is called the *entropy*† of the source. In the special case in which the letters from the source are equally probable, $P(x_i) = 1/n$ for all i, and, hence,

$$H(X) = -\sum_{i=1}^{n} \frac{1}{n} \log \frac{1}{n}$$

$$= \log n \tag{3-2-9}$$

In general, $H(X) \leq \log n$ (see Problem 3-5) for any given set of source letter probabilities. In other words, *the entropy of a discrete source is a maximum when the output letters are equally probable.*

Example 3-2-3

Consider a source that emits a sequence of statistically independent letters, where each output letter is either 0 with probability q or 1 with probability $1 - q$. The entropy of this source is

$$H(X) \equiv H(q) = -q \log q - (1 - q) \log (1 - q) \tag{3-2-10}$$

The binary entropy function $H(q)$ is illustrated in Fig. 3-2-1. We observe that the maximum value of the entropy function occurs at $q = \frac{1}{2}$ where $H(\frac{1}{2}) = 1$.

The average conditional self-information is called the *conditional entropy* and is defined

$$H(X \mid Y) = \sum_{i=1}^{n} \sum_{j=1}^{m} P(x_i, y_j) \log \frac{1}{P(x_i \mid y_j)} \tag{3-2-11}$$

We interpret $H(X \mid Y)$ as the information or uncertainty in X after Y is

†The term *entropy* is taken from statistical mechanics (thermodynamics), where a function similar to (3-2-8) is called (thermodynamic) entropy.

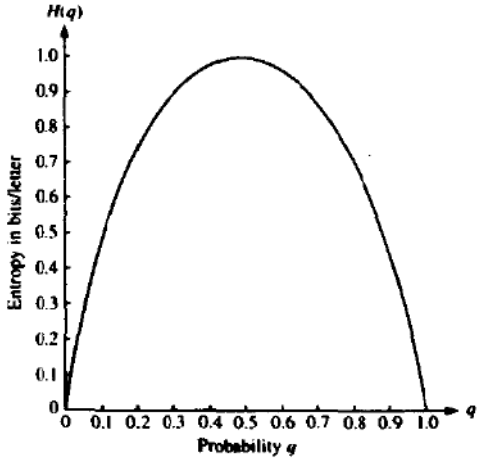


FIGURE 3-2-1 Binary entropy function.

observed. By combining (3-2-7), (3-2-8), and (3-2-11) we obtain the relationship

$$I(X;Y) = H(X) - H(X \mid Y) \tag{3-2-12}$$

Since $I(X;Y) \geq 0$, it follows that $H(X) \geq H(X \mid Y)$, with equality if and only if X and Y are statistically independent. If we interpret $H(X \mid Y)$ as the average amount of (conditional self-information) uncertainty in X after we observe Y, and $H(X)$ as the average amount of uncertainty (self-information) prior to the observation, then $I(X;Y)$ is the average amount of (mutual information) uncertainty provided about the set X by the observation of the set Y. Since $H(X) \geq H(X \mid Y)$, it is clear that conditioning on the observation Y does not increase the entropy.

Example 3-2-4

Let us evaluate the $H(X \mid Y)$ and $I(X;Y)$ for the binary-input, binary-output channel treated previously in Example 3-2-2 for the case where $p_0 = p_1 = p$. Let the probabilities of the input symbols be $P(X=0) = q$ and $P(X=1) = 1-q$. Then the entropy is

$$H(X) \equiv H(q) = -q \log q - (1-q) \log (1-q)$$

where $H(q)$ is the binary entropy function and the conditional entropy $H(X \mid Y)$ is defined by (3-2-11). A plot of $H(X \mid Y)$ as a function of q with

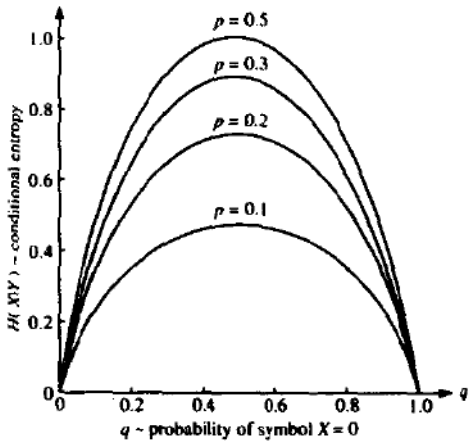


FIGURE 3-2-2 Conditional entropy for binary-input, binary-output symmetric channel.

p as a parameter is shown in Fig. 3-2-2. The average mutual information $I(X; Y)$ is plotted in Fig. 3-2-3.

As in the preceding example, when the conditional entropy $H(X \mid Y)$ is viewed in terms of a channel whose input is X and whose output is Y, $H(X \mid Y)$ is called the *equivocation* and is interpreted as the amount of average uncertainty remaining in X after observation of Y.

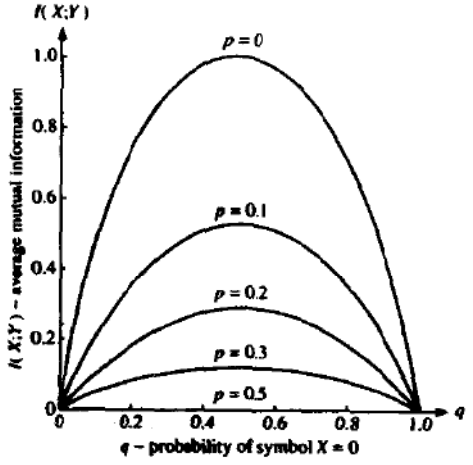


FIGURE 3-2-3 Average mutual information for binary-input, binary-output symmetric channel.

The results given above can be generalized to more than two random variables. In particular, suppose we have a block of k random variables $X_1X_2\cdots X_k$, with joint probability $P(x_1x_2\cdots x_k) \equiv P(X_1 = x_1, X_2 = x_2, \ldots, X_k = x_k)$. Then, the entropy for the block is defined as

$$H(X_1X_2\cdots X_k) = -\sum_{j_1=1}^{n_1}\sum_{j_2=1}^{n_2}\cdots\sum_{j_k=1}^{n_k} P(x_{j_1}x_{j_2}\cdots x_{j_k})\log P(x_{j_1}x_{j_2}\cdots x_{j_k}) \quad (3\text{-}2\text{-}13)$$

Since the joint probability $P(x_1x_2\cdots x_k)$ can be factored as

$$P(x_1x_2\cdots x_k) = P(x_1)P(x_2 \mid x_1)P(x_3 \mid x_1x_2)\cdots P(x_k \mid x_1x_2\cdots x_{k-1}) \quad (3\text{-}2\text{-}14)$$

it follows that

$$\begin{aligned} H(X_1X_2X_3\cdots X_k) &= H(X_1) + H(X_2 \mid X_1) + H(X_3 \mid X_1X_2) \\ &\quad + \ldots + H(X_k \mid X_1\cdots X_{k-1}) \\ &= \sum_{i=1}^{k} H(X_i \mid X_1X_2\cdots X_{i-1}) \end{aligned} \quad (3\text{-}2\text{-}15)$$

By applying the result $H(X) \geq H(X \mid Y)$, where $X = X_m$ and $Y = X_1X_2\cdots X_{m-1}$, in (3-2-15) we obtain

$$H(X_1X_2\cdots X_k) \leq \sum_{m=1}^{k} H(X_m) \quad (3\text{-}2\text{-}16)$$

with equality if and only if the random variables $X_1, X_2, \ldots, X_k$ are statistically independent.

3-2-2 Information Measures for Continuous Random Variables

The definition of mutual information given above for discrete random variables may be extended in a straightforward manner to continuous random variables. In particular, if X and Y are random variables with joint pdf $p(x, y)$ and marginal pdfs $p(x)$ and $p(y)$, the average mutual information between X and Y is defined as

$$I(X; Y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} p(x)p(y \mid x)\log\frac{p(y \mid x)p(x)}{p(x)p(y)}\,dx\,dy \quad (3\text{-}2\text{-}17)$$

Although the definition of the average mutual information carries over to

continuous random variables, the concept of self-information does not. The problem is that a continuous random variable requires an infinite number of binary digits to represent it exactly. Hence, its self-information is infinite and, therefore, its entropy is also infinite. Nevertheless, we shall define a quantity that we call the *differential entropy* of the continuous random variable X as

$$H(X) = -\int_{-\infty}^{\infty} p(x) \log p(x)\, dx \tag{3-2-18}$$

We emphasize that this quantity does *not* have the physical meaning of self-information, although it may appear to be a natural extension of the definition of entropy for a discrete random variable (see Problem 3-6).

By defining the average conditional entropy of X given Y as

$$H(X \mid Y) = -\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} p(x, y) \log p(x \mid y)\, dx\, dy \tag{3-2-19}$$

the average mutual information may be expressed as

$$I(X; Y) = H(X) - H(X \mid Y)$$

or, alternatively, as

$$I(X; Y) = H(Y) - H(Y \mid X)$$

In some cases of practical interest, the random variable X is discrete and Y is continuous. To be specific, suppose that X has possible outcomes x_i, $i = 1, 2, \ldots, n$, and Y is described by its marginal pdf $p(y)$. When X and Y are statistically dependent, we may express $p(y)$ as

$$p(y) = \sum_{i=1}^{n} p(y \mid x_i) P(x_i)$$

The mutual information provided about the event $X = x_i$ by the occurrence of the event $Y = y$ is

$$\begin{aligned} I(x_i; y) &= \log \frac{p(y \mid x_i) P(x_i)}{p(y) P(x_i)} \\ &= \log \frac{p(y \mid x_i)}{p(y)} \end{aligned} \tag{3-2-20}$$

Then, the average mutual information between X and Y is

$$I(X; Y) = \sum_{i=1}^{n} \int_{-\infty}^{\infty} p(y \mid x_i) P(x_i) \log \frac{p(y \mid x_i)}{p(y)}\, dy \tag{3-2-21}$$

Example 3-2-5

Suppose that X is a discrete random variable with two equally probable outcomes $x_1 = A$ and $x_2 = -A$. Let the conditional pdfs $p(y \mid x_i)$, $i = 1, 2$, be gaussian with mean x_i and variance σ^2. That is,

$$p(y \mid A) = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-(y-A)^2/2\sigma^2} \tag{3-2-22}$$

$$p(y \mid -A) = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-(y+A)^2/2\sigma^2} \tag{3-2-22}$$

The average mutual information obtained from (3-2-21) becomes

$$I(X;Y) = \frac{1}{2}\int_{-\infty}^{\infty}\left[p(y \mid A)\log\frac{p(y \mid A)}{p(y)} + p(y \mid -A)\log\frac{p(y \mid -A)}{p(y)}\right]dy \tag{3-2-23}$$

$$p(y) = \tfrac{1}{2}[p(y \mid A) + p(y \mid -A)] \tag{3-2-24}$$

In Chapter 7, it will be shown that the average mutual information $I(X;Y)$ given by (3-2-23) represents the channel capacity of a binary-input additive white gaussian noise channel.

3-3 CODING FOR DISCRETE SOURCES

In Section 3-2 we introduced a measure for the information content associated with a discrete random variable X. When X is the output of a discrete source, the entropy $H(X)$ of the source represents the average amount of information emitted by the source. In this section, we consider the process of encoding the output of a source, i.e., the process of representing the source output by a sequence of binary digits. A measure of the efficiency of a source-encoding method can be obtained by comparing the average number of binary digits per output letter from the source to the entropy $H(X)$.

The encoding of a discrete source having a finite alphabet size may appear, at first glance, to be a relatively simple problem. However, this is true only when the source is memoryless, i.e., when successive symbols from the source are statistically independent and each symbol is encoded separately. The discrete memoryless source (DMS) is by far the simplest model that can be devised for a physical source. Few physical sources, however, closely fit this idealized mathematical model. For example, successive output letters from a machine printing English text are expected to be statistically dependent. On the other hand, if the machine output is a computer program coded in Fortran, the sequence of output letters is expected to exhibit a much smaller dependence. In any case, we shall demonstrate that it is always more efficient to encode blocks of symbol instead of encoding each symbol separately. By making the block size sufficiently large, the average number of binary digits

per output letter from the source can be made arbitrarily close to the entropy of the source.

3-3-1 Coding for Discrete Memoryless Sources

Suppose that a DMS produces an output letter or symbol every τ_s seconds. Each symbol is selected from a finite alphabet of symbols x_i, $i = 1, 2, \ldots, L$, occurring with probabilities $P(x_i)$, $i = 1, 2, \ldots, L$. The entropy of the DMS in bits per source symbol is

$$H(X) = -\sum_{i=1}^{L} P(x_i) \log_2 P(x_i) \leq \log_2 L \tag{3-3-1}$$

where equality holds when the symbols are equally probable. The average number of bits per source symbol is $H(X)$ and the source rate in bits/s is defined as $H(X)/\tau_s$.

Fixed-Length Code Words First we consider a block encoding scheme that assigns a unique set of R binary digits to each symbol. Since there are L possible symbols, the number of binary digits per symbol required for unique encoding when L is a power of 2 is

$$R = \log_2 L \tag{3-3-2}$$

and, when L is not a power of 2, it is

$$R = \lfloor \log_2 L \rfloor + 1 \tag{3-3-3}$$

where $\lfloor x \rfloor$ denotes the largest integer less than x. The code rate R in bits per symbol is now R and, since $H(X) \leq \log_2 L$, it follows that $R \geq H(X)$.

The efficiency of the encoding for the DMS is defined as the ratio $H(X)/R$. We observe that when L is a power of 2 and the source letters are equally probable, $R = H(X)$. Hence, a fixed-length code of R bits per symbol attains 100% efficiency. However, if L is not a power of 2 but the source symbols are still equally probable, R differs from $H(X)$ by at most 1 bit per symbol. When $\log_2 L \gg 1$, the efficiency of this encoding scheme is high. On the other hand, when L is small, the efficiency of the fixed-length code can be increased by encoding a sequence of J symbols at a time. To accomplish the desired encoding, we require L^J unique code words. By using sequences of N binary digits, we can accommodate 2^N possible code words. N must be selected such that

$$N \geq J \log_2 L$$

Hence, the minimum integer value of N required is

$$N = \lfloor J \log_2 L \rfloor + 1 \tag{3-3-4}$$

Now the average number of bits per source symbol is $N/J = R$, and, thus, the

inefficiency has been reduced by approximately a factor of $1/J$ relative to the symbol-by-symbol encoding described above. By making J sufficiently large, the efficiency of the encoding procedure, measured by the ratio $JH(X)/N$, can be made as close to unity as desired.

The encoding methods described above introduce no distortion since the encoding of source symbols or blocks of symbols into code words is unique. This type of encoding is called *noiseless*.

Now, suppose we attempt to reduce the code rate R by relaxing the condition that the encoding process be unique. For example, suppose that only a fraction of the L^J blocks of symbols is encoded uniquely. To be specific, let us select the $2^N - 1$ most probable J-symbol blocks and encode each of them uniquely, while the remaining $L^J - (2^N - 1)$ J-symbol blocks are represented by the single remaining code word. This procedure results in a decoding failure or (distortion) probability of error every time a low probability block is mapped into this single code word. Let P_e denote this probability of error. Based on this block encoding procedure, Shannon (1948a) proved the following source coding theorem.

Source Coding Theorem I

Let X be the ensemble of letters from a DMS with finite entropy $H(X)$. Blocks of J symbols from the source are encoded into code words of length N from a binary alphabet. For any $\varepsilon > 0$, the probability P_e of a block decoding failure can be made arbitrarily small if

$$R \equiv \frac{N}{J} \geq H(X) + \varepsilon \tag{3-3-5}$$

and J is sufficiently large. Conversely, if

$$R \leq H(X) - \varepsilon \tag{3-3-6}$$

then P_e becomes arbitrarily close to 1 as J is made sufficiently large.

From this theorem, we observe that the average number of bits per symbol required to encode the output of a DMS with arbitrarily small probability of decoding failure is lower bounded by the source entropy $H(X)$. On the other hand, if $R < H(X)$, the decoding failure rate approaches 100% as J is arbitrarily increased.

Variable-Length Code Words When the source symbols are not equally probable, a more efficient encoding method is to use variable-length code

TABLE 3-3-1 VARIABLE-LENGTH CODES

Letter	$P(a_i)$	Code I	Code II	Code III
a_1	$\frac{1}{2}$	1	0	0
a_2	$\frac{1}{4}$	00	10	01
a_3	$\frac{1}{8}$	01	110	011
a_4	$\frac{1}{8}$	10	111	111

words. An example of such encoding is the Morse code, which dates back to the nineteenth century. In the Morse code, the letters that occur more frequently are assigned short code words and those that occur infrequently are assigned long code words. Following this general philosophy, we may use the probabilities of occurrence of the different source letters in the selection of the code words. The problem is to devise a method for selecting and assigning the code words to source letters. This type of encoding is called *entropy coding.*

For example, suppose that a DMS with output letters a_1, a_2, a_3, a_4 and corresponding probabilities $P(a_1) = \frac{1}{2}$, $P(a_2) = \frac{1}{4}$, and $P(a_3) = P(a_4) = \frac{1}{8}$ is encoded as shown in Table 3-3-1. Code I is a variable-length code that has a basic flaw. To see the flaw, suppose we are presented with the sequence 001001 .... Clearly, the first symbol corresponding to 00 is a_2. However, the next four bits are ambiguous (not uniquely decodable). They may be decoded either as a_4a_3 or as $a_1a_2a_1$. Perhaps, the ambiguity can be resolved by waiting for additional bits, but such a decoding delay is highly undesirable. We shall only consider codes that are decodable *instantaneously*, that is, without any decoding delay.

Code II in Table 3-3-1 is *uniquely decodable* and *instantaneously decodable.* It is convenient to represent the code words in this code graphically as terminal nodes of a tree, as shown in Fig. 3-3-1. We observe that the digit 0 indicates the end of a code word for the first three code words. This characteristic plus the fact that no code word is longer than three binary digits makes this code instantaneously decodable. Note that no code word in this code is a prefix of any other code word. In general, the *prefix condition* requires that for a given code word $\mathbf{C}_k$ of length k having elements $(b_1, b_2, \ldots, b_k)$, there is no other code word of length $l < k$ with elements $(b_1, b_2, \ldots, b_l)$ for $1 \leq l \leq k-1$. In

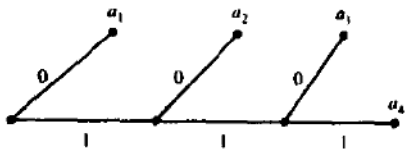


FIGURE 3-3-1 Code tree for code II in Table 3-3-1.

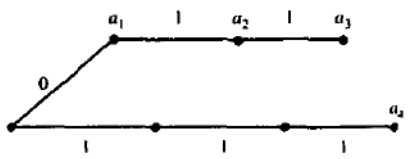


FIGURE 3-3-2 Code tree for code III in Table 3-3-1.

other words, there is no code word of length $l < k$ that is identical to the first l binary digits of another code word of length $k > l$. This property makes the code words instantaneously decodable.

Code III given in Table 3-3-1 has the tree structure shown in Fig. 3-3-2. We note that in this case the code is uniquely decodable but *not* instantaneously decodable. Clearly, this code does *not* satisfy the prefix condition.

Our main objective is to devise a systematic predure for constructing uniquely decodable variable-length codes that are efficient in the sense that the average number of bits per source letter, defined as the quantity

$$\bar{R} = \sum_{k=1}^{L} n_k P(a_k) \tag{3-3-7}$$

is minimized. The conditions for the existence of a code that satisfies the prefix condition are given by the Kraft inequality.

Kraft Inequality A necessary and sufficient condition for the existence of a binary code with code words having lengths $n_1 \leq n_2 \leq \ldots \leq n_L$ that satisfy the prefix condition is

$$\sum_{k=1}^{L} 2^{-n_k} \leq 1 \tag{3-3-8}$$

First, we prove that (3-3-8) is a sufficient condition for the existence of a code that satisfies the prefix condition. To construct such a code, we begin with a full binary tree of order $n = n_L$ that has 2^n terminal nodes and two nodes of order k stemming from each node of order $k-1$, for each k, $1 \leq k \leq n$. Let us select any node of order n_1 as the first code word $\mathbf{C}_1$. This choice eliminates 2^{n-n_1} terminal nodes (or the fraction 2^{-n_1} of the 2^n terminal nodes). From the remaining available nodes of order n_2, we select one node for the second code word $\mathbf{C}_2$. This choice eliminates 2^{n-n_2} terminal nodes (or the fraction 2^{-n_2} of the 2^n terminal nodes). This process continues until the last code word is assigned at terminal node $n = n_L$. Since, at the node of order $j < L$, the fraction of the number of terminal nodes eliminated is

$$\sum_{k=1}^{j} 2^{-n_k} < \sum_{k=1}^{L} 2^{-n_k} \leq 1$$

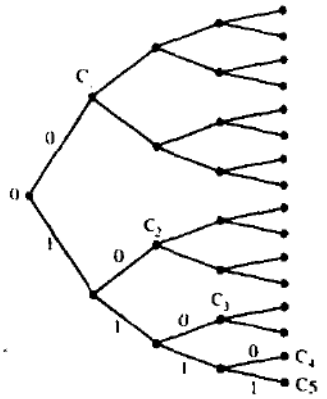


FIGURE 3-3-3 Construction of a binary tree code embedded in a full tree.

there is always a node of order $k > j$ available to be assigned to the next code word. Thus, we have constructed a code tree that is embedded in the full tree of 2^n nodes as illustrated in Fig. 3-3-3, for a tree having 16 terminal nodes and a source output consisting of five letters with $n_1 = 1$, $n_2 = 2$, $n_3 = 3$, and $n_4 = n_5 = 4$.

To prove that (3-3-8) is a necessary condition, we observe that in the code tree of order $n = n_L$, the number of terminal nodes eliminated from the total number of 2^n terminal nodes is

$$\sum_{k=1}^{L} 2^{n-n_k} \leq 2^n$$

Hence,

$$\sum_{k=1}^{L} 2^{-n_k} \leq 1$$

and the proof of (3-3-8) is complete.

The Kraft inequality may be used to prove the following (noiseless) source coding theorem, which applies to codes that satisfy the prefix condition.

Source Coding Theorem II

Let X be the ensemble of letters from a DMS with finite entropy $H(X)$, and output letters x_k, $1 \leq k \leq L$, with corresponding probabilities of occurrence p_k, $1 \leq k \leq L$. It is possible to construct a code that satisfies the prefix condition and has an average length $\bar{R}$ that satisfies the inequalities

$$H(X) \leq \bar{R} < H(X) + 1 \qquad (3\text{-}3\text{-}9)$$

To establish the lower bound in (3-3-9), we note that for code words that have length n_k, $1 \leq k \leq L$, the difference $H(X) - \bar{R}$ may be expressed as

$$H(X) - \bar{R} = \sum_{k=1}^{L} p_k \log_2 \frac{1}{p_k} - \sum_{k=1}^{L} p_k n_k$$

$$= \sum_{k=1}^{L} p_k \log_2 \frac{2^{-n_k}}{p_k} \tag{3-3-10}$$

Use of the inequality $\ln x \leq x - 1$ in (3-3-10) yields

$$H(X) - \bar{R} \leq (\log_2 e) \sum_{k=1}^{L} p_k \left(\frac{2^{-n_k}}{p_k} - 1 \right)$$

$$\leq (\log_2 e) \left(\sum_{k=1}^{L} 2^{-n_k} - 1 \right) \leq 0$$

where the last inequality follows from the Kraft inequality. Equality holds if and only if $p_k = 2^{-n_k}$ for $1 \leq k \leq L$.

The upper bound in (3-3-9) may be established under the constraint that n_k, $1 \leq k \leq L$, are integers, by selecting the $\{n_k\}$ such that $2^{-n_k} \leq p_k < 2^{-n_k+1}$. But if the terms $p_k \geq 2^{-n_k}$ are summed over $1 \leq k \leq L$, we obtain the Kraft inequality, for which we have demonstrated that there exists a code that satisfies the prefix condition. On the other hand, if we take the logarithm of $p_k < 2^{-n_k+1}$, we obtain

$$\log p_k < -n_k + 1$$

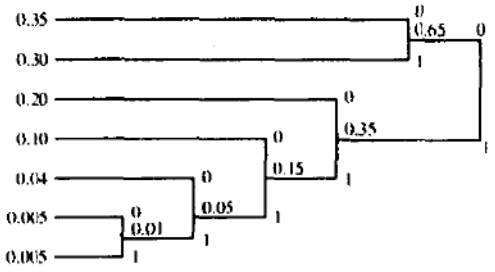
or, equivalently,

$$n_k < 1 - \log p_k \tag{3-3-11}$$

If we multiply both sides of (3-3-11) by p_k and sum over $1 \leq k \leq L$, we obtain the desired upper bound given in (3-3-9). This completes the proof of (3-3-9).

We have now established that variable length codes that satisfy the prefix condition are efficient source codes for any DMS with source symbols that are not equally probable. Let us now describe an algorithm for constructing such codes.

Huffman Coding Algorithm Huffman (1952) devised a variable-length encoding algorithm, based on the source letter probabilities $P(x_i)$, $i = 1, 2, \ldots, L$. This algorithm is optimum in the sense that the average number of binary digits required to represent the source symbols is a minimum, subject to the constraint that the code words satisfy the prefix condition, as defined above, which allows the received sequence to be uniquely and instantaneously decodable. We illustrate this encoding algorithm by means of two examples.



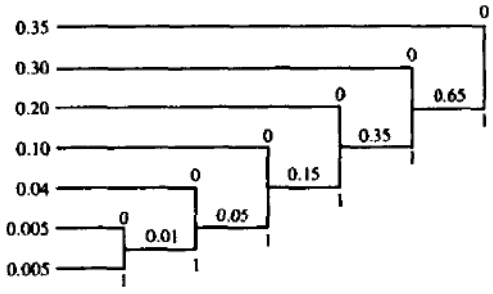
Letter	Probability	Self-information	Code
x_1	0.35	1.5146	00
x_2	0.30	1.7370	01
x_3	0.20	2.3219	10
x_4	0.10	3.3219	110
x_5	0.04	4.6439	1110
x_6	0.005	7.6439	11110
x_7	0.005	7.6439	11111
	$H(X) = 2.11$	$\bar{R} = 2.21$	

FIGURE 3-3-4 An example of variable-length-source encoding for a DMS.

Example 3-3-1

Consider a DMS with seven possible symbols $x_1, x_2, \ldots, x_7$ having the probabilities of occurrence illustrated in Fig. 3-3-4. We have ordered the source symbols in decreasing order of the probabilities, i.e., $P(x_1) > P(x_2) > \ldots > P(x_7)$. We begin the encoding process with the two least probable symbols x_6 and x_7. These two symbols are tied together as shown in Fig. 3-3-4, with the upper branch assigned a 0 and the lower branch assigned a 1. The probabilities of these two branches are added together at the node where the two branches meet to yield the probability 0.01. Now we have the source symbols $x_1, \ldots, x_5$ plus a new symbol, say x_6', obtained by combining x_6 and x_7. The next step is to join the two least probable symbols from the set x_1, x_2, x_3, x_4, x_5, x_6'. These are x_5 and x_6', which have a combined probability of 0.05. The branch from x_5 is assigned a 0 and the branch from x_6' is assigned a 1. This procedure continues until we exhaust the set of possible source letters. The result is a code tree with branches that contain the desired code words. The code words are obtained by beginning at the rightmost node in the tree and proceeding to the left. The resulting code words are listed in Fig. 3-3-4. The average number of binary digits per symbol for this code is $\bar{R} = 2.21$ bits/symbol. The entropy of the source is 2.11 bits/symbol.

We make the observation that the code is not necessarily unique. For example, at the next to the last step in the encoding procedure, we have a tie between x_1 and x_3', since these symbols are equally probable. At this point, we chose to pair x_1 with x_2. An alternative is to pair x_2 with x_3'. If we choose this



Letter	Code
x_1	0
x_2	10
x_3	110
x_4	1110
x_5	11110
x_6	111110
x_7	111111

$\bar{R} = 2.21$

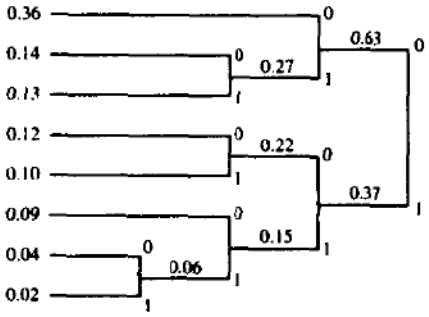
FIGURE 3-3-5 An alternative code for the DMS in Example 3-3-1.

pairing, the resulting code is illustrated in Fig. 3-3-5. The average number of bits per source symbol for this code is also 2.21. Hence, the resulting codes are equally efficient. Secondly, the assignment of a 0 to the upper branch and a 1 to the lower (less probable) branch is arbitrary. We may simply reverse the assignment of a 0 and 1 and still obtain an efficient code satisfying the prefix condition.

Example 3-3-2

As a second example, let us determine the Huffman code for the output of a DMS illustrated in Fig. 3-3-6. The entropy of this source is $H(X) = 2.63$ bits/symbol. The Huffman code as illustrated in Fig. 3-3-6 has an average length of $\bar{R} = 2.70$ bits/symbol. Hence, its efficiency is 0.97.

FIGURE 3-3-6 *Huffman code for Example 3-3-2.*



Letter	Code
x_1	00
x_2	010
x_3	011
x_4	100
x_5	101
x_6	110
x_7	1110
x_8	1111

$H(X) = 2.63$ $\bar{R} = 2.70$

The variable-length encoding (Huffman) algorithm described in the above examples generates a prefix code having an $\bar{R}$ that satisfies (3-3-9). However, instead of encoding on a symbol-by-symbol basis, a more efficient procedure is to encode blocks of J symbols at a time. In such a case, the bounds in (3-3-9) of source coding theorem II become

$$JH(X) \leq \bar{R}_J < JH(X) + 1, \tag{3-3-12}$$

since the entropy of a J-symbol block from a DMS is $JH(X)$, and $\bar{R}_J$ is the average number of bits per J-symbol blocks. If we divide (3-3-12) by J, we obtain

$$H(X) \leq \frac{\bar{R}_J}{J} < H(X) + \frac{1}{J} \tag{3-3-13}$$

where $\bar{R}_J/J = \bar{R}$ is the average number of bits per source symbol. Hence $\bar{R}$ can be made as close to $H(X)$ as desired by selecting J sufficiently large.

Example 3-3-3

The output of a DMS consists of *letters* x_1, x_2, and x_3 with probabilities 0.45, 0.35, and 0.20, respectively. The entropy of this source is $H(X) = 1.518$ bits/symbol. The Huffman code for this source, given in Table 3-3-2, requires $\bar{R}_1 = 1.55$ bits/symbol and results in an efficiency of 97.9%. If pairs of symbols are encoded by means of the Huffman algorithm, the resulting code is as given in Table 3-3-3. The entropy of the source output for pairs of letters is $2H(X) = 3.036$ bits/symbol pair. On the other hand, the Huffman code requires $\bar{R}_2 = 3.0675$ bits/symbol pair. Thus, the efficiency of the encoding increases to $2H(X)/\bar{R}_2 = 0.990$ or, equivalently, to 99.0%.

In summary, we have demonstrated that efficient encoding for a DMS may be done on a symbol-by-symbol basis using a variable-length code based on

TABLE 3-3-2 HUFFMAN CODE FOR EXAMPLE 3-3-3

Letter	Probability	Self-information	Code
x_1	0.45	1.156	1
x_2	0.35	1.520	00
x_3	0.20	2.330	01

$H(X) = 1.518$ bits/letter
$\bar{R}_1 = 1.55$ bits/letter
Efficiency = 97.9%

TABLE 3-3-3 HUFFMAN CODE FOR ENCODING PAIRS OF LETTERS

Letter pair	Probability	Self-information	Code
x_1x_1	0.2025	2.312	10
x_1x_2	0.1575	2.676	001
x_2x_1	0.1575	2.676	010
x_2x_2	0.1225	3.039	011
x_1x_3	0.09	3.486	111
x_3x_1	0.09	3.486	0000
x_2x_3	0.07	3.850	0001
x_3x_2	0.07	3.850	1100
x_3x_3	0.04	4.660	1101

$2H(X) = 3.036$ bits/letter pair
$\bar{R}_2 = 3.0675$ bits/letter pair
$\frac{1}{2}\bar{R}_2 = 1.534$ bits/letter
Efficiency = 99.0%

the Huffman algorithm. Furthermore, the efficiency of the encoding procedure is increased by encoding blocks of J symbols at a time. Thus, the output of a DMS with entropy $H(X)$ may be encoded by a variable-length code with an average number of bits per source letter that approaches $H(X)$ as closely as desired.

3-3-2 Discrete Stationary Sources

In the previous section, we described the efficient encoding of the output of a DMS. In this section, we consider discrete sources for which the sequence of output letters is statistically dependent. We limit our treatment to sources that are statistically stationary.

Let us evaluate the entropy of any sequence of letters from a stationary source. From the definition in (3-2-13) and the result given in (3-2-15), the entropy of a block of random variables $X_1X_2 \cdots X_k$ is

$$H(X_1X_2 \cdots X_k) = \sum_{i=1}^{k} H(X_i \mid X_1X_2 \cdots X_{i-1}) \tag{3-3-14}$$

where $H(X_i \mid X_1X_2 \cdots X_{i-1})$ is the conditional entropy of the ith symbol from the source given the previous $i-1$ symbols. The entropy per letter for the k-symbol block is defined as

$$H_k(X) = \frac{1}{k} H(X_1X_2 \cdots X_k) \tag{3-3-15}$$

We define the information content of a stationary source as the entropy per letter in (3-3-15) in the limit as $k \to \infty$. That is,

$$H_\infty(X) = \lim_{k\to\infty} H_k(X) = \lim_{k\to\infty} \frac{1}{k} H(X_1X_2 \cdots X_k) \tag{3-3-16}$$

The existence of this limit is established below.

As an alternative, we may define the entropy per letter from the source in terms of the conditional entropy $H(X_k \mid X_1X_2 \cdots X_{k-1})$ in the limit as k approaches infinity. Fortunately, this limit also exists and is identical to the limit in (3-3-16). That is,

$$H_\infty(X) = \lim_{k \to \infty} H(X_k \mid X_1X_2 \cdots X_{k-1}) \tag{3-3-17}$$

This result is also established below. Our development follows the approach in Gallager (1968).

First, we show that

$$H(X_k \mid X_1X_2 \cdots X_{k-1}) \leq H(X_{k-1} \mid X_1X_2 \cdots X_{k-2}) \tag{3-3-18}$$

for $k \geq 2$. From our previous result that conditioning on a random variable cannot increase entropy, we have

$$H(X_k \mid X_1X_2 \cdots X_{k-1}) \leq H(X_k \mid X_2X_3 \cdots X_{k-1}) \tag{3-3-19}$$

From the stationarity of the source, we have

$$H(X_k \mid X_2X_3 \cdots X_{k-1}) = H(X_{k-1} \mid X_1X_2 \cdots X_{k-2}) \tag{3-3-20}$$

Hence, (3-3-18) follows immediately. This result demonstrates that $H(X_k \mid X_1X_2 \cdots X_{k-1})$ is a nonincreasing sequence in k.

Second, we have the result

$$H_k(X) \geq H(X_k \mid X_1X_2 \cdots X_{k-1}) \tag{3-3-21}$$

which follows immediately from (3-3-14) and (3-3-15) and the fact that the last term in the sum of (3-3-14) is a lower bound on each of the other $k-1$ terms.

Third, from the definition of $H_k(X)$, we may write

$$\begin{aligned} H_k(X) &= \frac{1}{k}[H(X_1X_2 \cdots X_{k-1}) + H(X_k \mid X_1 \cdots X_{k-1})] \\ &= \frac{1}{k}[(k-1)H_{k-1}(X) + H(X_k \mid X_1 \cdots X_{k-1})] \\ &\leq \frac{k-1}{k} H_{k-1}(X) + \frac{1}{k} H_k(X) \end{aligned}$$

which reduces to

$$H_k(X) \leq H_{k-1}(X) \tag{3-3-22}$$

Hence, $H_k(X)$ is a nonincreasing sequence in k.

Since $H_k(X)$ and the conditional entropy $H(X_k \mid X_1 \cdots X_{k-1})$ are both

nonnegative and nonincreasing with k, both limits must exist. Their limiting forms can be established by using (3-3-14) and (3-3-15) to express $H_{k+j}(X)$ as

$$\begin{aligned} H_{k+j}(X) = \frac{1}{k+j} H(X_1 X_2 \cdots X_{k-1}) \\ + \frac{1}{k+j}[H(X_k \mid X_1 \cdots X_{k-1}) + H(X_{k+1} \mid X_1 \cdots X_k) \\ + \ldots + H(X_{k+j} \mid X_1 \cdots X_{k+j-1})] \end{aligned}$$

Since the conditional entropy is nonincreasing, the first term in the square brackets serves as an upper bound on the other terms. Hence,

$$H_{k+j}(X) \leq \frac{1}{k+j} H(X_1 X_2 \cdots X_{k-1}) + \frac{j+1}{k+j} H(X_k \mid X_1 X_2 \cdots X_{k-1}) \tag{3-3-23}$$

For a fixed k, the limit of (3-3-23) as $j \to \infty$ yields

$$H_\infty(X) \leq H(X_k \mid X_1 X_2 \cdots X_{k-1}) \tag{3-3-24}$$

But (3-3-24) is valid for all k; hence, it is valid for $k \to \infty$. Therefore,

$$H_\infty(X) \leq \lim_{k \to \infty} H(X_k \mid X_1 X_2 \cdots X_{k-1}) \tag{3-3-25}$$

On the other hand, from (3-3-21), we obtain in the limit as $k \to \infty$,

$$H_\infty(X) \geq \lim_{k \to \infty} H(X_k \mid X_1 X_2 \cdots X_{k-1}) \tag{3-3-26}$$

which establishes (3-3-17).

Now suppose we have a discrete stationary source that emits J letters with $H_J(X)$ as the entropy per letter. We can encode the sequence of J letters with a variable-length Huffman code that satisfies the prefix condition by following the procedure described in the previous section. The resulting code has an average number of bits for the J-letter block that satisfies the condition

$$H(X_1 \cdots X_J) \leq \bar{R}_J < H(X_1 \cdots X_J) + 1 \tag{3-3-27}$$

By dividing each term of (3-3-27) by J, we obtain the bounds on the average number $\bar{R} = \bar{R}_J/J$ of bits per source letter as

$$H_J(X) \leq \bar{R} < H_J(X) + \frac{1}{J} \tag{3-3-28}$$

By increasing the block size J, we can approach $H_J(X)$ arbitrarily closely, and in the limit as $J \to \infty$, $\bar{R}$ satisfies

$$H_\infty(X) \leq \bar{R} < H_\infty(X) + \varepsilon \tag{3-3-29}$$

where ε approaches zero as $1/J$. Thus, efficient encoding of stationary sources is accomplished by encoding large blocks of symbols into code words. We should emphasize, however, that the design of the Huffman code requires knowledge of the joint pdf for the J-symbol blocks.

ıe Lempel–Ziv Algorithm

From our preceding discussion, we have observed that the Huffman coding algorithm yields optimal source codes in the sense that the code words satisfy the prefix condition and the average block length is a minimum. To design a Huffman code for a DMS, we need to know the probabilities of occurrence of all the source letters. In the case of a discrete source with memory, we must know the joint probabilities of blocks of length $n \geq 2$. However, in practice, the statistics of a source output are often unknown. In principle, it is possible to estimate the probabilities of the discrete source output by simply observing a long information sequence emitted by the source and obtaining the probabilities empirically. Except for the estimation of the marginal probabilities $\{p_k\}$, corresponding to the frequency of occurrence of the individual source output letters, the computational complexity involved in estimating joint probabilities is extremely high. Consequently, the application of the Huffman coding method to source coding for many real sources with memory is generally impractical.

In contrast to the Huffman coding algorithm, the Lempel–Ziv source coding algorithm is designed to be independent of the source statistics. Hence, the Lempel–Ziv algorithm belongs to the class of *universal source coding algorithms*. It is a variable-to-fixed-length algorithm, where the encoding is performed as described below.

In the Lempel–Ziv algorithm, the sequence at the output of the discrete source is parsed into variable-length blocks, which are called *phrases*. A new phrase is introduced every time a block of letters from the source differs from some previous phrase in the last letter. The phrases are listed in a dictionary, which stores the location of the existing phrases. In encoding a new phrase, we simply specify the location of the existing phrase in the dictionary and append the new letter.

As an example, consider the binary sequence

1010110100100111010100001100111010110001101 1

Parsing the sequence as described above produces the following phrases:

1, 0, 10, 11, 01, 00, 100, 111, 010, 1000, 011, 001, 110, 101, 10001, 1011

We observe that each phrase in the sequence is a concatenation of a previous phrase with a new output letter from the source. To encode the phrases, we

TABLE 3-3-4 DICTIONARY FOR LEMPEL-ZIV ALGORITHM

	Dictionary location	Dictionary contents	Code word
1	0001	1	00001
2	0010	0	00000
3	0011	10	00010
4	0100	11	00011
5	0101	01	00101
6	0110	00	00100
7	0111	100	00110
8	1000	111	01001
9	1001	010	01010
10	1010	1000	01110
11	1011	011	01011
12	1100	001	01101
13	1101	110	01000
14	1110	101	00111
15	1111	10001	10101
16		1011	11101

construct a dictionary as shown in Table 3-3-4. The dictionary locations are numbered consecutively, beginning with 1 and counting up, in this case to 16, which is the number of phrases in the sequence. The different phrases corresponding to each location are also listed, as shown. The codewords are determined by listing the dictionary location (in binary form) of the previous phrase that matches the new phrase in all but the last location. Then, the new output letter is appended to the dictionary location of the previous phrase. Initially, the location 0000 is used to encode a phrase that has not appeared previously.

The source decoder for the code constructs an identical table at the receiving end of the communication system and decodes the received sequence accordingly.

It should be observed that the table encoded 44 source bits into 16 code words of five bits each, resulting in 80 coded bits. Hence, the algorithm provided no data compression at all. However, the inefficiency is due to the fact that the sequence we have considered is very short. As the sequence is increased in length, the encoding procedure becomes more efficient and results in a compressed sequence at the output of the source.

How do we select the overall length of the table? In general, no matter how large the table is, it will eventually overflow. To solve the overflow problem, the source encoder and source decoder must agree to remove phrases from the respective dictionaries that are not useful and substitute new phrases in their place.

The Lempel–Ziv algorithm is widely used in the compression of computer files. The "compress" and "uncompress" utilities under the UNIX© operating system and numerous algorithms under the MS-DOS operating system are implementations of various versions of this algorithm.

3-4 CODING FOR ANALOG SOURCES—OPTIMUM QUANTIZATION

As indicated in Section 3-1, an analog source emits a message waveform $x(t)$ that is a sample function of a stochastic process $X(t)$. When $X(t)$ is a bandlimited, stationary stochastic process, the sampling theorem allows us to represent $X(t)$ by a sequence of uniform samples taken at the Nyquist rate.

By applying the sampling theorem, the output of an analog source is converted to an equivalent discrete-time sequence of samples. The samples are then quantized in amplitude and encoded. One type of simple encoding is to represent each discrete amplitude level by a sequence of binary digits. Hence, if we have L levels, we need $R = \log_2 L$ bits per sample if L is a power of 2, or $R = \lfloor \log_2 L \rfloor + 1$ if L is not a power of 2. On the other hand, if the levels are not equally probable, and the probabilities of the output levels are known, we may use Huffman coding (also called *entropy coding*) to improve the efficiency of the encoding process.

Quantization of the amplitudes of the sampled signal results in data compression but it also introduces some distortion of the waveform or a loss of signal fidelity. The minimization of this distortion is considered in this section. Many of the results given in this section apply directly to a discrete-time, continuous amplitude, memoryless gaussian source. Such a source serves as a good model for the residual error in a number of source coding methods described in Section 3-5.

3-4-1 Rate-Distortion Function

Let us begin the discussion of signal quantization by considering the distortion introduced when the samples from the information source are quantized to a fixed number of bits. By the term "distortion," we mean some measure of the difference between the actual source samples $\{x_k\}$ and the corresponding quantized values $\tilde{x}_k$, which we denote by $d\{x_k, \tilde{x}_k\}$. For example, a commonly used distortion measure is the *squared-error distortion*, defined as

$$d(x_k, \tilde{x}_k) = (x_k - \tilde{x}_k)^2 \tag{3-4-1}$$

which is used to characterize the quantization error in PCM in Section 3-5-1. Other distortion measures may take the general form

$$d(x_k, \tilde{x}_k) = |x - \tilde{x}_k|^p \tag{3-4-2}$$

where p takes values from the set of positive integers. The case $p = 2$ has the advantage of being mathematically tractable.

If $d(x_k, \tilde{x}_k)$ is the distortion measure per letter, the distortion between a sequence of n samples $\mathbf{X}_n$ and the corresponding n quantized values $\tilde{\mathbf{X}}_n$ is the average over the n source output samples, i.e.,

$$d(\mathbf{X}_n, \tilde{\mathbf{X}}_n) = \frac{1}{n} \sum_{k=1}^{n} d(x_k, \tilde{x}_k) \tag{3-4-3}$$

The source output is a random process, and, hence, the n samples in $\mathbf{X}_n$ are random variables. Therefore, $d(\mathbf{X}_n, \tilde{\mathbf{X}}_n)$ is a random variable. Its expected value is defined as the distortion D, i.e.,

$$D = E[d(\mathbf{X}_n, \tilde{\mathbf{X}}_n)] = \frac{1}{n} \sum_{k=1}^{n} E[d(x_k, \tilde{x}_k)] = E[d(x, \tilde{x})] \tag{3-4-4}$$

where the last step follows from the assumption that the source output process is stationary.

Now suppose we have a memoryless source with a continuous-amplitude output $\mathbf{X}$ that has a pdf $p(x)$, a quantized amplitude output alphabet $\tilde{\mathbf{X}}$, and a per letter distortion measure $d(x, \tilde{x})$, where $x \in \mathbf{X}$ and $\tilde{x} \in \tilde{\mathbf{X}}$. Then, the minimum rate in bits per source output that is required to represent the output $\mathbf{X}$ of the memoryless source with a distortion less than or equal to D is called the *rate-distortion function* $R(D)$ and is defined as

$$R(D) = \min_{p(\tilde{x}|x): E[d(\mathbf{X}, \tilde{\mathbf{X}})] \le D} I(\mathbf{X}, \tilde{\mathbf{X}}) \tag{3-4-5}$$

where $I(\mathbf{X}; \tilde{\mathbf{X}})$ is the average mutual information between $\mathbf{X}$ and $\tilde{\mathbf{X}}$. In general, the rate $R(D)$ decreases as D increases or, conversely, $R(D)$ increases as D decreases.

One interesting model of a continuous-amplitude, memoryless information source is the gaussian source model. In this case, Shannon proved the following fundamental theorem on the rate-distortion function.

Theorem: Rate-Distortion Function for a Memoryless Gaussian Source (Shannon, 1959a)

The minimum information rate necessary to represent the output of a discrete-time, continuous-amplitude memoryless gaussian source based on a mean-square-error distortion measure per symbol (single letter distortion measure) is

$$R_g(D) = \begin{cases} \frac{1}{2} \log_2 (\sigma_x^2 / D) & (0 \le D \le \sigma_x^2) \\ 0 & (D > \sigma_x^2) \end{cases} \tag{3-4-6}$$

where σ_x^2 is the variance of the gaussian source output.

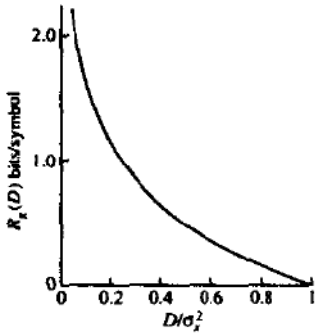


FIGURE 3-4-1 Rate distortion function for a continuous-amplitude memoryless gaussian source.

We should note that (3-4-6) implies that no information need be transmitted when the distortion $D \geq \sigma_x^2$. Specifically, $D = \sigma_x^2$ can be obtained by using zeros in the reconstruction of the signal. For $D > \sigma_x^2$, we can use statistically independent, zero-mean gaussian noise samples with a variance of $D - \sigma_x^2$ for the reconstruction. $R_g(D)$ is plotted in Fig. 3-4-1.

The rate distortion function $R(D)$ of a source is associated with the following basic source coding theorem in information theory.

Theorem: Source Coding with a Distortion Measure (Shannon, 1959a)

There exists an encoding scheme that maps the source output into code words such that for any given distortion D, the minimum rate $R(D)$ bits per symbol (sample) is sufficient to reconstruct the source output with an average distortion that is arbitrarily close to D.

It is clear, therefore, that the rate distortion function $R(D)$ for any source represents a lower bound on the source rate that is possible for a given level of distortion.

Let us return to the result in (3-4-6) for the rate distortion function of a memoryless gaussian source. If we reverse the functional dependence between D and R, we may express D in terms of R as

$$D_g(R) = 2^{-2R}\sigma_x^2 \tag{3-4-7}$$

This funcion is called the *distortion-rate function* for the discrete-time, memoryless gaussian source.

When we express the distortion in (3-4-7) in dB, we obtain

$$10 \log_{10} D_g(R) = -6R + 10 \log_{10} \sigma_x^2 \tag{3-4-8}$$

Note that the mean square distortion decreases at a rate of 6 dB/bit.

Explicit results on the rate distortion functions for memoryless non-gaussian sources are not available. However, there are useful upper and lower bounds

on the rate distortion function for any discrete-time, continuous-amplitude, memoryless source. An upper bound is given by the following theorem.

Theorem: Upper Bound on $R(D)$

The rate-distortion function of a memoryless, continuous-amplitude source with zero mean and finite variance σ_x^2 with respect to the mean-square-error distortion measure is upper bounded as

$$R(D) \leq \tfrac{1}{2} \log_2 \frac{\sigma_x^2}{D} \qquad (0 \leq D \leq \sigma_x^2) \tag{3-4-9}$$

A proof of this theorem is given by Berger (1971). It implies that the gaussian source requires the maximum rate among all other sources for a specified level of mean square distortion. Thus, the rate distortion $R(D)$ of any continuous-amplitude, memoryless source with zero mean and finite variance σ_x^2 satisfies the condition $R(D) \leq R_g(D)$. Similarly, the distortion-rate function of the same source satisfies the condition

$$D(R) \leq D_g(R) = 2^{-2R}\sigma_x^2 \tag{3-4-10}$$

A lower bound on the rate-distortion function also exists. This is called the *Shannon lower bound* for a mean-square-error distortion measure, and is given as

$$R^*(D) = H(X) - \tfrac{1}{2} \log_2 2\pi e D \tag{3-4-11}$$

where $H(X)$ is the differential entropy of the continuous-amplitude, memoryless source. The distortion-rate function corresponding to (3-4-11) is

$$D^*(R) = \frac{1}{2\pi e} 2^{-2[R - H(X)]} \tag{3-4-12}$$

Therefore, the rate-distortion function for any continuous-amplitude, memoryless source is bounded from above and below as

$$R^*(D) \leq R(D) \leq R_g(D) \tag{3-4-13}$$

and the corresponding distortion-rate function is bounded as

$$D^*(R) \leq D(R) \leq D_g(R) \tag{3-4-14}$$

The differential entropy of the memoryless gaussian source is

$$H_g(X) = \tfrac{1}{2} \log_2 2\pi e \sigma_x^2 \tag{3-4-15}$$

so that the lower bound $R^*(D)$ in (3-4-11) reduces to $R_g(D)$. Now, if we

express $D^*(R)$ in terms of decibels and normalize it by setting $\sigma_x^2 = 1$ [or dividing $D^*(R)$ by σ_x^2], we obtain from (3-4-12) TABLE 3-4-2

$$10 \log_{10} D^*(R) = -6R - 6[H_g(X) - H(X)] \tag{3-4-16}$$

or, equivalently,

$$10 \log_{10} \frac{D_g(R)}{D^*(R)} = 6[H_g(X) - H(X)] \text{ dB}$$
$$= 6[R_g(D) - R^*(D)] \text{ dB} \tag{3-4-17}$$

The relations in (3-4-16) and (3-4-17) allow us to compare the lower bound in the distortion with the upper bound which is the distortion for the gaussian source. We note that $D^*(R)$ also decreases at -6 dB/bit. We should also mention that the differential entropy $H(X)$ is upper-bounded by $H_g(X)$, as shown by Shannon (1948b).

Table 3-4-1 lists four pdfs that are models commonly used for source signal distributions. The table shows the differential entropies, the differences in rates in bits/sample, and the difference in distortion between the upper and lower bounds. Note that the gamma pdf shows the greatest deviation from the gaussian. The Laplacian pdf is the most similar to the gaussian, and the uniform pdf ranks second of the pdfs shown in the table. These results provide some benchmarks on the difference between the upper and lower bounds on distortion and rate.

Before concluding this section, let us consider a band-limited gaussian source with spectral density

$$\Phi(f) = \begin{cases} \sigma_x^2/2W & (|f| \leq W) \\ 0 & (|f| > W) \end{cases} \tag{3-4-18}$$

When the output of this source is sampled at the Nyquist rate, the samples are uncorrelated and, since the source is gaussian, they are also statistically

TABLE 3-4-1 DIFFERENTIAL ENTROPIES AND RATE DISTORTION COMPARISONS OF FOUR COMMON PDFs FOR SIGNAL MODELS

pdf	$p(x)$	$H(X)$	$R_g(D) - R^*(D)$ (bits/sample)	$D_g(R) - D^*(R)$ (dB)
Gaussian	$\frac{1}{\sqrt{2\pi}\,\sigma_x} e^{-x^2/2\sigma_x^2}$	$\frac{1}{2}\log_2(2\pi e\sigma_x^2)$	0	0
Uniform	$\frac{1}{2\sqrt{3}\,\sigma_x}, \lvert x\rvert \leq \sqrt{3}\,\sigma_x$	$\frac{1}{2}\log_2(12\sigma_x^2)$	0.255	1.53
Laplacian	$\frac{1}{\sqrt{2}\,\sigma_x} e^{-\sqrt{2}\lvert x\rvert/\sigma_x}$	$\frac{1}{2}\log_2(2e^2\sigma_x^2)$	0.104	0.62
Gamma	$\frac{\sqrt[4]{3}}{\sqrt{8\pi\sigma_x\lvert x\rvert}} e^{-\sqrt{3}\lvert x\rvert/2\sigma_x}$	$\frac{1}{2}\log_2(4\pi e^{0.423}\sigma_x^2/3)$	0.709	4.25

independent. Hence, the equivalent discrete-time gaussian source is memoryless. The rate-distortion function for each sample is given by (3-4-6). Therefore, the rate-distortion function for the band-limited white gaussian source in bits/s is

$$R_g(D) = W \log_2 \frac{\sigma_x^2}{D} \quad (0 \leq D \leq \sigma_x^2) \tag{3-4-19}$$

The corresponding distortion-rate function is

$$D_g(R) = 2^{-R/W}\sigma_x^2 \tag{3-4-20}$$

which, when expressed in decibels and normalized by σ_x^2, becomes

$$10 \log D_g(R)/\sigma_x^2 = -3R/W \tag{3-4-21}$$

The more general case in which the gaussian process is neither white nor band-limited has been treated by Gallager (1968) and Goblick and Holsinger (1967).

3-4-2 Scalar Quantization

In source encoding, the quantizer can be optimized if we know the probability density function of the signal amplitude at the input to the quantizer. For example, suppose that the sequence $\{x_n\}$ at the input to the quantizer has a pdf $p(x)$ and let $L = 2^R$ be the desired number of levels. We wish to design the optimum scalar quantizer that minimizes some function of the quantization error $q = \tilde{x} - x$, where $\tilde{x}$ is the quantized value of x. To elaborate, suppose that $f(\tilde{x} - x)$ denotes the desired function of the error. Then, the distortion resulting from quantization of the signal amplitude is

$$D = \int_{-\infty}^{\infty} f(\tilde{x} - x)p(x)\,dx \tag{3-4-22}$$

In general, an optimum quantizer is one that minimizes D by optimally selecting the output levels and the corresponding input range of each output level. This optimization problem has been considered by Lloyd (1982) and Max (1960), and the resulting optimum quantizer is usually called the *Lloyd–Max quantizer.*

For a uniform quantizer, the output levels are specified as $\tilde{x}_k = \frac{1}{2}(2k-1)\Delta$, corresponding to an input signal amplitude in the range $(k-1)\Delta \leq x < k\Delta$, where Δ is the step size. When the uniform quantizer is symmetric with an even number of levels, the average distortion in (3-4-22) may be expressed as

$$D = 2\sum_{k=1}^{L/2-1} \int_{(k-1)\Delta}^{k\Delta} f(\tfrac{1}{2}(2k-1)\Delta - x)p(x)\,dx + 2\int_{(L/2-1)\Delta}^{\infty} f(\tfrac{1}{2}(2k-1)\Delta - x)p(x)\,dx \tag{3-4-23}$$

TABLE 3-4-2 OPTIMUM STEP SIZES FOR UNIFORM QUANTIZATION OF A GAUSSIAN RANDOM VARIABLE

Number of output levels	Optimum step size Δ_{opt}	Minimum MSE D_{min}	$10 \log D_{min}$ (dB)
2	1.596	0.3634	−4.4
4	0.9957	0.1188	−9.25
8	0.5860	0.03744	−14.27
16	0.3352	0.01154	−19.38
32	0.1881	0.00349	−24.57

In this case, the minimization of D is carried out with respect to the step-size parameter Δ. By differentiating D with respect to Δ, we obtain

$$\sum_{k=1}^{L/2-1}(2k-1)\int_{(k-1)\Delta}^{k\Delta} f(\tfrac{1}{2}(2k-1)\Delta - x)p(x)\,dx + (L-1)\int_{-(L/2-1)\Delta}^{\infty} f'(\tfrac{1}{2}(L-1)\Delta - x)p(x)\,dx = 0 \quad (3\text{-}4\text{-}24)$$

where $f'(x)$ denotes the derivative of $f(x)$.

By selecting the error criterion function $f(x)$, the solution of (3-4-24) for the optimum step size can be obtained numerically on a digital computer for any given pdf $p(x)$. For the mean-square-error criterion, for which $f(x) = x^2$, Max (1960) evaluated the optimum step size Δ_{opt} and the minimum mean square error when the pdf $p(x)$ is zero-mean gaussian with unit variance. Some of these results are given in Table 3-4-2. We observe that the minimum mean square distortion D_{min} decreases by a little more than 5 dB for each doubling of the number of levels L. Hence, each additional bit that is employed in a uniform quantizer with optimum step size Δ_{opt} for a gaussian-distributed signal amplitude reduces the distortion by more than 5 dB.

By relaxing the constraint that the quantizer be uniform, the distortion can be reduced further. In this case, we let the output level be $\tilde{x} = \tilde{x}_k$ when the input signal amplitude is in the range $x_{k-1} \leq x < x_k$. For an L-level quantizer, the end points are $x_0 = -\infty$ and $x_L = \infty$. The resulting distortion is

$$D = \sum_{k=1}^{L}\int_{x_{k-1}}^{x_k} f(\tilde{x}_k - x)p(x)\,dx \quad (3\text{-}4\text{-}25)$$

which is now minimized by optimally selecting the $\{\tilde{x}_k\}$ and $\{x_k\}$.

The necessary conditions for a minimum distortion are obtained by differentiating D with respect to the $\{x_k\}$ and $\{\tilde{x}_k\}$. The result of this minimization is the pair of equations

$$f(\tilde{x}_k - x_k) = f(\tilde{x}_{k+1} - x_k), \quad k = 1, 2, \ldots, L-1 \quad (3\text{-}4\text{-}26)$$

$$\int_{x_{k-1}}^{x_k} f'(\tilde{x}_k - x)p(x)\,dx = 0, \quad k = 1, 2, \ldots, L \quad (3\text{-}4\text{-}27)$$

TABLE 3-4-3 OPTIMUM FOUR-LEVEL QUANTIZER FOR A GAUSSIAN RANDOM VARIABLE

Level k	x_k	$\tilde{x}_k$
1	−0.9816	−1.510
2	0.0	−0.4528
3	0.9816	0.4528
4	∞	1.510

$D_{\min} = 0.1175$
$10 \log D_{\min} = -9.3$ dB

As a special case, we again consider minimizing the mean square value of the distortion. In this case, $f(x) = x^2$ and, hence, (3-4-26) becomes

$$x_k = \tfrac{1}{2}(\tilde{x}_k + \tilde{x}_{k+1}), \qquad k = 1, 2, \ldots, L-1 \tag{3-4-28}$$

which is the midpoint between $\tilde{x}_k$ and $\tilde{x}_{k+1}$. The corresponding equations determining $\{\tilde{x}_k\}$ are

$$\int_{x_{k-1}}^{x_k} (\tilde{x}_k - x)p(x)\,dx = 0, \qquad k = 1, 2, \ldots, L \tag{3-4-29}$$

Thus, $\tilde{x}_k$ is the centroid of the area of $p(x)$ between x_{k-1} and x_k. These equations may be solved numerically for any given $p(x)$.

Tables 3-4-3 and 3-4-4 give the results of this optimization obtained by Max

TABLE 3-4-4 OPTIMUM EIGHT-LEVEL QUANTIZER FOR A GAUSSIAN RANDOM VARIABLE (MAX, 1960)

Level k	x_k	$\tilde{x}_k$
1	−1.748	−2.152
2	−1.050	−1.344
3	−0.5006	−0.7560
4	0	−0.2451
5	0.5006	0.2451
6	1.050	0.7560
7	1.748	1344
8	∞	2.152

$D_{\min} = 0.03454$
$10 \log D_{\min} = -14.62$ dB

TABLE 3-4-5 COMPARISON OF OPTIMUM UNIFORM AND NONUNIFORM QUANTIZERS FOR A GAUSSIAN RANDOM VARIABLE (MAX, 1960; PAEZ AND GLISSON, 1972)

R (bits/sample)	$10\log_{10} D_{min}$	
	Uniform (dB)	Nonuniform (dB)
1	−4.4	−4.4
2	−9.25	−9.30
3	−14.27	−14.62
4	−19.38	−20.22
5	−24.57	−26.02
6	−29.83	−31.89
7	−35.13	−37.81

(1960) for the optimum four-level and eight-level quantizers of a gaussian distributed signal amplitude having zero mean and unit variance. In Table 3-4-5, we compare the minimum mean square distortion of a uniform quantizer to that of a nonuniform quantizer for the gaussian-distributed signal amplitude. From the results of this table, we observe that the difference in the performance of the two types of quantizers is relatively small for small values of R (less than 0.5 dB for $R \leq 3$), but it increases as R increases. For example, at $R = 5$, the nonuniform quantizer is approximately 1.5 dB better than the uniform quantizer.

It is instructive to plot the minimum distortion as a function of the bit rate $R = \log_2 L$ bits per source sample (letter) for both the uniform and nonuniform quantizers. These curves are illustrated in Fig. 3-4-2. The functional dependence of the distortion D on the bit rate R may be expressed as $D(R)$, the distortion-rate function. We observe that the distortion-rate function for the optimum nonuniform quantizer falls below that of the optimum uniform quantizer.

Since any quantizer reduces a continuous amplitude source into a discrete amplitude source, we may treat the discrete amplitude as letters, say $\tilde{X} = \{\tilde{x}_k, 1 \leq k \leq L\}$, with associated probabilities $\{p_k\}$. If the signal amplitudes are statistically independent, the discrete source is memoryless and, hence, its entropy is

$$H(\tilde{X}) = -\sum_{k=1}^{L} p_k \log_2 p_k \tag{3-4-30}$$

For example, the optimum four-level nonuniform quantizer for the gaussian-distributed signal amplitude results in the probabilities $p_1 = p_4 = 0.1635$ for the two outer levels and $p_2 = p_3 = 0.3365$ for the two inner levels. The entropy for the discrete source is $H(\tilde{X}) = 1.911$ bits/letter. Hence, with

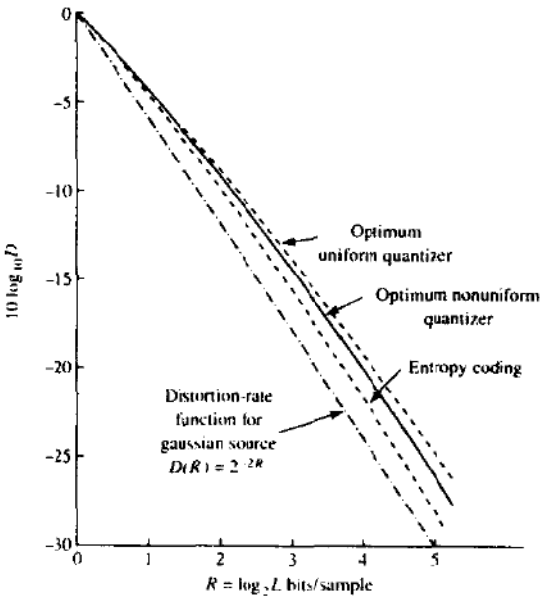


FIGURE 3-4-2 Distortion versus rate curves for discrete-time memoryless gaussian source.

entropy coding (Huffman coding) of blocks of output letters, we can achieve the minimum distortion of -9.30 dB with 1.911 bits/letter instead of 2 bits/letter. Max (1960) has given the entropy for the discrete source letters resulting from quantization. Table 3-4-6 lists the values of the entropy for the nonuniform quantizer. These values are also plotted in Fig. 3-4-2 and labeled *entropy coding*.

From this discussion, we conclude that the quantizer can be optimized when the pdf of the continuous source output is known. The optimum quantizer of $L = 2^R$ levels results in a minimum distortion of $D(R)$, where $R = \log_2 L$

TABLE 3-4-6 ENTROPY OF THE OUTPUT OF AN OPTIMUM NONUNIFORM QUANTIZER FOR A GAUSSIAN RANDOM VARIABLE (MAX, 1960)

R (bits/sample)	Entropy (bits/letter)	Distortion $10 \log_{10} D_{min}$
1	1.0	−4.4
2	1.911	−9.30
3	2.825	−14.62
4	3.765	−20.22
5	4.730	−26.02

bits/sample. Thus, this distortion can be achieved by simply representing each quantized sample by R bits. However, more efficient encoding is possible. The discrete source output that results from quantization is characterized by a set of probabilities $\{p_k\}$ that can be used to design efficient variable-length codes for the source output (entropy coding). The efficiency of any encoding method can be compared with the distortion-rate function or, equivalently, the rate-distortion function for the discrete-time, continuous-amplitude source that is characterized by the given pdf.

If we compare the performance of the optimum nonuniform quantizer with the distortion-rate function, we find, for example, that at a distortion of -26 dB, entropy coding is 0.41 bits/sample more than the minimum rate given by (3-4-8), and simple block coding of each letter requires 0.68 bits/sample more than the minimum rate. We also observe that the distortion rate functions for the optimal uniform and nonuniform quantizers for the gaussian source approach the slope of -6 dB/bit asymptotically for large R.

3-4-3 Vector Quantization

In the previous section, we considered the quantization of the output signal from a continuous-amplitude source when the quantization is performed on a sample-by-sample basis, i.e., by scalar quantization. In this section, we consider the joint quantization of a block of signal samples or a block of signal parameters. This type of quantization is called *block* or *vector quantization*. It is widely used in speech coding for digital cellular systems.

A fundamental result of rate-distortion theory is that better performance can be achieved by quantizing vectors instead of scalars, even if the continuous-amplitude source is memoryless. If, in addition, the signal samples or signal parameters are statistically dependent, we can exploit the dependency by jointly quantizing blocks of samples or parameters and, thus, achieve an even greater efficiency (lower bit rate) compared with that which is achieved by scalar quantization.

The vector quantization problem may be formulated as follows. We have an n-dimensional vector $\mathbf{X} = [x_1\ x_2\ \cdots\ x_n]$ with real-valued, continuous-amplitude components $\{x_k, 1 \leq k \leq n\}$ that are described by a joint pdf $p(x_1, x_2, \ldots, x_n)$. The vector $\mathbf{X}$ is quantized into another n-dimensional vector $\tilde{\mathbf{X}}$ with components $\{\tilde{x}_k, 1 \leq k \leq n\}$. We express the quantization as $Q(\cdot)$, so that

$$\tilde{\mathbf{X}} = Q(\mathbf{X}) \tag{3-4-31}$$

where $\tilde{\mathbf{X}}$ is the output of the vector quantizer when the input vector is $\mathbf{X}$.

Basically, vector quantization of blocks of data may be viewed as a pattern recognition problem involving the classification of blocks of data into a discrete number of categories or *cells* in a way that optimizes some fidelity criterion, such as mean square distortion. For example, let us consider the quantization

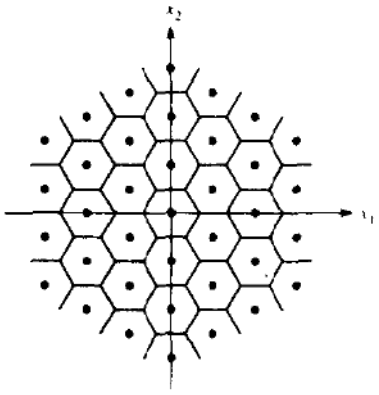


FIGURE 3-4-3 An example of quantization in two-dimensional space.

of two-dimensional vectors $\mathbf{X} = [x_1\ x_2]$. The two-dimensional space is partitioned into cells as illustrated in Fig. 3-4-3, where we have arbitrarily selected hexagonal-shaped cells $\{C_k\}$. All input vectors that fall in cell C_k are quantized into the vector $\tilde{\mathbf{X}}_k$, which is shown in Fig. 3-4-3 as the center of the hexagon. In this example, there are $L = 37$ vectors, one for each of the 37 cells into which the two-dimensional space has been partitioned. We denote the set of possible output vectors as $\{\tilde{\mathbf{X}}_k,\ 1 \leq k \leq L\}$.

In general, quantization of the n-dimensional vector $\mathbf{X}$ into an n-dimensional vector $\tilde{\mathbf{X}}$ introduces a quantization error or a distortion $d(\mathbf{X}, \tilde{\mathbf{X}})$. The average distortion over the set of input vectors $\mathbf{X}$ is

$$\begin{aligned} D &= \sum_{k=1}^{L} P(\mathbf{X} \in C_k) E[d(\mathbf{X}, \tilde{\mathbf{X}}_k) \mid \mathbf{X} \in C_k] \\ &= \sum_{k=1}^{L} P(\mathbf{X} \in C_k) \int_{\mathbf{X} \in C_k} d(\mathbf{X}, \tilde{\mathbf{X}}_k) p(\mathbf{X})\, d\mathbf{X} \end{aligned} \tag{3-4-32}$$

where $P(\mathbf{X} \in C_k)$ is the probability that the vector $\mathbf{X}$ falls in the cell C_k and $p(\mathbf{X})$ is the joint pdf of the n random variables. As in the case of scalar quantization, we can minimize D by selecting the cells $\{C_k,\ 1 \leq k \leq L\}$ for a given pdf $p(\mathbf{X})$.

A commonly used distortion measure is the mean square error (l_2 norm) defined as

$$d_2(\mathbf{X}, \tilde{\mathbf{X}}) = \frac{1}{n}(\mathbf{X} - \tilde{\mathbf{X}})'(\mathbf{X} - \tilde{\mathbf{X}}) = \frac{1}{n}\sum_{k=1}^{n} (x_k - \tilde{x}_k)^2 \tag{3-4-33}$$

or, more generally, the weighted mean square error

$$d_{2W}(\mathbf{X}, \tilde{\mathbf{X}}) = (\mathbf{X} - \tilde{\mathbf{X}})'\mathbf{W}(\mathbf{X} - \tilde{\mathbf{X}}) \tag{3-4-34}$$

where $\mathbf{W}$ is a positive-definite weighting matrix. Usually, $\mathbf{W}$ is selected to be the inverse of the covariance matrix of the input data vector $\mathbf{X}$.

Other distortion measures that are sometimes used are special cases of the l_p norm defined as

$$d_p(\mathbf{X}, \tilde{\mathbf{X}}) = \frac{1}{n} \sum_{k=1}^{n} |x_k - \tilde{x}_k|^p \tag{3-4-35}$$

The special case $p = 1$ is often used as an alternative to $p = 2$.

Vector quantization is not limited to quantizing a block of signal samples of a source waveform. It can also be applied to quantizing a set of parameters extracted from the data. For example, in linear predictive coding (LPC), described in Section 3-5-3, the parameters extracted from the signal are the prediction coefficients, which are the coefficients in the all-pole filter model for the source that generates the observed data. These parameters can be considered as a block and quantized as a block by application of some appropriate distortion measure. In the case of speech encoding, an appropriate distortion measure, proposed by Itakura and Saito (1968, 1975), is the weighted square error where the weighting matrix $\mathbf{W}$ is selected to be the normalized autocorrelation matrix $\mathbf{\Phi}$ of the observed data.

In speech processing, an alternative set of parameters that may be quantized as a block and transmitted to the receiver is the set of reflection coefficients $\{a_{ii}, 1 \leq i \leq m\}$. Yet another set of parameters that is sometimes used for vector quantization in linear predictive coding of speech comprises the log-area ratios $\{r_k\}$, which are defined in terms of the reflection coefficients as

$$r_k = \log \frac{1 + a_{kk}}{1 - a_{kk}}, \quad 1 \leq k \leq m \tag{3-4-36}$$

Now, let us return to the mathematical formulation of vector quantization and let us consider the partitioning of the n-dimensional space into L cells $\{C_k, 1 \leq k \leq L\}$ so that the average distortion is minimized over all L-level quantizers. There are two conditions for optimality. The first is that the optimal quantizer employs a nearest-neighbor selection rule, which may be expressed mathematically as

$$Q(\mathbf{X}) = \mathbf{X}_k$$

if and only if

$$D(\mathbf{X}, \tilde{\mathbf{X}}_k) \leq D(\mathbf{X}, \tilde{\mathbf{X}}_j), \quad k \neq j, \quad 1 \leq j \leq L \tag{3-4-37}$$

The second condition necessary for optimality is that each output vector $\tilde{\mathbf{X}}_k$ be chosen to minimize the average distortion in cell C_k. In other words, $\tilde{\mathbf{X}}_k$ is the vector in C_k that minimizes

$$D_k = E[d(\mathbf{X}, \tilde{\mathbf{X}}) \mid \mathbf{X} \in C_k] = \int_{\mathbf{X} \in C_k} d(\mathbf{X}, \tilde{\mathbf{X}}) p(\mathbf{X})\, d\mathbf{X} \tag{3-4-38}$$

The vector $\tilde{\mathbf{X}}_k$ that minimizes D_k is called the *centroid* of the cell. Thus, these conditions for optimality can be applied to partition the n-dimensional space

into cells $\{C_k, 1 \leq k \leq L\}$ when the joint pdf $p(\mathbf{X})$ is known. It is clear that these two conditions represent the generalization of the optimum scalar quantization problem to the n-dimensional vector quantization problem. In general, we expect the code vectors to be closer together in regions where the joint pdf is large and farther apart in regions where $p(\mathbf{X})$ is small.

As an upper bound on the distortion of a vector quantizer, we may use the distortion of the optimal scalar quantizer, which can be applied to each component of the vector as described in the previous section. On the other hand, the best performance that can be achieved by optimum vector quantization is given by the rate-distortion function or, equivalently, the distortion-rate function.

The distortion-rate function, which was introduced in the previous section, may be defined in the context of vector quantization as follows. Suppose we form a vector $\mathbf{X}$ of dimension n from n consecutive samples $\{x_m\}$. The vector $\mathbf{X}$ is then quantized to form $\tilde{\mathbf{X}} = Q(\mathbf{X})$, where $\tilde{\mathbf{X}}$ is a vector from the set of $\{\tilde{\mathbf{X}}_k, 1 \leq k \leq L\}$. As described above, the average distortion D resulting from representing $\mathbf{X}$ by $\tilde{\mathbf{X}}$ is $E[d(\mathbf{X}, \tilde{\mathbf{X}})]$, where $d(\mathbf{X}, \tilde{\mathbf{X}})$ is the distortion per dimension, e.g.,

$$d(\mathbf{X}, \tilde{\mathbf{X}}) = \frac{1}{n} \sum_{k=1}^{n} (x_k - \tilde{x}_k)^2$$

The vectors $\{\tilde{\mathbf{X}}_k, 1 \leq k \leq L\}$ can be transmitted at an average bit rate of

$$R = \frac{H(\tilde{\mathbf{X}})}{n} \text{ bits/sample} \tag{3-4-39}$$

where $H(\tilde{\mathbf{X}})$ is the entropy of the quantized source output defined as

$$H(\tilde{\mathbf{X}}) = -\sum_{i=1}^{L} p(\tilde{\mathbf{X}}_i) \log_2 P(\tilde{\mathbf{X}}_i) \tag{3-4-40}$$

For a given average rate R, the minimum achievable distortion $D_n(R)$ is

$$D_n(R) = \min_{Q(\mathbf{X})} E[d(\mathbf{X}, \tilde{\mathbf{X}})] \tag{3-4-41}$$

where $R \geq H(\tilde{\mathbf{X}})/n$ and the minimum in (3-4-41) is taken over all possible mappings $Q(\mathbf{X})$. In the limit as the number of dimensions n is allowed to approach infinity, we obtain

$$D(R) = \lim_{n \to \infty} D_n(R) \tag{3-4-42}$$

where $D(R)$ is the distortion-rate function that was introduced in the previous section. It is apparent from this development that the distortion-rate function can be approached arbitrarily closely by increasing the size n of the vectors.

The development above is predicated on the assumption that the joint pdf $p(\mathbf{X})$ of the data vector is known. However, in practice, the joint pdf $p(\mathbf{X})$ of the data may not be known. In such a case, it is possible to select the

quantized output vectors adaptively from a set of training vectors $\mathbf{X}(m)$. Specifically, suppose that we are given a set of M training vectors where M is much greater than L ($M \gg L$). An iterative clustering algorithm, called the *K means algorithm*, where in our case $K = L$, can be applied to the training vectors. This algorithm iteratively subdivides the M training vectors into L clusters such that the two necessary conditions for optimality are satisfied. The K means algorithm may be described as follows [Makhoul *et al.* (1985)].

K Means Algorithm

Step 1 Initialize by setting the iteration number $i = 0$. Choose a set of output vectors $\tilde{\mathbf{X}}_k(0)$, $1 \leq k \leq L$.

Step 2 Classify the training vectors $\{\mathbf{X}(m),\ 1 \leq m \leq M\}$ into the clusters $\{C_k\}$ by applying the nearest-neighbor rule

$$\mathbf{X} \in C_k(i) \quad \text{iff } D(\mathbf{X}, \tilde{\mathbf{X}}_k(i)) \leq D(\mathbf{X}, \tilde{\mathbf{X}}_j(i)) \quad \text{for all } k \neq j$$

Step 3 Recompute (set i to $i+1$) the output vectors of every cluster by computing the centroid

$$\tilde{\mathbf{X}}_k(i) = \frac{1}{M_k} \sum_{\mathbf{X} \in C_k} \mathbf{X}(m), \qquad 1 \leq k \leq L$$

of the training vectors that fall in each cluster. Also, compute the resulting distortion $D(i)$ at the ith iteration.

Step 4 Terminate the test if the change $D(i-1) - D(i)$ in the average distortion is relatively small. Otherwise, go to Step 2.

The K means algorithm converges to a local minimum (see Anderberg, 1973; Linde *et al.*, 1980). By beginning the algorithm with different sets of initial output vectors $\{\mathbf{X}_k(0)\}$ and each time performing the optimization described in the K means algorithm, it is possible to find a global optimum. However, the computational burden of this search procedure may limit the search to a few initializations.

Once we have selected the output vectors $\{\tilde{\mathbf{X}}_k, 1 \leq k \leq L\}$, each signal vector $\mathbf{X}(m)$ is quantized to the output vector that is nearest to it according to the distortion measure that is adopted. If the computation involves evaluating the distance between $\mathbf{X}(m)$ and each of the L possible output vectors $\{\tilde{\mathbf{X}}_k\}$, the procedure constitutes a *full search*. If we assume that each computation requires n multiplications and additions, the computational requirement for a full search is

$$\mathscr{C} = nL \tag{3-4-43}$$

multiplication and additions per input vector.

If we select L to be a power of 2 then $\log_2 L$ is the number of bits required to represent each vector. Now, if R denotes the bit rate per sample [per component or dimension of $\mathbf{X}(m)$], we have $nR = \log_2 L$, and, hence, the computational cost is

$$\mathscr{C} = n2^{nR} \tag{3-4-44}$$

Note that the number of computations grows exponentially with the dimensionality parameter n and the bit rate R per dimension. Because of this exponential increase of the computational cost, vector quantization has been applied to low-bit-source encoding, such as coding the reflection coefficients or log area ratios in LPC.

The computational cost associated with full search can be reduced by slightly suboptimum algorithms (see Chang *et al.*, 1984; Gersho, 1982).

In order to demonstrate the benefits of vector quantization compared with scalar quantization, we present the following example taken from Makhoul *et al.* (1985).

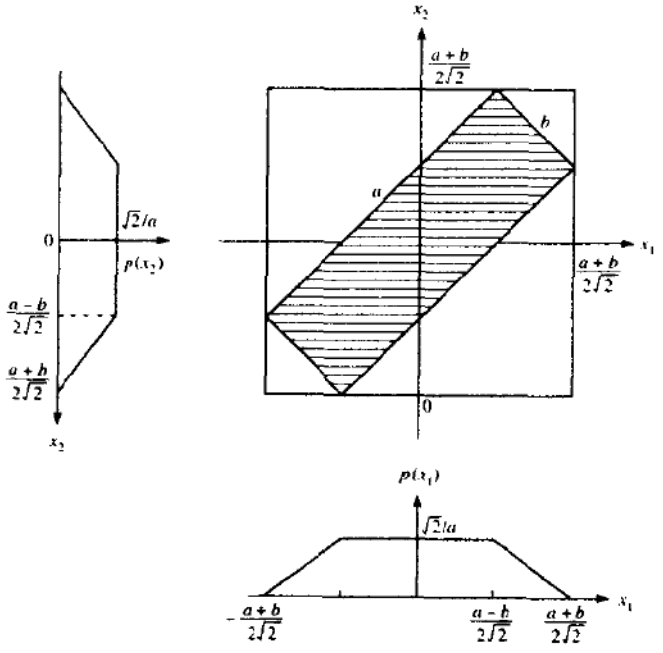
Example 3-4-1

Let x_1 and x_2 be two random variables with a uniform joint pdf

$$p(x_1, x_2) \equiv p(\mathbf{X}) = \begin{cases} \dfrac{1}{ab} & (\mathbf{X} \in \mathbf{C}) \\ 0 & (\text{otherwise}) \end{cases} \tag{3-4-45}$$

where $\mathbf{C}$ is the rectangular region illustrated in Fig. 3-4-4. Note that the rectangle is rotated by 45° relative to the horizontal axis. Also shown in Fig. 3-4-4 are the marginal densities $p(x_1)$ and $p(x_2)$.

FIGURE 3-4-4 A uniform pdf in two dimensions. (*Makhoul* et al., *1985.*)



If we quantize x_1 and x_2 separately by using uniform intervals of length Δ, the number of levels needed is

$$L_1 = L_2 = \frac{a+b}{\sqrt{2}\,\Delta} \tag{3-4-46}$$

Hence, the number of bits needed for coding the vector $\mathbf{X} = [x_1 \;\; x_2]$ is

$$R_x = R_1 + R_2 = \log_2 L_1 + \log_2 L_2$$

$$R_x = \log_2 \frac{(a+b)^2}{2\Delta^2} \tag{3-4-47}$$

Thus, scalar quantization of each component is equivalent to vector quantization with the total number of levels

$$L_x = L_1 L_2 = \frac{(a+b)^2}{2\Delta^2} \tag{3-4-48}$$

We observe that this approach is equivalent to covering the large square that encloses the rectangle by square cells, where each cell represents one of the L_x quantized regions. Since $p(\mathbf{X}) = 0$ except for $\mathbf{X} \in C$, this encoding is wasteful and results in an increase of the bit rate.

If we were to cover only the region for which $p(\mathbf{X}) \neq 0$ with squares having area Δ^2, the total number of levels that will result is the area of the rectangle divided by Δ^2, i.e.,

$$L_x' = \frac{ab}{\Delta^2} \tag{3-4-49}$$

Therefore, the difference in bit rate between the scalar and vector quantization methods is

$$R_x - R_x' = \log_2 \frac{(a+b)^2}{2ab} \tag{3-4-50}$$

For instance, if $a = 4b$, the difference in bit rate is

$$R_x - R_x' = 1.64 \text{ bits/vector}$$

Thus, vector quantization is 0.82 bits/sample better for the same distortion.

It is interesting to note that a linear transformation (rotation by 45°) will decorrelate x_1 and x_2 and render the two random variables statistically independent. Then scalar quantization and vector quantization achieve the same efficiency. Although a linear transformation can decorrelate a vector of random variables, it does not result in statistically independent random variables, in general. Consequently, vector quantization will always equal or exceed the performance of scalar quantization (see Problem 3-40).

Vector quantization has been applied to several types of speech encoding

methods including both waveform and model-based methods which are treated in Section 3-5. In model-based methods such as LPC, vector quantization has made possible the coding of speech at rates below 1000 bits/s (see Buzo *et al.*, 1980; Roucos *et al.*, 1982; Paul 1983). When applied to waveform encoding methods, it is possible to obtain good quality speech at 16 000 bits/s, or, equivalently, at $R = 2$ bits/sample. With additional computational complexity, it may be possible in the future to implement waveform encoders producing good quality speech at a rate of $R = 1$ bit/sample.

3-5 CODING TECHNIQUES FOR ANALOG SOURCES

A number of coding techniques for analog sources have been developed over the past 40 years. Most of these have been applied to the encoding of speech and images. In this section, we briefly describe several of these methods and use speech encoding as an example in assessing their performance.

It is convenient to subdivide analog source encoding methods into three types. One type is called *temporal waveform coding*. In this type of encoding, the source encoder is designed to represent digitally the temporal characteristics of the source waveform. A second type of source encoding is *spectral waveform coding*. The signal waveform is usually subdivided into different frequency bands, and either the time waveform in each band or its spectral characteristics are encoded for transmission. The third type of source encoding is based on a mathematical model of the source and is called *model-based coding*.

3-5-1 Temporal Waveform Coding

There are several analog source coding techniques that are designed to represent the time-domain characteristics of the signal. The most commonly used methods are described in this section.

Pulse Code Modulation† (PCM) Let $x(t)$ denote a sample function emitted by a source and let x_n denote the samples taken at a sampling rate $f_s \geq 2W$, where W is the highest frequency in the spectrum of $x(t)$. In PCM, each sample of the signal is quantized to one of 2^R amplitude levels, where R is the number of binary digits used to represent each sample. Thus the rate from the source is Rf_s bits/s.

The quantization process may be modeled mathematically as

$$\tilde{x}_n = x_n + q_n \tag{3-5-1}$$

where $\tilde{x}_n$ represents the quantized value of x_n and q_n represents the quantization error, which we treat as an additive noise. Assuming that a

† PCM, DPCM, and ADPCM are source coding techniques. They are not digital modulation methods.

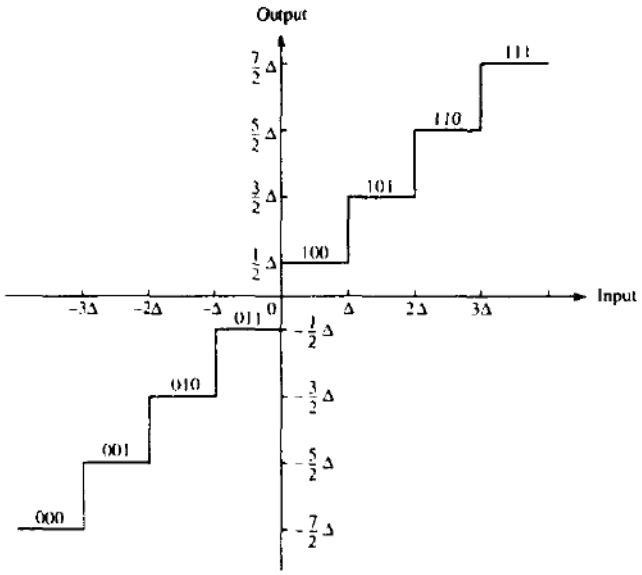


FIGURE 3-5-1 Input–output characteristic for a uniform quantizer.

uniform quantizer is used, having the input–output characteristic illustrated in Fig. 3-5-1, the quantization noise is well characterized statistically by the uniform pdf

$$p(q) = \frac{1}{\Delta}, \qquad -\tfrac{1}{2}\Delta \le q \le \tfrac{1}{2}\Delta \tag{3-5-2}$$

where the step size of the quantizer is $\Delta = 2^{-R}$. The mean square value of the quantization error is

$$E(q^2) = \tfrac{1}{12}\Delta^2 = \tfrac{1}{12} \times 2^{-2R} \tag{3-5-3}$$

Measured in decibels, the mean square value of the noise is

$$10 \log \tfrac{1}{12}\Delta^2 = 10 \log (\tfrac{1}{12} \times 2^{-2R}) = -6R - 10.8 \text{ dB} \tag{3-5-4}$$

We observe that the quantization noise decreases by 6 dB/bit used in the quantizer. For example, a 7 bit quantizer results in a quantization noise power of −52.8 dB.

Many source signals such as speech waveforms have the characteristic that small signal amplitudes occur more frequently than large ones. However, a uniform quantizer provides the same spacing between successive levels throughout the entire dynamic range of the signal. A better approach is to employ a nonuniform quantizer. A nonuniform quantizer characteristic is usually obtained by passing the signal through a nonlinear device that compresses the signal amplitude, followed by a uniform quantizer. For

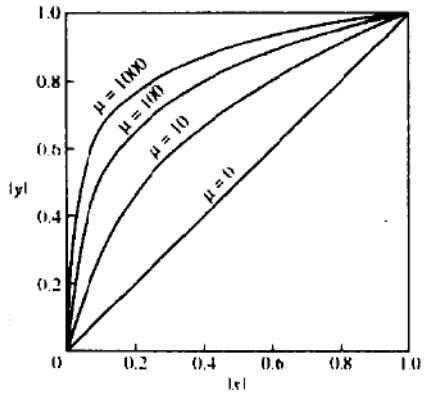


FIGURE 3-5-2 Input–output magnitude characteristic for a logarithmic compressor.

example, a logarithmic compressor has an input–output magnitude characteristics of the form

$$|y| = \frac{\log(1 + \mu |x|)}{\log(1 + \mu)} \tag{3-5-5}$$

where $|x| \leq 1$ is the magnitude of the input, $|y|$ is the magnitude of the output, and μ is a parameter that is selected to give the desired compression characteristic. Figure 3-5-2 illustrates this compression relationship for several values of μ. The value $\mu = 0$ corresponds to no compression.

In the encoding of speech waveforms, for example, the value of $\mu = 255$ has been adopted as a standard in the USA and Canada. This value results in about a 24 dB reduction in the quantization noise power relative to uniform quantization, as shown by Jayant (1974). Consequently, a 7 bit quantizer used in conjunction with a $\mu = 255$ logarithmic compressor produces a quantization noise power of approximately -77 dB compared with the -53 dB for uniform quantization.

In the reconstruction of the signal from the quantized values, the inverse logarithmic relation is used to expand the signal amplitude. The combined compressor–expandor pair is termed a *compandor*.

Differential Pulse Code Modulation (DPCM) In PCM, each sample of the waveform is encoded independently of all the others. However, most source signals sampled at the Nyquist rate or faster exhibit significant correlation between successive samples. In other words, the average change in amplitude between successive samples is relatively small. Consequently, an encoding scheme that exploits the redundancy in the samples will result in a lower bit rate for the source output.

A relatively simple solution is to encode the differences between successive samples rather than the samples themselves. Since differences between samples are expected to be smaller than the actual sampled amplitudes, fewer bits are required to represent the differences. A refinement of this general approach is

to predict the current sample based on the previous p samples. To be specific, let x_n denote the current sample from the source and let $\hat{x}_n$ denote the predicted value of x_n, defined as

$$\hat{x}_n = \sum_{i=1}^{p} a_i x_{n-i} \tag{3-5-6}$$

Thus $\hat{x}_n$ is a weighted linear combination of the past p samples and the $\{a_i\}$ are the predictor coefficients. The $\{a_i\}$ are selected to minimize some function of the error between x_n and $\hat{x}_n$.

A mathematically and practically convenient error function is the mean square error (MSE). With the MSE as the performance index for the predictor, we select the $\{a_i\}$ to minimize

$$\begin{aligned}\mathscr{E}_p = E(e_n^2) &= E\left[\left(x_n - \sum_{i=1}^{p} a_i x_{n-i}\right)^2\right] \\ &= E(x_n^2) - 2\sum_{i=1}^{p} a_i E(x_n x_{n-i}) + \sum_{i=1}^{p}\sum_{j=1}^{p} a_i a_j E(x_{n-i} x_{n-j})\end{aligned} \tag{3-5-7}$$

Assuming that the source output is (wide-sense) stationary, we may express (3-5-7) as

$$\mathscr{E}_p = \phi(0) - 2\sum_{i=1}^{p} a_i \phi(i) + \sum_{i=1}^{p}\sum_{j=1}^{p} a_i a_j \phi(i-j) \tag{3-5-8}$$

where $\phi(m)$ is the autocorrelation function of the sampled signal sequence x_n. Minimization of $\mathscr{E}_p$ with respect to the predictor coefficients $\{a_i\}$ results in the set of linear equations

$$\sum_{i=1}^{p} a_i \phi(i-j) = \phi(j), \qquad j = 1, 2, \ldots, p \tag{3-5-9}$$

Thus, the values of the predictor coefficients are established. When the autocorrelation function $\phi(n)$ is not known *a priori*, it may be estimated from the samples $\{x_n\}$ using the relation†

$$\hat{\phi}(n) = \frac{1}{N}\sum_{i=1}^{N-n} x_i x_{i+n}, \qquad n = 0, 1, 2, \ldots, p \tag{3-5-10}$$

and the estimate $\hat{\phi}(n)$ is used in (3-5-9) to solve for the coefficients $\{a_i\}$. Note that the normalization factor of $1/N$ in (3-5-10) drops out when $\hat{\phi}(n)$ is substituted in (3-5-9).

The linear equations in (3-5-9) for the predictor coefficients are called the *normal equations* or the *Yule–Walker equations*. There is an algorithm developed by Levinson (1947) and Durbin (1959) for solving these equations efficiently. It is described in Appendix A. We shall deal with the solution in greater detail in the subsequent discussion on linear predictive coding.

† The estimation of the autocorrelation function from a finite number of observations $\{x_i\}$ is a separate issue, which is beyond the scope of this discussion. The estimate in (3-5-10) is one that is frequently used in practice.

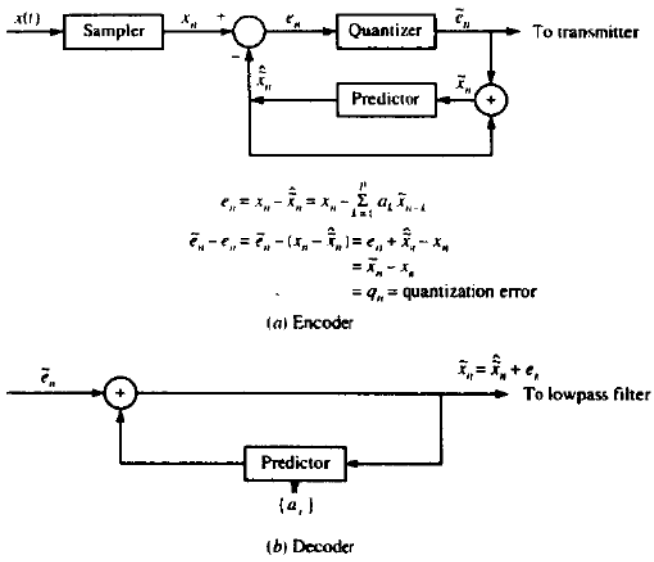


FIGURE 3-5-3 (*a*) Block diagram of a DPCM encoder. (*b*) DPCM decoder at the receiver.

Having described the method for determining the predictor coefficients, let us now consider the block diagram of a practical DPCM system, shown in Fig. 3-5-3(*a*). In this configuration, the predictor is implemented with the feedback loop around the quantizer. The input to the predictor is denoted by $\tilde{x}_n$, which represents the signal sample x_n modified by the quantization process, and the output of the predictor is

$$\hat{\tilde{x}}_n = \sum_{i=1}^{p} a_i \tilde{x}_{n-i} \tag{3-5-11}$$

The difference

$$e_n = x_n - \hat{\tilde{x}}_n \tag{3-5-12}$$

is the input to the quantizer and $\tilde{e}_n$ denotes the output. Each value of the quantized prediction error $\tilde{e}_n$ is encoded into a sequence of binary digits and transmitted over the channel to the destination. The quantized error $\tilde{e}_n$ is also added to the predicted value $\hat{\tilde{x}}_n$ to yield $\tilde{x}_n$.

At the destination, the same predictor that was used at the transmitting end is synthesized and its output $\hat{\tilde{x}}_n$ is added to $\tilde{e}_n$ to yield $\tilde{x}_n$. The signal $\tilde{x}_n$ is the desired excitation for the predictor and also the desired output sequence from which the reconstructed signal $\tilde{x}(t)$ is obtained by filtering, as shown in Fig. 3-5-3(*b*).

The use of feedback around the quantizer, as described above, ensures that the error in $\tilde{x}_n$ is simply the quantization error $q_n = \tilde{e}_n - e_n$ and that there is no

accumulation of previous quantization errors in the implementation of the decoder. That is,

$$\begin{aligned} q_n &= \tilde{e}_n - e_n \\ &= \tilde{e}_n - (x_n - \hat{x}_n) \\ &= \tilde{x}_n - x_n \end{aligned} \tag{3-5-13}$$

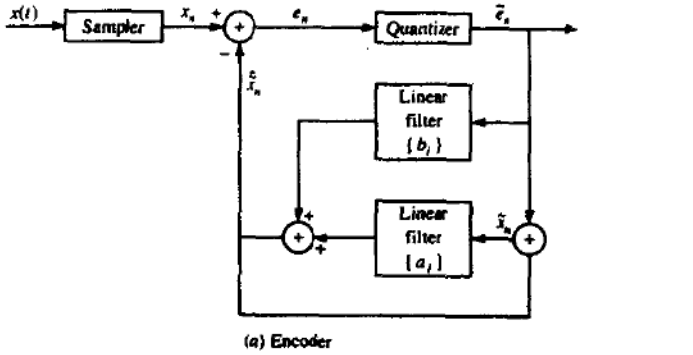
Hence $\tilde{x}_n = x_n + q_n$. This means that the quantized sample $\tilde{x}_n$ differs from the input x_n by the quantization error q_n independent of the predictor used. Therefore, the quantization errors do not accumulate.

In the DPCM system illustrated in Fig. 3-5-3, the estimate or predicted value $\hat{x}_n$ of the signal sample x_n is obtained by taking a linar combination of past values $\tilde{x}_{n-k}$, $k = 1, 2, \ldots, p$, as indicated by (3-5-11). An improvement in the quality of the estimate is obtained by including linearly filtered past values of the quantized error. Specifically, the $\hat{\hat{x}}_n$ estimate may be expressed as

$$\hat{\hat{x}}_n = \sum_{i=1}^{p} a_i \tilde{x}_{n-i} + \sum_{i=1}^{m} b_i \tilde{e}_{n-i} \tag{3-5-14}$$

where $\{b_i\}$ are the coefficients of the filter for the quantized error sequence $\tilde{e}_n$. The block diagrams of the encoder at the transmitter and the decoder at the receiver are shown in Fig. 3-5-4. The two sets of coefficients $\{a_i\}$ and $\{b_i\}$ are selected to minimize some function of the error $e_n = x_n - \hat{\hat{x}}_n$, such as the mean square error.

FIGURE 3-5-4 DPCM modified by the addition of linearly filtered error sequence.



Adaptive PCM and DPCM Many real sources are quasistationary in nature. One aspect of the quasistationary characteristic is that the variance and the autocorrelation function of the source output vary slowly with time. PCM and DPCM encoders, however, are designed on the basis that the source output is stationary. The efficiency and performance of these encoders can be improved by having them adapt to the slowly time-variant statistics of the source.

In both PCM and DPCM, the quantization error q_n resulting from a uniform quantizer operating on a quasistationary input signal will have a time-variant variance (quantization noise power). One improvement that reduces the dynamic range of the quantization noise is the use of an adaptive quantizer. Although the quantizer can be made adaptive in different ways, a relatively simple method is to use a uniform quantizer that varies its step size in accordance with the variance of the past signal samples. For example, a short-term running estimate of the variance of x_n can be computed from the input sequence $\{x_n\}$ and the step size can be adjusted on the basis of such an estimate. In its simplest form, the algorithm for the step-size adjustment employs only the previous signal sample. Such an algorithm has been successfully used by Jayant (1974) in the encoding of speech signals. Figure 3-5-5 illustrates such a (3 bit) quantizer in which the step size is adjusted recursively according to the relation

$$\Delta_{n+1} = \Delta_n M(n) \tag{3-5-15}$$

FIGURE 3-5-5 Example of a quantizer with an adaptive step size. (*Jayant, 1974.*)

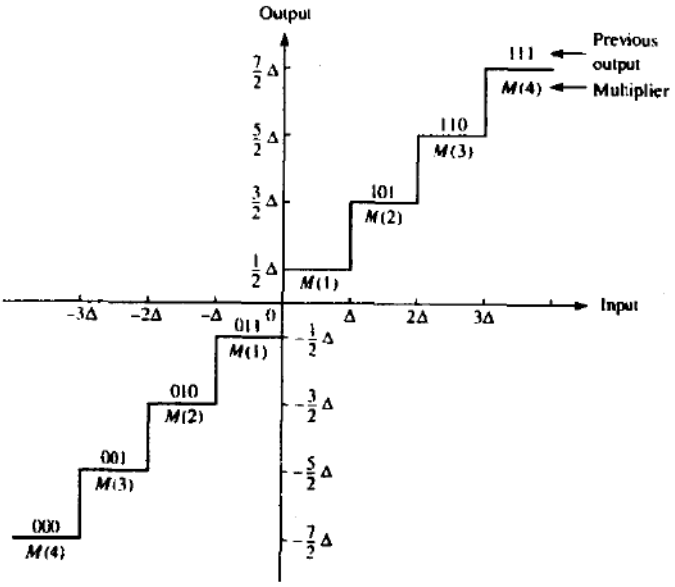


TABLE 3-5-1 MULTIPLICATION FACTORS FOR ADAPTIVE STEP SIZE ADJUSTMENT (JAYANT, 1974)

	PCM			DPCM		
	2	3	4	2	3	4
$M(1)$	0.60	0.85	0.80	0.80	0.90	0.90
$M(2)$	2.20	1.00	0.80	1.60	0.90	0.90
$M(3)$		1.00	0.80		1.25	0.90
$M(4)$		1.50	0.80		1.70	0.90
$M(5)$			1.20			1.20
$M(6)$			1.60			1.60
$M(7)$			2.00			2.00
$M(8)$			2.40			2.40

where $M(n)$ is a factor, whose value depends on the quantizer level for the sample x_n, and Δ_n is the step size of the quantizer for processing x_n. Values of the multiplication factors optimized for speech encoding have been given by Jayant (1974). These values are displayed in Table 3-5-1 for 2, 3, and 4 bit adaptive quantization.

In DPCM, the predictor can also be made adaptive when the source output in quasistationary. The coefficients of the predictor can be changed periodically to reflect the changing signal statistics of the source. The linear equations given by (3-5-9) still apply, with the short-term estimate of the autocorrelation function of x_n substituted in place of the ensemble correlation function. The predictor coefficients thus determined may be transmitted along with the quantized error $\tilde{e}(n)$ to the receiver, which implements the same predictor. Unfortunately, the transmission of the predictor coefficients results in a higher bit rate over the channel, offsetting, in part, the lower data rate achieved by having a quantizer with fewer bits (fewer levels) to handle the reduced dynamic range in the error e_n resulting from adaptive prediction.

As an alternative, the predictor at the receiver may compute its own prediction coefficients from $\tilde{e}_n$ and $\tilde{x}_n$, where

$$\tilde{x}_n = \tilde{e}_n + \sum_{i=1}^{p} a_i \tilde{x}_{n-i} \tag{3-5-16}$$

If we neglect the quantization noise, $\tilde{x}_n$ is equivalent to x_n. Hence, $\tilde{x}_n$ may be used to estimate the autocorrelation function $\phi(n)$ at the receiver, and the resulting estimates can be used in (3-5-9) in place of $\phi(n)$ to solve for the predictor coefficients. For sufficiently fine quantization, the difference between x_n and $\tilde{x}_n$ is very small. Hence, the estimate of $\phi(n)$ obtained from $\tilde{x}_n$ is usually adequate for determining the predictor coefficients. Implemented in this manner, the adaptive predictor results in a lower source data rate.

Instead of using the block processing approach for determining the

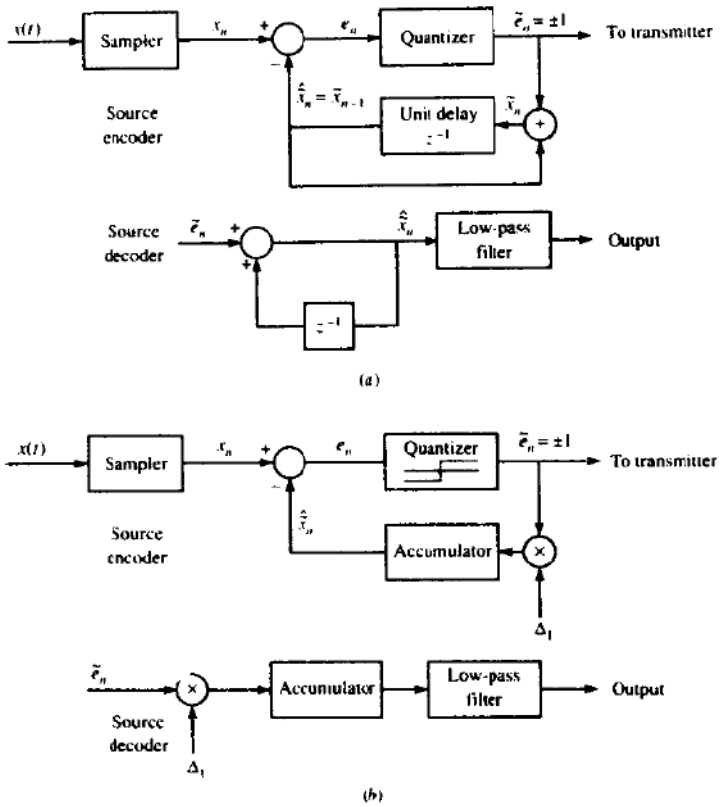


FIGURE 3-5-6 (*a*) Block diagram of a delta modulation system. (*b*) An equivalent realization of a delta modulation system.

predictor coefficients $\{a_i\}$ as described above, we may adapt the predictor coefficients on a sample-by-sample basis by using a gradient-type algorithm, similar in form to the adaptive gradient equalization algorithm that is described in Chapter 11. Similar gradient-type algorithms have also been devised for adapting the filter coefficients $\{a_i\}$ and $\{b_i\}$ of the DPCM system shown in Fig. 3-5-4. For details on such algorithms, the reader may refer to the book by Jayant and Noll (1984).

Delta Modulation (DM) Delta modulation may be viewed as a simplified form of DPCM in which a two-level (1 bit) quantizer is used in conjunction with a fixed first-order predictor. The block diagram of a DM encoder–decoder is shown in Fig. 3-5-6(*a*). We note that

$$\hat{\tilde{x}}_n = \tilde{x}_{n-1} = \hat{\tilde{x}}_{n-1} + \tilde{e}_{n-1} \tag{3-5-17}$$

Since

$$\begin{aligned} q_n &= \tilde{e}_n - e_n \\ &= \tilde{e}_n - (x_n - \hat{x}_n) \end{aligned}$$

It follows that

$$\hat{x}_n = x_{n-1} + q_{n-1}$$

Thus the estimated (predicted) value of x_n is really the previous sample x_{n-1} modified by the quantization noise q_{n-1}. We also note that the difference equation (3-5-17) represents an integrator with an input $\tilde{e}_n$. Hence, an equivalent realization of the one-step predictor is an accumulator with an input equal to the quantized error signal $\tilde{e}_n$. In general, the quantized error signal is scaled by some value, say Δ_1, which is called the *step size*. This equivalent realization is illustrated in Fig. 3-5-6(*b*). In effect, the encoder shown in Fig. 3-5-6 approximates a waveform $x(t)$ by a linear staircase function. In order for the approximation to be relatively good, the waveform $x(t)$ must change slowly relative to the sampling rate. This requirement implies that the sampling rate must be several (a factor of at least 5) times the Nyquist rate.

At any given sampling rate, the performance of the DM encoder is limited by two types of distortion, as illustrated in Fig. 3-5-7. One is called *slope-overload distortion*. It is due to the use of a step size Δ_1 that is too small to follow portions of the waveform that have a steep slope. The second type of distortion, called *granular noise*, results from using a step size that is too large in parts of the waveform having a small slope. The need to minimize both of these two types of distortion results in conflicting requirements in the selection of the step size Δ_1. One solution is to select Δ_1 to minimize the sum of the mean square values of these two distortions.

Even when Δ_1 is optimized to minimize the total mean square value of the slope-overload distortion and the granular noise, the performance of the DM encoder may still be less than satisfactory. An alternative solution is to employ a variable step size that adapts itself to the short-term characteristics of the source signal. That is, the step size is increased when the waveform has a steep

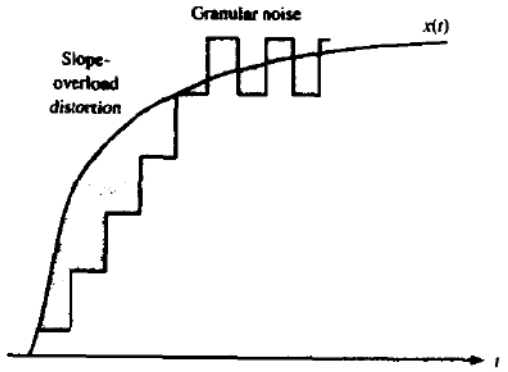


FIGURE 3-5-7 An example of slope overload distortion and granular noise in a delta modulation encoder.

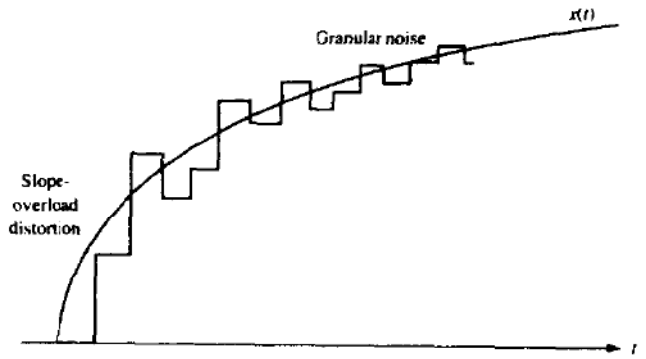


FIGURE 3-5-8 An example of variable-step-size delta modulation encoding.

slope and decreased when the waveform has a relatively small slope. This adaptive characteristic is illustrated in Fig. 3-5-8.

A variety of methods can be used to adaptively set the step size in every iteration. The quantized error sequence $\tilde{e}_n$ provides a good indication of the slope characteristics of the waveform being encoded. When the quantized error $\tilde{e}_n$ is changing signs between successive iterations, this is an indication that the slope of the waveform in that locality is relatively small. On the other hand, when the waveform has a steep slope, successive values of the error $\tilde{e}_n$ are expected to have identical signs. From these observations, it is possible to devise algorithms that decrease or increase the step size depending on successive values of $\tilde{e}_n$. A relatively simple rule devised by Jayant (1970) is to adaptively vary the step size according to the relation

$$\Delta_n = \Delta_{n-1} K^{\tilde{e}_n \tilde{e}_{n-1}}, \quad n = 1, 2, \ldots$$

where $K \geq 1$ is a constant that is selected to minimize the total distortion. A block diagram of a DM encoder–decoder that incorporates this adaptive algorithm is illustrated in Fig. 3-5-9.

Several other variations of adaptive DM encoding have been investigated and described in the technical literature. A particularly effective and popular technique first proposed by Greefkes (1970) is called *continuously variable slope delta modulation* (CVSD). In CVSD the adaptive step-size parameter may be expressed as

$$\Delta_n = \alpha \Delta_{n-1} + k_1$$

if $\tilde{e}_n$, $\tilde{e}_{n-1}$, and $\tilde{e}_{n-2}$ have the same sign; otherwise,

$$\Delta_n = \alpha \Delta_{n-1} + k_2$$

The parameters α, k_1, and k_2 are selected such that $0 < \alpha < 1$ and $k_1 \gg k_2 > 0$. For more discussion on this and other variations of adaptive DM, the interested reader is referred to the papers by Jayant (1974) and Flanagan *et al.* (1979), which contain extensive references.

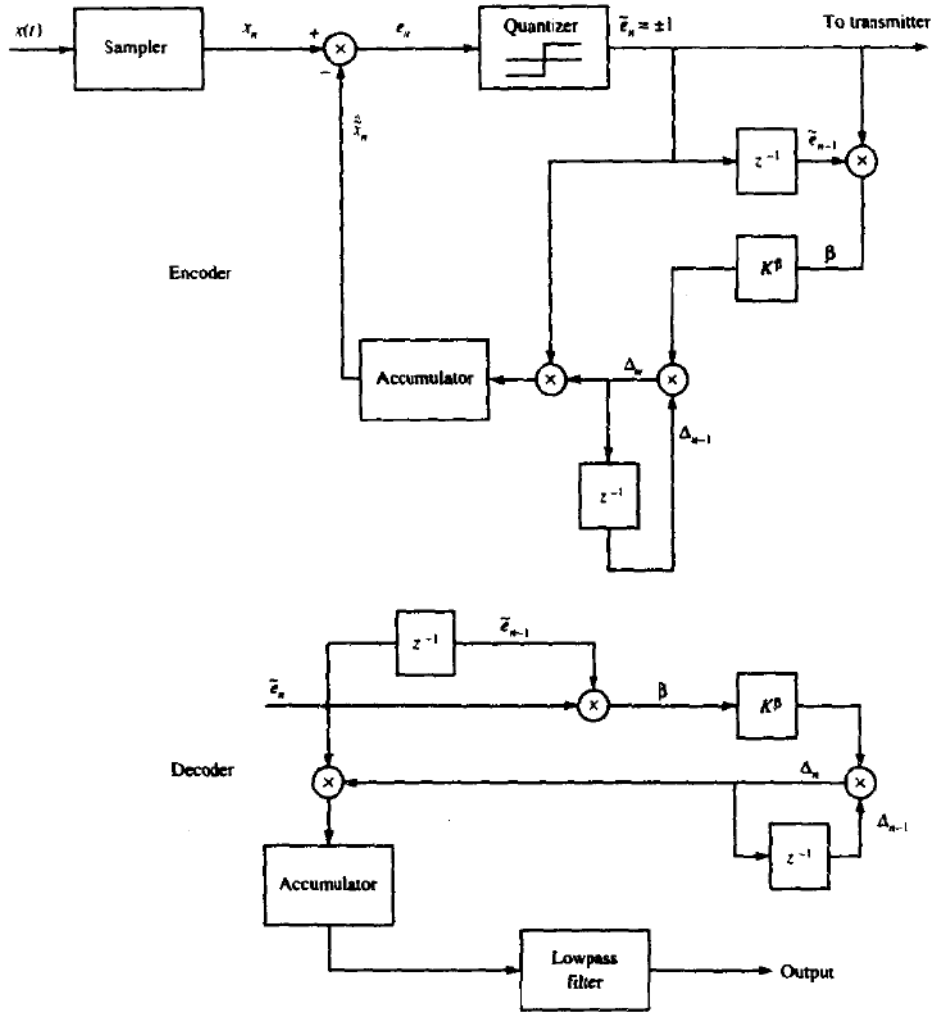


FIGURE 3-5-9 An example of a delta modulation system with adaptive step size.

PCM, DPCM, adaptive PCM, and adaptive DPCM and DM are all source encoding techniques that attempt to faithfully represent the output waveform from the source. The following class of waveform encoding methods is based on a spectral decomposition of the source signal.

3-5-2 Spectral Waveform Coding

In this section, we briefly describe waveform coding methods that filter the source output signal into a number of frequency bands or subbands and separately encode the signal in each subband. The waveform encoding may be

performed either on the time-domain waveforms in each subband or on the frequency-domain representation of the corresponding time-domain waveform in each subband.

Subband Coding In subband coding (SBC) of speech and image signals, the signal is divided into a small number of subbands and the time waveform in each subband is encoded separately. In speech coding, for example, the lower-frequency bands contain most of the spectral energy in voiced speech. In addition, quantization noise is more noticeable to the ear in the lower-frequency bands. Consequently, more bits are used for the lower-band signals and fewer are used for the higher-frequency bands.

Filter design is particularly important in achieving good performance in SBC. In practice, quadrature–mirror filters (QMFs) are generally used because they yield an alias-free response due to their perfect reconstruction property (see Vaidyanathan, 1993). By using QMFs in subband coding, the lower-frequency band is repeatedly subdivided by factors of two, thus creating octave-band filters. The output of each QMF filter is decimated by a factor of two, in order to reduce the sampling rate. For example, suppose that the bandwidth of a speech signal extends to 3200 Hz. The first pair of QMFs divides the spectrum into the low (0–1600 Hz) and high (1600–3200 Hz) bands. Then, the low band is split into low (0–800 Hz) and high (800–1600 Hz) bands by the use of another pair of QMFs. A third subdivision by another pair of QMFs can split the 0–800 Hz band into low (0–400 Hz) and high (400–800 Hz) bands. Thus, with three pairs of QMFs, we have obtained signals in the frequency bands 0–400, 400–800, 800–1600 and 1600–3200 Hz. The time-domain signal in each subband may now be encoded with different precision. In practice, adaptive PCM has been used for waveform encoding of the signal in each subband.

Adaptive Transform Coding In adaptive transform coding (ATC), the source signal is sampled and subdivided into frames of N_f samples, and the data in each frame is transformed into the spectral domain for coding and transmission. At the source decoder, each frame of spectral samples is transformed back into the time domain and the signal is synthesized from the time-domain samples and passed through a D/A converter. To achieve coding efficiency, we assign more bits to the more important spectral coefficients and fewer bits to the less important spectral coefficients. In addition, by designing an adaptive allocation in the assignment of the total number of bits to the spectral coefficients, we can adapt to possibly changing statistics of the source signal.

An objective in selecting the transformation from the time domain to the frequency domain is to achieve uncorrelated spectral samples. In this sense, the Karhunen–Loéve transform (KLT) is optimal in that it yields spectral values that are uncorrelated, but the KLT is generally difficult to compute (see

Wintz, 1972). The DFT and the *discrete cosine transform* (DCT) are viable alternatives, although they are suboptimum. Of these two, the DCT yields good performance compared with the KLT, and is generally used in practice (see Campanella and Robinson, 1971; Zelinsky and Noll, 1977).

In speech coding using ATC, it is possible to attain communication-quality speech at a rate of about 9600 bits/s.

3-5-3 Model-Based Source Coding

In contrast to the waveform encoding methods described above, model-based source coding represents a completely different approach. In this, the source is modeled as a linear system (filter) that, when excited by an appropriate input signal, results in the observed source output. Instead of transmitting the samples of the source waveform to the receiver, the parameters of the linear system are transmitted along with an appropriate excitation signal. If the number of parameters is sufficiently small, the model-based methods provide a large compression of the data.

The most widely used model-based coding method is called *linear predictive coding* (LPC). In this, the sampled sequence, denoted by x_n, $n = 0, 1, \ldots, N-1$, is assumed to have been generated by an all-pole (discrete-time) filter having the transfer function

$$H(z) = \frac{G}{1 - \sum_{k=1}^{p} a_k z^{-k}} \tag{3-5-18}$$

Appropriate excitation functions are an impulse, a sequence of impulses, or a sequence of white noise with unit variance. In any case, suppose that the input sequence is denoted by v_n, $n = 0, 1, 2, \ldots$. Then the output sequence of the all-pole model satisfies the difference equation

$$x_n = \sum_{k=1}^{p} a_k x_{n-k} + G v_n, \qquad n = 0, 1, 2, \ldots \tag{3-5-19}$$

In general, the observed source output x_n, $n = 0, 1, 2, \ldots, N-1$, does not satisfy the difference equation (3-5-19), but only its model does. If the input is a white-noise sequence or an impulse, we may form an estimate (or prediction) of x_n by the weighted linear combination

$$\hat{x}_n = \sum_{k=1}^{p} a_k x_{n-k}, \qquad n > 0 \tag{3-5-20}$$

The difference between x_n and $\hat{x}_n$, namely,

$$\begin{aligned} e_n &= x_n - \hat{x}_n \\ &= x_n - \sum_{k=1}^{p} a_k x_{n-k} \end{aligned} \tag{3-5-21}$$

represents the error between the observed value x_n and the estimated (predicted) value $\hat{x}_n$. The filter coefficients $\{a_k\}$ can be selected to minimize the mean square value of this error.

Suppose for the moment that the input $\{v_n\}$ is a white-noise sequence. Then, the filter output x_n is a random sequence and so is the difference $e_n = x_n - \hat{x}_n$. The ensemble average of the squared error is

$$\begin{aligned}\mathscr{E}_p &= E(e_n^2) \\ &= E\left[\left(x_n - \sum_{k=1}^{p} a_k x_{n-k}\right)^2\right] \\ &= \phi(0) - 2\sum_{k=1}^{p} a_k \phi(k) + \sum_{k=1}^{p}\sum_{m=1}^{p} a_k a_m \phi(k-m)\end{aligned} \tag{3-5-22}$$

where $\phi(m)$ is the autocorrelation function of the sequence x_n, $n = 0, 1, \ldots, N-1$. But $\mathscr{E}_p$ is identical to the MSE given by (3-5-8) for a predictor used in DPCM. Consequently, minimization of $\mathscr{E}_p$ in (3-5-22) yields the set of normal equations given previously by (3-5-9). To completely specify the filter $H(z)$, we must also determine the filter gain G. From (3-5-19), we have

$$E[(Gv_n)^2] = G^2 E(v_n^2) = G^2 = E\left[\left(x_n - \sum_{k=1}^{p} a_k x_{n-k}\right)^2\right] = \mathscr{E}_p \tag{3-5-23}$$

where $\mathscr{E}_p$ is the residual MSE obtained from (3-5-22) by substituting the optimum prediction coefficients, which result from the solution of (3-5-9). With this substitution, the expression for $\mathscr{E}_p$ and, hence, G^2 simplifies to

$$\mathscr{E}_p = G^2 = \phi(0) - \sum_{k=1}^{p} a_k \phi(k) \tag{3-5-24}$$

In practice, we do not usually know *a priori* the true autocorrelation function of the source output. Hence, in place of $\phi(n)$, we substitute an estimate $\hat{\phi}(n)$ as given by (3-5-10), which is obtained from the set of samples x_n, $n = 0, 1, \ldots, N-1$, emitted by the source.

As indicated previously, the Levinson–Durbin algorithm derived in Appendix A may be used to solve for the predictor coefficients $\{a_k\}$ recursively, beginning with a first-order predictor and iterating the order of the predictor up to order p. The recursive equations for the $\{a_k\}$ may be expressed as

$$\begin{aligned}a_{ii} &= \frac{\hat{\phi}(i) - \sum_{k=1}^{i-1} a_{i-1,k}\hat{\phi}(i-k)}{\hat{\mathscr{E}}_{i-1}} \qquad i = 2, 3, \ldots, p \\ a_{ik} &= a_{i-1,k} - a_{ii}a_{i-1,i-k}, \qquad 1 \le k \le i-1 \\ \hat{\mathscr{E}}_i &= (1 - a_{ii})\hat{\mathscr{E}}_{i-1} \\ a_{11} &= \frac{\hat{\phi}(1)}{\hat{\phi}(0)}, \qquad \hat{\mathscr{E}}_0 = \hat{\phi}(0)\end{aligned} \tag{3-5-25}$$

where a_{ik}, $k = 1, 2, \ldots, i$, are the coefficients of the ith-order predictor. The desired coefficients for the predictor of order p are

$$a_k \equiv a_{pk}, \quad k = 1, 2, \ldots, p \tag{3-5-26}$$

and the residual MSE is

$$\begin{aligned} \hat{\mathscr{E}} = G^2 &= \hat{\phi}(0) - \sum_{k=1}^{p} a_k \hat{\phi}(k) \\ &= \hat{\phi}(0) \prod_{i=1}^{p} (1 - a_{ii}^2) \end{aligned} \tag{3-5-27}$$

We observe that the recursive relations in (3-5-25) give us not only the coefficients of the predictor for order p, but also the predictor coefficients of all orders less than p.

The residual MSE $\hat{\mathscr{E}}_i$, $i = 1, 2, \ldots, p$, forms a monotone decreasing sequence, i.e. $\hat{\mathscr{E}}_p \leq \hat{\mathscr{E}}_{p-1} \leq \ldots \leq \hat{\mathscr{E}}_1 \leq \hat{\mathscr{E}}_0$, and the prediction coefficients a_{ii} satisfy the condition

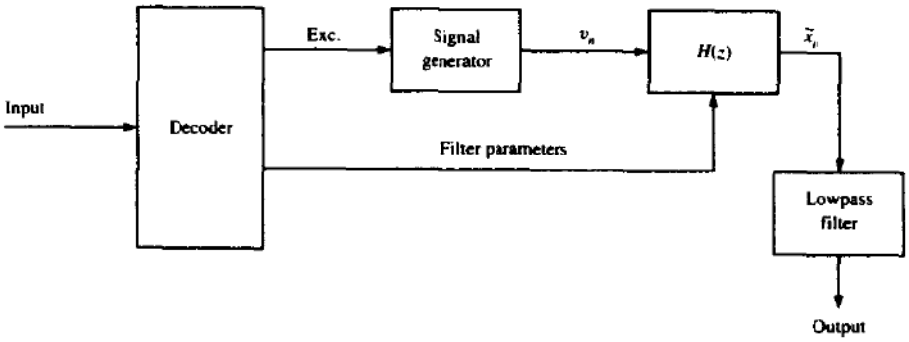
$$|a_{ii}| < 1, \quad i = 1, 2, \ldots, p \tag{3-5-28}$$

This condition is necessary and sufficient for all the poles of $H(z)$ to be inside the unit circle. Thus (3-5-28) ensures that the model is stable.

LPC has been successfully used in the modeling of a speech source. In this case, the coefficients a_{ii}, $i = 1, 2, \ldots, p$, are called *reflection coefficients* as a consequence of their correspondence to the reflection coefficients in the acoustic tube model of the vocal tract (see Rabiner and Schafer, 1978; Deller *et al.*, 1993).

Once the predictor coefficients and the gain G have been estimated from the source output $\{x_n\}$, each parameter is coded into a sequence of binary digits and transmitted to the receiver. Source decoding or waveform synthesis may be accomplished at the receiver as illustrated in Fig. 3-5-10. The signal generator is used to produce the excitation function $\{v_n\}$, which is scaled by G

FIGURE 3-5-10 Block diagram of a waveform synthesizer (source decoder) for an LPC system.



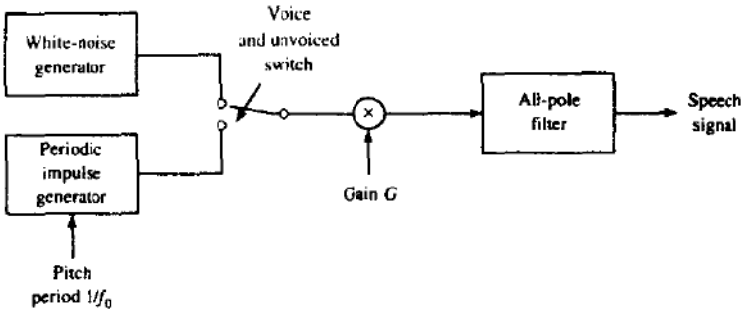


FIGURE 3-5-11 Block diagram model of the generation of a speech signal.

to produce the desired input to the all-pole filter model $H(z)$ synthesized from the received prediction coefficients. The analog signal may be reconstructed by passing the output sequence from $H(z)$ through an analog filter that basically performs the function of interpolating the signal between sample points. In this realization of the waveform synthesizer, the excitation function and the gain parameter must be transmitted along with the prediction coefficients to the receiver.

When the source output is stationary, the filter parameters need to be determined only once. However, the statistics of most sources encountered in practice are at best quasistationary. Under these circumstances, it is necessary to periodically obtain new estimates of the filter coefficients, the gain G, and the type of excitation function, and to transmit these estimates to the receiver.

Example 3-5-1

The block diagram shown in Fig. 3-5-11 illustrates a model for a speech source. There are two mutually exclusive excitation functions to model voiced and unvoiced speech sounds. On a short-time basis, voiced speech is periodic with a fundamental frequency f_0 or a pitch period $1/f_0$ that depends on the speaker. Thus voiced speech is generated by exciting an all-pole filter model of the vocal tract by a periodic impulse train with a period equal to the desired pitch period. Unvoiced speech sounds are generated by exciting the all-pole filter model by the output of a random-noise generator. The speech encoder at the transmitter must determine the proper excitation function, the pitch period for voiced speech, the gain parameter G, and the prediction coefficients. These parameters are encoded into binary digits and transmitted to the receiver. Typically, the voiced and unvoiced information requires 1 bit, the pitch period is adequately represented by 6 bits, and the gain parameter may be represented by 5 bits after its dynamic range is compressed logarithmically. The prediction coefficients require 8–10 bits/coefficient for adequate representation (see Rabiner and Schafer, 1978). The reason for such high accuracy is that relatively small changes in

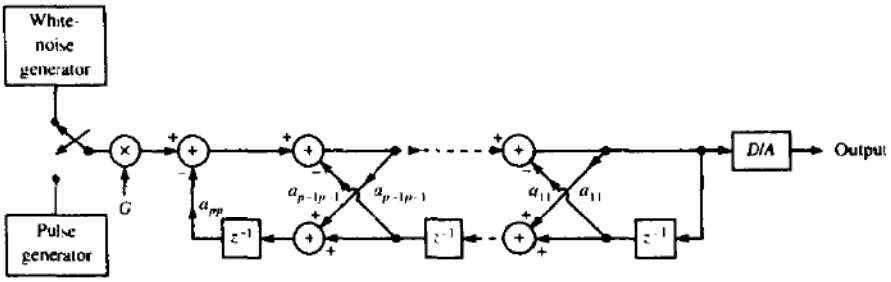


FIGURE 3-5-12 All-pole lattice filter for synthesizing the speech signal.

the prediction coefficients result in a large change in the pole positions of the filter model $H(z)$. The accuracy requirements may be lessened by transmitting the reflection coefficients a_{ii}, which have a smaller dynamic range. These are adequately represented by 6 bits. Thus, for a predictor of order $p = 10$ [five poles in $H(z)$], the total number of bits is 72. Due to the quasistationary nature of the speech signal, the linear system model must be changed periodically, typically once every 15–30 ms. Consequently, the bit rate from the source encoder is in the range 4800–2400 bit/s.

When the reflection coefficients are transmitted to the decoder, it is not necessary to recompute the prediction coefficients in order to realize the speech synthesizer. Instead, the synthesis is performed by realizing a lattice filter, shown in Fig. 3-5-12, which utilizes the reflection coefficients directly and which is equivalent to the linear prediction filter.

The linear all-pole filter model, for which the filter coefficients are estimated via linear prediction, is by far the simplest linear model for a source. A more general source model is a linear filter that contains both poles and zeros. In a pole–zero model, the source output x_n satisfies the difference equation

$$x_n = \sum_{k=1}^{p} a_k x_{n-k} + \sum_{k=0}^{q} b_k v_{n-k}$$

where v_n is the input excitation sequence. The problem now is to estimate the filter parameters $\{a_k\}$ and $\{b_k\}$ from the data x_i, $i = 0, 1, \ldots, N-1$, emitted by the source. However, the MSE criterion applied to the minimization of the error $e_n = x_n - \hat{x}_n$, where $\hat{x}_n$ is an estimate of x_n, results in a set of nonlinear equations for the parameters $\{a_k\}$ and $\{b_k\}$. Consequently, the evaluation of the $\{a_k\}$ and $\{b_k\}$ becomes tedious and difficult mathematically. To avoid having to solve the nonlinear equations, a number of suboptimum methods have been devised for pole–zero modeling. A discussion of these techniques would lead us too far afield, however.

LPC as described above forms the basis for more complex model-based source encoding methods. When applied to speech coding, the model-based

methods are generally called *vocoders* (for voice coders). In addition to the conventional LPC vocoder described above, other types of vocoders that have been implemented include the residual excited LPC (RELP) vocoder, the multipulse LPC vocoder, the code-excited LPC (CELP) vocoder, and the vector-sum-excited LPC (VSELP) vocoder. The CELP and VSELP vocoders employ vector-quantized excitation codebooks to achieve communication quality speech at low bit rates.

Before concluding this section, we consider the application of waveform encoding and LPC to the encoding of speech signals and compare the bit rates of these coding techniques.

Encoding Methods Applied to Speech Signals The transmission of speech signals over telephone lines, radio channels, and satellite channels constitutes by far the largest part of our daily communications. It is understandable, therefore, that over the past three decades more research has been performed on speech encoding than on any other type of information-bearing signal. In fact, all the encoding techniques described in this section have been applied to the encoding of speech signals. It is appropriate, therefore, to compare the efficiency of these methods in terms of the bit rate required to transmit the speech signal.

The speech signal is assumed to be band-limited to the frequency range 200–3200 Hz and sampled at a nominal rate of 8000 samples/s for all encoders except DM, where the sampling rate is f_s identical to the bit rate. For an LPC encoder, the parameters given in Example 3-5-1 are assumed.

Table 3-5-2 summarizes the main characteristics of the encoding methods described in this section and the required bit rate. In terms of the quality of the speech signal synthesized at the receiver from the (error-free) binary sequence, all the waveform encoding methods (PCM, DPCM, ADPCM, DM, ADM) provide telephone (toll) quality speech. In other words, a listener would have difficulty discerning the difference between the digitized speech and the analog speech waveform. ADPCM and ADM are particularly efficient waveform encoding techniques. With CVSD, it is possible to operate down to 9600 bits/s

TABLE 3-5-2 ENCODING TECHNIQUES APPLIED TO SPEECH SIGNALS

Encoding method	Quantizer	Coder	Transmission rate (bits/s)
PCM	Linear	12 bits	96 000
Log PCM	Logarithamic	7–8 bits	56 000–64 000
DPCM	Logarithmic	4–6 bits	32 000–48 000
ADPCM	Adaptive	3–4 bits	24 000–32 000
DM	Binary	1 bit	32 000–64 000
ADM	Adaptive binary	1 bit	16 000–32 000
LPC			2400–4800

with some noticeable waveform distortion. In fact, at rates below 16 000 bits/s, the distortion produced by waveform encoders increases significantly. Consequently, these techniques are not used below 9600 bits/s.

For rates below 9600 bits/s, encoding techniques, such as LPC, that are based on linear models of the source are usually employed. The synthesized speech obtained from this class of encoding techniques is intelligible. However, the speech signal has a synthetic quality and there is noticeable distortion.

3-6 BIBLIOGRAPHICAL NOTES AND REFERENCES

Source coding has been an area of intense research activity since the publication of Shannon's classic papers in 1948 and the paper by Huffman (1952). Over the years, major advances have been made in the development of highly efficient source data compression algorithms. Of particular significance is the research on universal source coding and universal quantization published by Ziv (1985), Ziv and Lempel (1977, 1978), Davisson (1973), Gray (1975), and Davisson *et al.* (1981).

Treatments of rate distortion theory are found in the books by Gallager (1968), Berger (1971), Viterbi and Omura (1979), Blahut (1987) and Gray (1990).

Much work has been done over the past several decades on speech encoding methods. Our treatment provides an overview of this important topic. A more comprehensive treatment is given in the books by Rabiner and Schafer (1978), Jayant and Noll (1984), and Deller *et al.* (1993). In addition to these texts, there have been special issues of the *IEEE Transactions on Communications* (April 1979 and April 1982) and, more recently, the *IEEE Journal on Selected Areas in Communications* (February 1988) devoted to speech encoding. We should also mention the publication by IEEE Press of a book containing reprints of published papers on waveform quantization and coding, edited by Jayant (1976).

Over the past decade, we have also seen a number of important developments in vector quantization. Our treatment of this topic was based on the tutorial paper by Makhoul *et al.* (1985). A comprehensive treatment of vector quantization and signal compression is provided in the book by Gersho and Gray (1992).

PROBLEMS

3-1 Consider the joint experiment described in Problem 2-1 with the given joint probabilities $P(A_i, B_j)$. Suppose we observe the outcomes A_i, $i = 1, 2, 3, 4$ of experiment A.

a Determine the mutual information $I(B_j; A_i)$ for $j = 1, 2, 3$ and $i = 1, 2, 3, 4$, in bits.

b Determine the average mutual information $I(B; A)$.

3-2 Suppose the outcomes B_j, $j = 1, 2, 3$, in Problem 3-1 represent the three possible output letters from the DMS. Determine the entropy of the source.

3-3 Prove that $\ln u \leq u - 1$ and also demonstrate the validity of this inequality by plotting $\ln u$ and $u - 1$ on the same graph.

3-4 X and Y are two discrete random variables with probabilities

$$P(X = x, Y = y) \equiv P(x, y)$$

Show that $I(X; Y) \geq 0$, with equality if and only if X and Y are statistically independent.

[*Hint:* Use the inequality $\ln u < u - 1$, for $0 < u < 1$, to show that $-I(X; Y) \leq 0$.]

3-5 The output of a DMS consists of the possible letters $x_1, x_2, \ldots, x_n$, which occur with probabilities $p_1, p_2, \ldots, p_n$, respectively. Prove that the entropy $H(X)$ of the source is at most $\log n$.

3-6 Determine the differential entropy $H(X)$ of the uniformly distributed random variable X with pdf

$$p(x) = \begin{cases} a^{-1} & (0 \leq x \leq a) \\ 0 & (\text{otherwise}) \end{cases}$$

for the following three cases:

a $a = 1$;

b $a = 4$;

c $a = \frac{1}{4}$.

Observe from these results that $H(X)$ is not an absolute measure, but only a relative measure of randomness.

3-7 A DMS has an alphabet of eight letters, x_i, $i = 1, 2, \ldots, 8$, with probabilities 0.25, 0.20, 0.15, 0.12, 0.10, 0.08, 0.05, and 0.05.

a Use the Huffman encoding procedure to determine a binary code for the source output.

b Determine the average number $\bar{R}$ of binary digits per source letter.

c Determine the entropy of the source and compare it with $\bar{R}$.

3-8 A DMS has an alphabet of five letters, x_i, $i = 1, 2, \ldots, 5$, each occurring with probability $\frac{1}{5}$. Evaluate the efficiency of a fixed-length binary code in which

a each letter is encoded separately into a binary sequence;

b two letters at a time are encoded into a binary sequence;

c three letters at a time are encoded into a binary sequence.

3-9 Recall (3-2-6):

$$I(x_i; y_j) = I(x_i) - I(x_i \mid y_j)$$

Prove that

a $I(x_i; y_j) = I(y_j) - I(y_j \mid x_i)$;

b $I(x_i; y_j) = I(x_i) + I(y_j) - I(x_i y_j)$, where $I(x_i y_j) = -\log P(x_i, y_j)$.

3-10 Let X be a geometrically distributed random variable; that is,

$$p(X = k) = p(1 - p)^{k-1}, \quad k = 1, 2, 3, \ldots$$

a Find the entropy of X.

b Knowing that $X > K$, where K is a positive integer, what is the entropy of X?

3-11 Let X and Y denote two jointly distributed discrete valued random variables.

a Show that

$$H(X) = -\sum_{x,y} P(x, y) \log P(x)$$

$$H(Y) = -\sum_{x,y} P(x, y) \log P(y)$$

b Use the above result to show that

$$H(X, Y) \leq H(X) + H(Y)$$

When does equality hold?

c Show that

$$H(X \mid Y) \leq H(X)$$

with equality if and only if X and Y are independent.

3-12 Two binary random variables X and Y are distributed according to the joint distributions $p(X = Y = 0) = p(X = 0, Y = 1) = p(X = Y = 1) = \frac{1}{3}$. Compute $H(X)$, $H(Y)$, $H(X \mid Y)$, $H(Y \mid X)$, and $H(X, Y)$.

3-13 A Markov process is a process with one-step memory, i.e., a process such that

$$p(x_n \mid x_{n-1}, x_{n-2}, x_{n-3}, \ldots) = p(x_n \mid x_{n-1})$$

for all n. Show that, for a stationary Markov process, the entropy rate is given by $H(X_n \mid X_{n-1})$.

3-14 Let $Y = g(X)$, where g denotes a deterministic function. Show that, in general, $H(Y) \leq H(X)$. When does equality hold?

3-15 Show that $I(X; Y) = H(X) + H(Y) - H(XY)$.

3-16 Show that, for statistically independent events,

$$H(X_1 X_2 \cdots X_n) = \sum_{i=1}^{n} H(X_i)$$

3-17 For a noiseless channel, show that $H(X \mid Y) = 0$.

3-18 Show that

$$I(X_3; X_2 \mid X_1) = H(X_3 \mid X_1) - H(X_3 \mid X_1 X_2)$$

and that

$$H(X_3 \mid X_1) \geq H(X_3 \mid X_1 X_2)$$

3-19 Let X be a random variable with pdf $p_X(x)$ and let $Y = aX + b$ be a linear transformation of X, where a and b are two constants. Determine the differential entropy $H(Y)$ in terms of $H(X)$.

3-20 The outputs x_1, x_2, and x_3 of a DMS with corresponding probabilities $p_1 = 0.45$, $p_2 = 0.35$, and $p_3 = 0.20$ are transformed by the linear transformation $Y = aX + b$, where a and b are constants. Determine the entropy $H(Y)$ and comment on what effect the transformation has had on the entropy of X.

3-21 The optimum four-level nonuniform quantizer for a gaussian-distributed signal amplitude results in the four levels a_1, a_2, a_3, and a_4, with corresponding probabilities of occurrence $p_1 = p_2 = 0.3365$ and $p_3 = p_4 = 0.1635$.

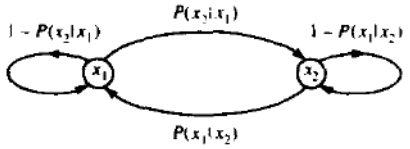


FIGURE P3-22

a Design a Huffman code that encodes a single level at a time and determine the average bit rate.

b Design a Huffman code that encodes two output levels at a time and determine the average bit rate.

c What is the minimum rate obtained by encoding J output levels at a time as $J \to \infty$?

3-22 A first-order Markov source is characterized by the state probabilities $P(x_i)$, $i = 1, 2, \ldots, L$, and the transition probabilities $P(x_k \mid x_i)$, $k = 1, 2, \ldots, L$, and $k \neq i$. The entropy of the Markov source is

$$H(X) = \sum_{k=1}^{L} P(x_k) H(X \mid x_k)$$

where $H(X \mid x_k)$ is the entropy conditioned on the source being in state x_k.

Determine the entropy of the binary, first-order Markov source shown in Fig. P3-22, which has the transition probabilities $P(x_2 \mid x_1) = 0.2$ and $P(x_1 \mid x_2) = 0.3$. [Note that the conditional entropies $H(X \mid x_1)$ and $H(X \mid x_2)$ are given by the binary entropy functions $H[P(x_2 \mid x_1)]$ and $H[P(x_1 \mid x_2)]$, respectively.] How does the entropy of the Markov source compare with the entropy of a binary DMS with the same output letter probabilities $P(x_1)$ and $P(x_2)$?

3-23 A memoryless source has the alphabet $\mathcal{A} = \{-5, -3, -1, 0, 1, 3, 5\}$, with corresponding probabilities $\{0.05, 0.1, 0.1, 0.15, 0.05, 0.25, 0.3\}$.

a Find the entropy of the source.

b Assuming that the source is quantized according to the quantization rule

$$q(-5) = q(-3) = 4$$
$$q(-1) = q(0) = q(1) = 0$$
$$q(3) = q(5) = 4$$

find the entropy of the quantized source.

3-24 Design a *ternary* Huffman code, using 0, 1, and 2 as letters, for a source with output alphabet probabilities given by $\{0.05, 0.1, 0.15, 0.17, 0.18, 0.22, 0.13\}$. What is the resulting average codeword length? Compare the average codeword length with the entropy of the source. (In what base would you compute the logarithms in the expression for the entropy for a meaningful comparison?)

3-25 Find the Lempel–Ziv source code for the binary source sequence

0001001000000110000100000001000000101000010000001101000000001100

Recover the original sequence back from the Lempel–Ziv source code.
[*Hint:* You require two passes of the binary sequence to decide on the size of the dictionary.]

3-26 Find the differential entropy of the continuous random variable X in the following cases:

a X is an exponential random variable with parameter $\lambda > 0$, i.e.,

$$f_X(x) = \begin{cases} \lambda^{-1} e^{-x/\lambda} & (x > 0) \\ 0 & (\text{otherwise}) \end{cases}$$

b X is a Laplacian random variable with parameter $\lambda > 0$, i.e.,

$$f_X(x) = \frac{1}{2\lambda} e^{-|x|/\lambda}$$

c X is a triangular random variable with parameter $\lambda > 0$, i.e.,

$$f_X(x) = \begin{cases} (x + \lambda)/\lambda^2 & (-\lambda \leq x \leq 0) \\ (-x + \lambda)/\lambda^2 & (0 < x \leq \lambda) \\ 0 & (\text{otherwise}) \end{cases}$$

3-27 It can be shown that the rate-distortion function for a Laplacian source, $f_X(x) = (2\lambda)^{-1} e^{-|x|/\lambda}$ with an absolute value of error-distortion measure $d(x, \hat{x}) = |x - \hat{x}|$ is given by

$$R(D) = \begin{cases} \log(\lambda/D) & (0 \leq D \leq \lambda) \\ 0 & (D > \lambda) \end{cases}$$

(see Berger, 1971).

a How many bits per sample are required to represent the outputs of this source with an average distortion not exceeding $\frac{1}{2}\lambda$?

b Plot $R(D)$ for three different values of λ and discuss the effect of changes in λ on these plots.

3-28 It can be shown that if X is a zero-mean continuous random variable with variance σ^2, its rate distortion function, subject to squared error distortion measure, satisfies the lower and upper bounds given by the inequalities

$$h(X) - \tfrac{1}{2} \log 2\pi e D \leq R(D) \leq \tfrac{1}{2} \log \tfrac{1}{2}\sigma^2$$

where $h(X)$ denotes the differential entropy of the random variable X (see Cover and Thomas, 1991).

a Show that, for a Gaussian random variable, the lower and upper bounds coincide.

b Plot the lower and upper bounds for a Laplacian source with $\sigma = 1$.

c Plot the lower and upper bounds for a triangular source with $\sigma = 1$.

3-29 A stationary random process has an autocorrelation function given by $R_X = \frac{1}{2}A^2 e^{-|\tau|} \cos 2\pi f_0 \tau$ and it is known that the random process never exceeds 6 in magnitude. Assuming $A = 6$, how many quantization levels are required to guarantee a signal-to-quantization noise ratio of at least 60 dB?

3-30 An additive white gaussian noise channel has the output $Y = X + G$, where X is the channel input and G is the noise with probability density function

$$p(n) = \frac{1}{\sqrt{2\pi}\,\sigma_n} e^{-n^2/2\sigma_n^2}$$

If X is a white gaussian input with $E(X) = 0$ and $E(X^2) = \sigma_x^2$, determine

a the conditional differential entropy $H(X \mid G)$;

b the average mutual information $I(X; Y)$.

3-31 A DMS has an alphabet of eight letters, x_i, $i = 1, 2, \ldots, 8$, with probabilities

given in Problem 3-7. Use the Huffman encoding procedure to determine a ternary code (using symbols 0, 1, and 2) for encoding the source output.

[*Hint:* Add a symbol x_9 with probability $p_9 = 0$, and group three symbols at a time.]

3-32 Determine whether there exists a binary code with code word lengths $(n_1, n_2, n_3, n_4) = (1, 2, 2, 3)$ that satisfy the prefix condition.

3-33 Consider a binary block code with 2^n code words of the same length n. Show that the Kraft inequality is satisfied for such a code.

3-34 Show that the entropy of an n-dimensional gaussian vector $\mathbf{X} = [x_1\ x_2\ \ldots\ x_n]$ with zero mean and covariance matrix $\mathbf{M}$ is

$$H(\mathbf{X}) = \tfrac{1}{2}\log_2 (2\pi e)^n\, |\boldsymbol{M}|$$

3-35 Consider a DMS with output bits (0, 1) that are equiprobable. Define the distortion measure as $D = P_e$, where P_e is the probability of error in transmitting the binary symbols to the user over a BSC. Then the rate distortion function is (Berger, 1971)

$$R(D) = 1 + D\log_2 D + (1 - D)\log_2 (1 - D), \qquad 0 \leq D = P_e \leq \tfrac{1}{2}$$

Plot $R(D)$ for $0 \leq D \leq \frac{1}{2}$.

3-36 Evaluate the rate distortion function for an M-ary symmetric channel where $D = P_M$ and

$$R(D) = \log_2 M + D\log_2 D + (1 - D)\log_2 \frac{1 - D}{M - 1}$$

for $M = 2, 4, 8$, and 16. P_M is the probability of error.

3-37 Consider the use of the weighted mean-square-error (MSE) distortion measure defined as

$$d_w(\mathbf{X}, \tilde{\mathbf{X}}) = (\mathbf{X} - \tilde{\mathbf{X}})'\mathbf{W}(\mathbf{X} - \tilde{\mathbf{X}})$$

where $\mathbf{W}$ is a symmetric, positive-definite wieghting matrix. By factorizing $\mathbf{W}$ as $\mathbf{W} = \mathbf{P}'\mathbf{P}$, show that $d_w(\mathbf{X}, \tilde{\mathbf{X}})$ is equivalent to an unweighted MSE distortion measure $d_2(\mathbf{X}', \tilde{\mathbf{X}}')$ involving transformed vectors $\mathbf{X}'$ and $\tilde{\mathbf{X}}'$.

3-38 Consider a stationary stochastic signal sequence $\{X(n)\}$ with zero mean and autocorrelation sequence

$$\phi(n) = \begin{cases} 1 & (n = 0) \\ \frac{1}{2} & (n = \pm 1) \\ 0 & (\text{otherwise}) \end{cases}$$

a Determine the prediction coefficient of the first-order minimum MSE predictor for $\{X(n)\}$ given by

$$\hat{x}(n) = a_1 x(n - 1)$$

and the corresponding minimum mean square error $\mathscr{E}_1$.

b Repeat (a) for the second-order predictor

$$\hat{x}(n) = a_1 x(n - 1) + a_2 x(n - 2)$$

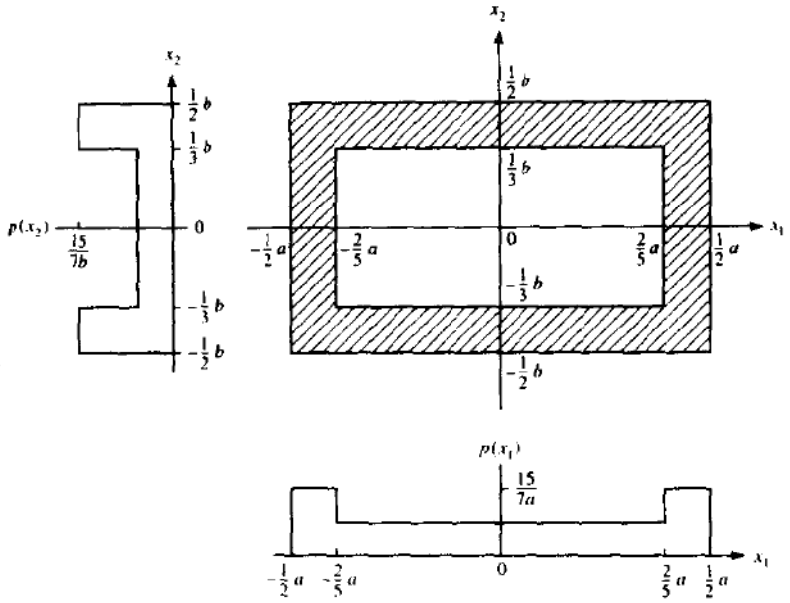


FIGURE P3-39

3-39 Consider the encoding of the random variables x_1 and x_2 that are characterized by the joint pdf $p(x_1, x_2)$ given by

$$p(x_1, x_2) = \begin{cases} 15/7ab & (x_1, x_2 \in C) \\ 0 & (\text{otherwise}) \end{cases}$$

as shown in Fig. P3-39. Evaluate the bit rates required for uniform quantization of x_1 and x_2 separately (scalar quantization) and combined (vector) quantization of (x_1, x_2). Determine the difference in bit rate when $a = 4b$.

3-40 Consider the encoding of two random variables X and Y that are uniformly distributed on the region between two squares as shown in Fig. P3-40.

a Find $f_X(x)$ and $f_Y(y)$.

b Assume that each of the random variables X and Y are quantized using four level uniform quantizers. What is the resulting distortion? What is the resulting number of bits per (X, Y) pair?

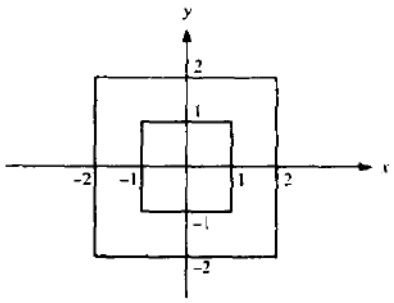


FIGURE P3-40

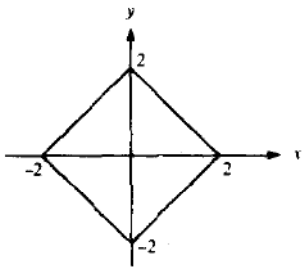


FIGURE P3-41

c Now assume that instead of scalar quantizers for X and Y, we employ a vector quantizer to achieve the same level of distortion as in (b). What is the resulting number of bits per source output pair (X, Y)?

3-41 Two random variables X and Y are uniformly distributed on the square shown in Fig. P3-41.

a Find $f_X(x)$ and $f_Y(y)$.

b Assume that each of the random variables X and Y are quantized using four level uniform quantizers. What is the resulting distortion? What is the resulting number of bits per (X, Y) pair?

c Now assume that, instead of scalar quantizers for X and Y, we employ a vector quantizer with the same number of bits per source output pair (X, Y) as in (b). What is the resulting distortion for this vector quantizer?

4

CHARACTERIZATION OF COMMUNICATION SIGNALS AND SYSTEMS

Signals can be categorized in a number of different ways, such as random versus deterministic, discrete time versus continuous time, discrete amplitude versus continuous amplitude, lowpass versus bandpass, finite energy versus infinite energy, finite average power versus infinite average power, etc. In this chapter, we treat the characterization of signals and systems that are usually encountered in the transmission of digital information over a communication channel. In particular, we introduce the representation of various forms of digitally modulated signals and describe their spectral characteristics.

We begin with the characterization of bandpass signals and systems, including the mathematical representation of bandpass stationary stochastic processes. Then, we present a vector space representation of signals. We conclude with the representation of digitally modulated signals and their spectral characteristics.

4-1 REPRESENTATION OF BANDPASS SIGNALS AND SYSTEMS

Many digital information-bearing signals are transmitted by some type of carrier modulation. The channel over which the signal is transmitted is limited in bandwidth to an interval of frequencies centered about the carrier, as in double-sideband modulation, or adjacent to the carrier, as in single-sideband modulation. Signals and channels (systems) that satisfy the condition that their bandwidth is much smaller than the carrier frequency are termed *narrowband bandpass signals and channels (systems)*. The modulation performed at the

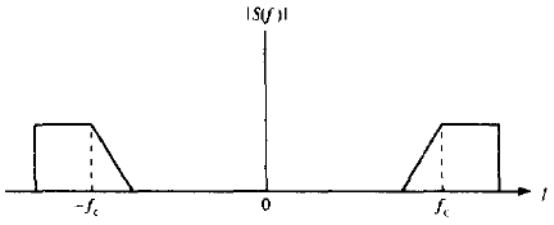


FIGURE 4-1-1 Spectrum of a bandpass signal.

transmitting end of the communication system to generate the bandpass signal and the demodulation performed at the receiving end to recover the digital information involve frequency translations. With no loss of generality and for mathematical convenience, it is desirable to reduce all bandpass signals and channels to equivalent lowpass signals and channels. As a consequence, the results of the performance of the various modulation and demodulation techniques presented in the subsequent chapters are independent of carrier frequencies and channel frequency bands. The representation of bandpass signals and systems in terms of equivalent lowpass waveforms and the characterization of bandpass stationary stochastic processes are the main topics of this section.

4-1-1* Representation of Bandpass Signals

Suppose that a real-valued signal $s(t)$ has a frequency content concentrated in a narrow band of frequencies in the vicinity of a frequency f_c, as shown in Fig. 4-1-1. Our objective is to develop a mathematical representation of such signals. First, we construct a signal that contains only the positive frequencies in $s(t)$. Such a signal may be expressed as

$$S_+(f) = 2u(f)S(f) \tag{4-1-1}$$

where $S(f)$ is the Fourier transform of $s(t)$ and $u(f)$ is the unit step function. The equivalent time-domain expression for (4-1-1) is

$$\begin{aligned} s_+(t) &= \int_{-\infty}^{\infty} S_+(f)e^{j2\pi ft}\,df \\ &= F^{-1}[2u(f)] \bigstar F^{-1}[S(f)] \end{aligned} \tag{4-1-2}$$

The signal $s_+(t)$ is called the *analytic signal* or the *pre-envelope* of $s(t)$. We note that $F^{-1}[S(f)] = s(t)$ and

$$F^{-1}[2u(f)] = \delta(t) + \frac{j}{\pi t} \tag{4-1-3}$$

Hence,

$$s_+(t) = \left[\delta(t) + \frac{j}{\pi t}\right] \bigstar s(t)$$

$$= s(t) + j\frac{1}{\pi t} \bigstar s(t) \tag{4-1-4}$$

We define $\hat{s}(t)$ as

$$\hat{s}(t) = \frac{1}{\pi t} \bigstar s(t)$$

$$= \frac{1}{\pi}\int_{-\infty}^{\infty} \frac{s(\tau)}{t-\tau}\,d\tau \tag{4-1-5}$$

The signal $\hat{s}(t)$ may be viewed as the output of the filter with impulse response

$$h(t) = \frac{1}{\pi t}, \qquad -\infty < t < \infty \tag{4-1-6}$$

when excited by the input signal $s(t)$. Such a filter is called a *Hilbert transformer*. The frequency response of this filter is simply

$$H(f) = \int_{-\infty}^{\infty} h(t)e^{-j2\pi ft}\,dt$$

$$= \frac{1}{\pi}\int_{-\infty}^{\infty} \frac{1}{t} e^{-j2\pi ft}\,dt$$

$$= \begin{cases} -j & (f > 0) \\ 0 & (f = 0) \\ j & (f < 0) \end{cases} \tag{4-1-7}$$

We observe that $|H(f)| = 1$ and that the phase response $\Theta(f) = -\frac{1}{2}\pi$ for $f > 0$ and $\Theta(f) = \frac{1}{2}\pi$ for $f < 0$. Therefore, this filter is basically a 90° phase shifter for all frequencies in the input signal.

The analytic signal $s_+(t)$ is a bandpass signal. We may obtain an equivalent lowpass representation by performing a frequency translation of $S_+(f)$. Thus, we define $S_l(f)$ as

$$S_l(f) = S_+(f + f_c) \tag{4-1-8}$$

The equivalent time-domain relation is

$$s_l(t) = s_+(t)e^{-j2\pi f_c t}$$

$$= [s(t) + j\hat{s}(t)]e^{-j2\pi f_c t} \tag{4-1-9}$$

or, equivalently,

$$s(t) + j\hat{s}(t) = s_l(t)e^{j2\pi f_c t} \tag{4-1-10}$$

In general, the signal $s_l(t)$ is complex-valued (see Problem 4-5), and may be expressed as

$$s_l(t) = x(t) + jy(t) \tag{4-1-11}$$

If we substitute for $s_l(t)$ in (4-1-11) and equate real and imaginary parts on each side, we obtain the relations

$$s(t) = x(t)\cos 2\pi f_c t - y(t)\sin 2\pi f_c t \tag{4-1-12}$$

$$\hat{s}(t) = x(t)\sin 2\pi f_c t + y(t)\cos 2\pi f_c t \tag{4-1-13}$$

The expression (4-1-12) is the desired form for the representation of a bandpass signal. The low-frequency signal components $x(t)$ and $y(t)$ may be viewed as amplitude modulations impressed on the carrier components $\cos 2\pi f_c t$ and $\sin 2\pi f_c t$, respectively. Since these carrier components are in phase quadrature, $x(t)$ and $y(t)$ are called the *quadrature components* of the bandpass signal $s(t)$.

Another representation of the signal in (4-1-12) is

$$\begin{aligned} s(t) &= \text{Re}\{[x(t) + jy(t)]e^{j2\pi f_c t}\} \\ &= \text{Re}[s_l(t)e^{j2\pi f_c t}] \end{aligned} \tag{4-1-14}$$

where Re denotes the real part of the complex-valued quantity in the brackets following. The lowpass signal $s_l(t)$ is usually called the *complex envelope* of the real signal $s(t)$, and is basically the *equivalent lowpass signal.*

Finally, a third possible representation of a bandpass signal is obtained by expressing $s_l(t)$ as

$$s_l(t) = a(t)e^{j\theta(t)} \tag{4-1-15}$$

where

$$a(t) = \sqrt{x^2(t) + y^2(t)} \tag{4-1-16}$$

$$\theta(t) = \tan^{-1}\frac{y(t)}{x(t)} \tag{4-1-17}$$

Then

$$\begin{aligned} s(t) &= \text{Re}[s_l(t)e^{j2\pi f_c t}] \\ &= \text{Re}[a(t)e^{j[2\pi f_c t + \theta(t)]}] \\ &= a(t)\cos[2\pi f_c t + \theta(t)] \end{aligned} \tag{4-1-18}$$

The signal $a(t)$ is called the *envelope* of $s(t)$, and $\theta(t)$ is called the *phase* of $s(t)$. Therefore, (4-1-12), (4-1-14), and (4-1-18) are equivalent representations of bandpass signals.

The Fourier transform of $s(t)$ is

$$\begin{aligned} S(f) &= \int_{-\infty}^{\infty} s(t)e^{-j2\pi ft}\,dt \\ &= \int_{-\infty}^{\infty} \{\text{Re}[s_l(t)e^{j2\pi f_c t}]\}e^{-j2\pi ft}\,dt \end{aligned} \tag{4-1-19}$$

Use of the identity

$$\text{Re}(\xi) = \tfrac{1}{2}(\xi + \xi^*) \tag{4-1-20}$$

in (4-1-19) yields the result

$$S(f) = \frac{1}{2}\int_{-\infty}^{\infty} [s_l(t)e^{j2\pi f_c t} + s_l^*(t)e^{-j2\pi f_c t}]e^{-j2\pi f t}\,dt$$
$$= \tfrac{1}{2}[S_l(f - f_c) + S_l^*(-f - f_c)] \qquad (4\text{-}1\text{-}21)$$

where $S_l(f)$ is the Fourier transform of $s_l(t)$. This is the basic relationship between the spectrum of the real bandpass signal $S(f)$ and the spectrum of the equivalent lowpass signal $S_l(f)$.

The energy in the signal $s(t)$ is defined as

$$\mathscr{E} = \int_{-\infty}^{\infty} s^2(t)\,dt$$
$$= \int_{-\infty}^{\infty} \{\mathrm{Re}\,[s_l(t)e^{j2\pi f_c t}]\}^2\,dt \qquad (4\text{-}1\text{-}22)$$

When the identity in (4-1-20) is used in (4-1-22), we obtain the following result:

$$\mathscr{E} = \frac{1}{2}\int_{-\infty}^{\infty} |s_l(t)|^2\,dt$$
$$+ \frac{1}{2}\int_{-\infty}^{\infty} |s_l(t)|^2 \cos[4\pi f_c t + 2\theta(t)]\,dt \qquad (4\text{-}1\text{-}23)$$

Consider the second integral in (4-1-23). Since the signal $s(t)$ is narrowband, the real envelope $a(t) \equiv |s_l(t)|$ or, equivalently, $a^2(t)$ varies slowly relative to the rapid variations exhibited by the cosine function. A graphical illustration of the integrand in the second integral of (4-1-23) is shown in Fig. 4-1-2. The value of the integral is just the net area under the cosine function modulated by $a^2(t)$. Since the modulating waveform $a^2(t)$ varies slowly relative to the cosine function, the net area contributed by the second integral is very small relative to the value of the first integral in (4-1-23) and, hence, it can be

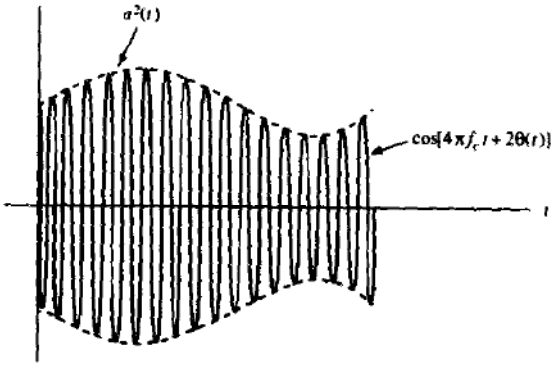


FIGURE 4-1-2 The signal $a^2(t)\cos[4\pi f_c t + 2\theta(t)]$.

neglected. Thus, for all practical pruposes, the energy in the bandpass signal $s(t)$, expressed in terms of the equivalent lowpass signal $s_l(t)$, is

$$\mathcal{E} = \frac{1}{2}\int_{-\infty}^{\infty} |s_l(t)|^2 \, dt \tag{4-1-24}$$

where $|s_l(t)|$ is just the envelope $a(t)$ of $s(t)$.

4-1-2 Representation of Linear Bandpass Systems

A linear filter or system may be described either by its impulse response $h(t)$ or by its frequency response $H(f)$, which is the Fourier transform of $h(t)$. Since $h(t)$ is real,

$$H^*(-f) = H(f) \tag{4-1-25}$$

Let us define $H_l(f - f_c)$ as

$$H_l(f - f_c) = \begin{cases} H(f) & (f > 0) \\ 0 & (f < 0) \end{cases} \tag{4-1-26}$$

Then

$$H_l^*(-f - f_c) = \begin{cases} 0 & (f > 0) \\ H^*(-f) & (f < 0) \end{cases} \tag{4-1-27}$$

Using (4-1-25), we have

$$H(f) = H_l(f - f_c) + H_l^*(-f - f_c) \tag{4-1-28}$$

which resembles (4-1-21) except for the factor $\frac{1}{2}$. The inverse transform of $H(f)$ in (4-1-28) yields $h(t)$ in the form

$$\begin{aligned} h(t) &= h_l(t)e^{j2\pi f_c t} + h_l^*(t)e^{-j2\pi f_c t} \\ &= 2\,\mathrm{Re}\,[h_l(t)e^{j2\pi f_c t}] \end{aligned} \tag{4-1-29}$$

where $h_l(t)$ is the inverse Fourier transform of $H_l(f)$. In general, the impulse response $h_l(t)$ of the equivalent lowpass system is complex-valued.

4-1-3 Response of a Bandpass System to a Bandpass Signal

In Sections 4-1-1 and 4-1-2, we have shown that narrowband bandpass signals and systems can be represented by equivalent lowpass signals and systems. In this section, we demonstrate that the output of a bandpass system to a

bandpass input signal is simply obtained from the equivalent lowpass input signal and the equivalent lowpass impulse response of the system.

Suppose that $s(t)$ is a narrowband bandpass signal and $s_l(t)$ is the equivalent lowpass signal. This signal excites a narrowband bandpass system characterized by its bandpass impulse response $h(t)$ or by its equivalent lowpass impulse response $h_l(t)$. The output of the bandpass system is also a bandpass signal, and, therefore, it can be expressed in the form

$$r(t) = \text{Re}\,[r_l(t)e^{j2\pi f_c t}] \tag{4-1-30}$$

where $r(t)$ is related to the input signal $s(t)$ and the impulse response $h(t)$ by the convolution integral

$$r(t) = \int_{-\infty}^{\infty} s(\tau)h(t-\tau)\,d\tau \tag{4-1-31}$$

Equivalently, the output of the system, expressed in the frequency domain, is

$$R(f) = S(f)H(f) \tag{4-1-32}$$

Substituting from (4-1-21) for $S(f)$ and from (4-1-28) for $H(f)$, we obtain the result

$$R(f) = \tfrac{1}{2}[S_l(f-f_c) + S_l^*(-f-f_c)][H_l(f-f_c) + H_l^*(-f-f_c)] \tag{4-1-33}$$

When $s(t)$ is a narrowband signal and $h(t)$ is the impulse response of a narrowband system, $S_l(f-f_c) \approx 0$ and $H_l(f-f_c) = 0$ for $f < 0$. It follows from this narrowband condition that

$$S_l(f-f_c)H_l^*(-f-f_c) = 0, \qquad S_l^*(-f-f_c)H_l(f-f_c) = 0$$

Therefore, (4-1-33) simplifies to

$$\begin{aligned} R(f) &= \tfrac{1}{2}[S_l(f-f_c)H_l(f-f_c) + S_l^*(-f-f_c)H_l^*(-f-f_c)] \\ &= \tfrac{1}{2}[R_l(f-f_c) + R_l^*(-f-f_c)] \end{aligned} \tag{4-1-34}$$

where

$$R_l(f) = S_l(f)H_l(f) \tag{4-1-35}$$

is the output spectrum of the equivalent lowpass system excited by the equivalent lowpass signal. It is clear that the time domain relation for the output $r_l(t)$ is given by the convolution of $s_l(t)$ with $h_l(t)$. That is,

$$r_l(t) = \int_{-\infty}^{\infty} s_l(\tau)h_l(t-\tau)\,d\tau \tag{4-1-36}$$

The combination of (4-1-36) with (4-1-30) gives the relationship between the bandpass output signal $r(t)$ and the equivalent lowpass time functions $s_l(t)$ and $h_l(t)$. This simple relationship allows us to ignore any linear frequency translations encountered in the modulation of a signal for purposes of matching its spectral content to the frequency allocation of a particular channel. Thus, for mathematical convenience, we shall deal only with the transmission of equivalant lowpass signals through equivalent lowpass channels.

4-1-4 Representation of Bandpass Stationary Stochastic Processes

The representation of bandpass signals presented in Section 4-1-1 applied to deterministic signals. In this section, we extend the representation to sample functions of a bandpass stationary stochastic process. In particular, we derive the important relations between the correlation functions and power spectra of the bandpass signal and the correlation functions and power spectra of the equivalent lowpass signal.

Suppose that $n(t)$ is a sample function of a wide-sense stationary stochastic process with zero mean and power spectral density $\Phi_{nn}(f)$. The power spectral density is assumed to be zero outside of an interval of frequencies centered around $\pm f_c$, where f_c is termed the *carrier frequency*. The stochastic process $n(t)$ is said to be a *narrowband bandpass process* if the width of the spectral density is much smaller than f_c. Under this condition, a sample function of the process $n(t)$ can be represented by any of the three equivalent forms given in Section 4-1-1, namely,

$$n(t) = a(t)\cos[2\pi f_c t + \theta(t)] \tag{4-1-37}$$

$$= x(t)\cos 2\pi f_c t - y(t)\sin 2\pi f_c t \tag{4-1-38}$$

$$= \text{Re}\,[z(t)e^{j2\pi f_c t}] \tag{4-1-39}$$

where $a(t)$ is the envelope and $\theta(t)$ is the phase of the real-valued signal, $x(t)$ and $y(t)$ are the quadrature components of $n(t)$, and $z(t)$ is called the *complex envelope of* $n(t)$.

Let us consider the form given by (4-1-38) in more detail. First, we observe that if $n(t)$ is zero mean, then $x(t)$ and $y(t)$ must also have zero mean values. In addition, the stationarity of $n(t)$ implies that the autocorrelation and cross-correlation functions of $x(t)$ and $y(t)$ satisfy the following properties:

$$\phi_{xx}(\tau) = \phi_{yy}(\tau) \tag{4-1-40}$$

$$\phi_{xy}(\tau) = -\phi_{yx}(\tau) \tag{4-1-41}$$

That these two properties follow from the stationarity of $n(t)$ is now demonstrated. The autocorrelation function $\phi_{nn}(\tau)$ of $n(t)$ is

$$\begin{aligned} E[n(t)n(t+\tau)] &= E\{[x(t)\cos 2\pi f_c t - y(t)\sin 2\pi f_c t] \\ &\quad \times [x(t+\tau)\cos 2\pi f_c(t+\tau) \\ &\quad\quad - y(t+\tau)\sin 2\pi f_c(t+\tau)]\} \\ &= \phi_{xx}(\tau)\cos 2\pi f_c t \cos 2\pi f_c(t+\tau) \\ &\quad + \phi_{yy}(\tau)\sin 2\pi f_c t \sin 2\pi f_c(t+\tau) \\ &\quad - \phi_{xy}(\tau)\sin 2\pi f_c t \cos 2\pi f_c(t+\tau) \\ &\quad - \phi_{yx}(\tau)\cos 2\pi f_c t \sin 2\pi f_c(t+\tau) \end{aligned} \tag{4-1-42}$$

Use of the trigonometric identities

$$\begin{aligned} \cos A \cos B &= \tfrac{1}{2}[\cos(A-B) + \cos(A+B)] \\ \sin A \sin B &= \tfrac{1}{2}[\cos(A-B) - \cos(A+B)] \\ \sin A \cos B &= \tfrac{1}{2}[\sin(A-B) + \sin(A+B)] \end{aligned} \tag{4-1-43}$$

in (4-1-42) yields the result

$$\begin{aligned} E[n(t)n(t+\tau)] &= \tfrac{1}{2}[\phi_{xx}(\tau) + \phi_{yy}(\tau)]\cos 2\pi f_c \tau \\ &\quad + \tfrac{1}{2}[\phi_{xx}(\tau) - \phi_{yy}(\tau)]\cos 2\pi f_c(2t+\tau) \\ &\quad - \tfrac{1}{2}[\phi_{yx}(\tau) - \phi_{xy}(\tau)]\sin 2\pi f_c \tau \\ &\quad - \tfrac{1}{2}[\phi_{yx}(\tau) + \phi_{xy}(\tau)]\sin 2\pi f_c(2t+\tau) \end{aligned} \tag{4-1-44}$$

Since $n(t)$ is stationary, the right-hand side of (4-1-44) must be independent of t. But this condition can only be satisfied if (4-1-40) and (4-1-41) hold. As a consequence, (4-1-44) reduces to

$$\phi_{nn}(\tau) = \phi_{xx}(\tau)\cos 2\pi f_c \tau - \phi_{yx}(\tau)\sin 2\pi f_c \tau \tag{4-1-45}$$

We note that the relation between the autocorrelation function $\phi_{nn}(\tau)$ of the bandpass process and the autocorrelation and cross-correlation functions $\phi_{xx}(\tau)$ and $\phi_{yx}(\tau)$ of the quadrature components is identical in form to (4-1-38), which expresses the bandpass process in terms of the quadrature components.

The autocorrelation function of the equivalent lowpass process

$$z(t) = x(t) + jy(t) \tag{4-1-46}$$

is defined as

$$\phi_{zz}(\tau) = \tfrac{1}{2}E[z^*(t)z(t+\tau)] \tag{4-1-47}$$

Substituting (4-1-46) into (4-1-47) and performing the expectation operation, we obtain

$$\phi_{zz}(\tau) = \tfrac{1}{2}[\phi_{xx}(\tau) + \phi_{yy}(\tau) - j\phi_{xy}(\tau) + j\phi_{yx}(\tau)] \tag{4-1-48}$$

Now if the symmetry properties given in (4-1-40) and (4-1-41) are used in (4-1-48), we obtain

$$\phi_{zz}(\tau) = \phi_{xx}(\tau) + j\phi_{yx}(\tau) \tag{4-1-49}$$

which relates the autocorrelation function of the complex envelope to the autocorrelation and cross-correlation functions of the quadrature components. Finally, we incorporate the result given by (4-1-49) into (4-1-45), and we have

$$\phi_{nn}(\tau) = \text{Re}\,[\phi_{zz}(\tau)e^{j2\pi f_c\tau}] \tag{4-1-50}$$

Thus, the autocorrelation function $\phi_{nn}(\tau)$ of the bandpass stochastic process is uniquely determined from the autocorrelation function $\phi_{zz}(\tau)$ of the equivalent lowpass process $z(t)$ and the carrier frequency f_c.

The power density spectrum $\Phi_{nn}(f)$ of the stochastic process $n(t)$ is the Fourier transform of $\phi_{nn}(\tau)$. Hence,

$$\begin{aligned}\Phi_{nn}(f) &= \int_{-\infty}^{\infty} \{\text{Re}\,[\phi_{zz}(\tau)e^{j2\pi f_c\tau}]\}e^{-j2\pi f\tau}\,d\tau \\ &= \tfrac{1}{2}[\Phi_{zz}(f - f_c) + \Phi_{zz}(-f - f_c)]\end{aligned} \tag{4-1-51}$$

where $\Phi_{zz}(f)$ is the power density spectrum of the equivalent lowpass process $z(t)$. Since the autocorrelation function of $z(t)$ satisfies the property $\phi_{zz}(\tau) = \phi^*_{zz}(-\tau)$, it follows that $\Phi_{zz}(f)$ is a real-valued function of frequency.

Properties of the Quadrature Components It was just demonstrated above that the cross-correlation function of the quadrature components $x(t)$ and $y(t)$ of the bandpass stationary stochastic process $n(t)$ satisfies the symmetry condition in (4-1-41). Furthermore, any cross-correlation function satisfies the condition

$$\phi_{yx}(\tau) = \phi_{xy}(-\tau) \tag{4-1-52}$$

From these two conditions, we conclude that

$$\phi_{xy}(\tau) = -\phi_{xy}(-\tau) \tag{4-1-53}$$

That is, $\phi_{xy}(\tau)$ is an odd function of τ. Consequently, $\phi_{xy}(0) = 0$, and, hence, $x(t)$ and $y(t)$ are uncorrelated (for $\tau = 0$, only). Of course, this does not mean that the processes $x(t)$ and $y(t + \tau)$ are uncorrelated for all τ, since that would imply that $\phi_{xy}(\tau) = 0$ for all τ. If, indeed, $\phi_{xy}(\tau) = 0$ for all τ, then $\phi_{zz}(\tau)$ is real and the power spectral density $\Phi_{zz}(f)$ satisfies the condition

$$\Phi_{zz}(f) = \Phi_{zz}(-f) \tag{4-1-54}$$

and vice versa. That is, $\Phi_{zz}(f)$ is symmetric about $f = 0$.

In the special case in which the stationary stochastic process $n(t)$ is gaussian, the quadrature components $x(t)$ and $y(t+\tau)$ are jointly gaussian. Moreover, for $\tau = 0$, they are statistically independent, and, hence, their joint probability density function is

$$p(x, y) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2} \tag{4-1-55}$$

where the variance σ^2 is defined as $\sigma^2 = \phi_{xx}(0) = \phi_{yy}(0) = \phi_{nn}(0)$.

Representation of White Noise White noise is a stochastic process that is defined to have a flat (constant) power spectral density over the entire frequency range. This type of noise cannot be expressed in terms of quadrature components, as a result of its wideband character.

In problems concerned with the demodulation of narrowband signals in noise, it is mathematically convenient to model the additive noise process as white and to represent the noise in terms of quadrature components. This can be accomplished by postulating that the signals and noise at the receiving terminal have passed through an ideal bandpass filter, having a passband that includes the spectrum of the signals but is much wider. Such a filter will introduce negligible, if any, distortion on the signal but it does eliminate the noise frequency components outside of the passband.

The noise resulting from passing the white noise process through a spectrally flat (ideal) bandpass filter is termed *bandpass white noise* and has the power spectral density depicted in Fig. 4-1-3. Bandpass white noise can be represented by any of the forms given in (4-1-37), (4-1-38), and (4-1-39). The equivalent lowpass noise $z(t)$ has a power spectral density

$$\Phi_{zz}(f) = \begin{cases} N_0 & (|f| \leq \frac{1}{2}B) \\ 0 & (|f| > \frac{1}{2}B) \end{cases} \tag{4-1-56}$$

and its autocorrelation function is

$$\phi_{zz}(\tau) = N_0 \frac{\sin \pi B\tau}{\pi\tau} \tag{4-1-57}$$

The limiting form of $\phi_{zz}(\tau)$ as B approaches infinity is

$$\phi_{zz}(\tau) = N_0\delta(\tau) \tag{4-1-58}$$

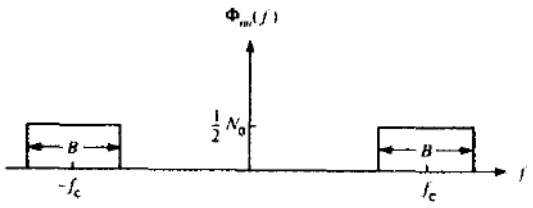


FIGURE 4-1-3 Bandpass noise with a flat spectrum.

The power spectral density for white noise and bandpass white noise is symmetric about $f = 0$, so $\phi_{xy}(\tau) = 0$ for all τ. Therefore,

$$\phi_{zz}(\tau) = \phi_{xx}(\tau) = \phi_{yy}(\tau) \tag{4-1-59}$$

That is, the quadrature components $x(t)$ and $y(t)$ are uncorrelated for all time shifts τ and the autocorrelation functions of $z(t)$, $x(t)$, and $y(t)$ are all equal.

4-2 SIGNAL SPACE REPRESENTATIONS

In this section, we demonstrate that signals have characteristics that are similar to vectors and develop a vector representation for signal waveforms. We begin with some basic definitions and concepts involving vectors.

4-2-1 Vector Space Concepts

A vector $\mathbf{v}$ in an n-dimensional space is characterized by its n components $[v_1 \ v_2 \ \dots \ v_n]$. It may also be represented as a linear combination of *unit vectors* or *basis vectors* $\mathbf{e}_i$, $1 \leq i \leq n$, i.e.,

$$\mathbf{v} = \sum_{i=1}^{n} v_i \mathbf{e}_i \tag{4-2-1}$$

where, by definition, a unit vector has length unity and v_i is the projection of the vector $\mathbf{v}$ onto the unit vector $\mathbf{e}_i$.

The *inner product* of two n-dimensional vectors $\mathbf{v}_1 = [v_{11} \ v_{12} \ \dots \ v_{1n}]$ and $\mathbf{v}_2 = [v_{21} \ v_{22} \ \dots \ v_{2n}]$ is defined as

$$\mathbf{v}_1 \cdot \mathbf{v}_2 = \sum_{i=1}^{n} v_{1i} v_{2i} \tag{4-2-2}$$

Two vectors $\mathbf{v}_1$ and $\mathbf{v}_2$ are orthogonal if $\mathbf{v}_1 \cdot \mathbf{v}_2 = 0$. More generally, a set of m vectors $\mathbf{v}_k$, $1 \leq k \leq m$, are orthogonal if

$$\mathbf{v}_i \cdot \mathbf{v}_j = 0 \tag{4-2-3}$$

for all $1 \leq i, j \leq m$ and $i \neq j$.

The *norm* of a vector $\mathbf{v}$ is denoted by $\|\mathbf{v}\|$ and is defined as

$$\|\mathbf{v}\| = (\mathbf{v} \cdot \mathbf{v})^{1/2} = \sqrt{\sum_{i=1}^{n} v_i^2} \tag{4-2-4}$$

which is simply its length. A set of m vectors is said to be *orthonormal* if the vectors are orthogonal and each vector has a unit norm. A set of m vectors is said to be *linearly independent* if no one vector can be represented as a linear combination of the remaining vectors.

Two n-dimensional vectors $\mathbf{v}_1$ and $\mathbf{v}_2$ satisfy the *triangle inequality*

$$\|\mathbf{v}_1 + \mathbf{v}_2\| \leq \|\mathbf{v}_1\| + \|\mathbf{v}_2\| \tag{4-2-5}$$

with equality if $\mathbf{v}_1$ and $\mathbf{v}_2$ are in the same direction, i.e., $\mathbf{v}_1 = a\mathbf{v}_2$ where a is a

positive real scalar. From the triangle inequality there follows the *Cauchy–Schwartz inequality*

$$|\mathbf{v}_1 \cdot \mathbf{v}_2| \leq \|\mathbf{v}_1\| \, \|\mathbf{v}_2\| \tag{4-2-6}$$

with equality if $\mathbf{v}_1 = a\mathbf{v}_2$. The norm square of the sum of two vectors may be expressed as

$$\|\mathbf{v}_1 + \mathbf{v}_2\|^2 = \|\mathbf{v}_1\|^2 + \|\mathbf{v}_2\|^2 + 2\mathbf{v}_1 \cdot \mathbf{v}_2 \tag{4-2-7}$$

If $\mathbf{v}_1$ and $\mathbf{v}_2$ are orthogonal then $\mathbf{v}_1 \cdot \mathbf{v}_2 = 0$ and, hence,

$$\|\mathbf{v}_1 + \mathbf{v}_2\|^2 = \|\mathbf{v}_1\|^2 + \|\mathbf{v}_2\|^2 \tag{4-2-8}$$

This is the Pythagorean relation for two orthogonal n-dimensional vectors.

From matrix algebra, we recall that a linear transformation in an n-dimensional vector space is a matrix transformation of the form

$$\mathbf{v}' = \mathbf{A}\mathbf{v} \tag{4-2-9}$$

where the matrix $\mathbf{A}$ transforms the vector $\mathbf{v}$ into some vector $\mathbf{v}'$. In the special case where $\mathbf{v}' = \lambda\mathbf{v}$, i.e.,

$$\mathbf{A}\mathbf{v} = \lambda\mathbf{v} \tag{4-2-10}$$

where λ is some (positive or negative) scalar, the vector $\mathbf{v}$ is called an *eigenvector* of the transformation and λ is the corresponding *eigenvalue.*

Finally, let us review the Gram–Schmidt procedure for constructing a set of orthonormal vectors from a set of n-dimensional vectors $\mathbf{v}_i$, $1 \leq i \leq m$. We begin by arbitrarily selecting a vector from the set, say $\mathbf{v}_1$. By normalizing its length, we obtain the first vector, say

$$\mathbf{u}_1 = \frac{\mathbf{v}_1}{\|\mathbf{v}_1\|} \tag{4-2-11}$$

Next, we may select $\mathbf{v}_2$ and, first, subtract the projection of $\mathbf{v}_2$ onto $\mathbf{u}_1$. Thus, we obtain

$$\mathbf{u}_2' = \mathbf{v}_2 - (\mathbf{v}_2 \cdot \mathbf{u}_1)\mathbf{u}_1 \tag{4-2-12}$$

Then, we normalize the vector $\mathbf{u}_2'$ to unit length. This yields

$$\mathbf{u}_2 = \frac{\mathbf{u}_2'}{\|\mathbf{u}_2'\|} \tag{4-2-13}$$

The procedure continues by selecting $\mathbf{v}_3$ and subtracting the projections of $\mathbf{v}_3$ into $\mathbf{u}_1$ and $\mathbf{u}_2$. Thus, we have

$$\mathbf{u}_3' = \mathbf{v}_3 - (\mathbf{v}_3 \cdot \mathbf{u}_1)\mathbf{u}_1 - (\mathbf{v}_3 \cdot \mathbf{u}_2)\mathbf{u}_2 \tag{4-2-14}$$

Then, the orthonormal vector $\mathbf{u}_3$ is

$$\mathbf{u}_3 = \frac{\mathbf{u}_3'}{\|\mathbf{u}_3'\|} \tag{4-2-15}$$

By continuing this procedure, we shall construct a set of n_1, orthonormal vectors, where $n_1 \leq n$, in general. If $m < n$ then $n_1 \leq m$, and if $m \geq n$ then $n_1 \leq n$.

4-2-2 Signal Space Concepts

As in the case of vectors, we may develop a parallel treatment for a set of signals defined on some interval $[a, b]$. The *inner product* of two generally complex-valued signals $x_1(t)$ and $x_2(t)$ is denoted by $\langle x_1(t), x_2(t)\rangle$ and defined as

$$\langle x_1(t), x_2(t)\rangle = \int_a^b x_1(t)x_2^*(t)\,dt \tag{4-2-16}$$

The signals are orthogonal if their inner product is zero.

The *norm* of a signal is defined as

$$\|x(t)\| = \left(\int_a^b |x(t)|^2\,dt\right)^{1/2} \tag{4-2-17}$$

A set of m signals are *orthonormal* if they are orthogonal and their norms are all unity. A set of m signals is *linearly independent*, if no signal can be represented as a linear combination of the remaining signals.

The *triangle inequality* for two signals is simply

$$\|x_1(t) + x_2(t)\| \leq \|x_1(t)\| + \|x_2(t)\| \tag{4-2-18}$$

and the *Cauchy–Schwartz inequality* is

$$\left|\int_a^b x_1(t)x_2^*(t)\,dt\right| \leq \left|\int_a^b |x_1(t)|^2\,dt\right|^{1/2} \left|\int_a^b |x_2(t)|^2\,dt\right|^{1/2} \tag{4-2-19}$$

with equality when $x_2(t) = ax_1(t)$, where a is any complex number.

4-2-3 Orthogonal Expansions of Signals

In this section, we develop a vector representation for signal waveforms, and, thus, we demonstrate an equivalence between a signal waveform and its vector representation.

Suppose that $s(t)$ is a deterministic, real-valued signal with finite energy

$$\mathcal{E}_s = \int_{-\infty}^{\infty} [s(t)]^2\,dt \tag{4-2-20}$$

Furthermore, suppose that there exists a set of functions $\{f_n(t), n = 1, 2, \ldots, N\}$ that are orthonormal in the sense that

$$\int_{-\infty}^{\infty} f_n(t)f_m(t)\,dt = \begin{cases} 0 & (m \neq n) \\ 1 & (m = n) \end{cases} \tag{4-2-21}$$

We may approximate the signal $s(t)$ by a weighted linear combination of these functions, i.e.,

$$\hat{s}(t) = \sum_{k=1}^{K} s_k f_k(t) \tag{4-2-22}$$

where $\{s_k,\ 1 \leq k \leq K\}$ are the coefficients in the approximation of $s(t)$. The approximation error incurred is

$$e(t) = s(t) - \hat{s}(t) \tag{4-2-23}$$

Let us select the coefficients $\{s_k\}$ so as to minimize the energy $\mathscr{E}_e$ of the approximation error. Thus,

$$\begin{aligned} \mathscr{E}_e &= \int_{-\infty}^{\infty} [s(t) - \hat{s}(t)]^2\, dt \\ &= \int_{-\infty}^{\infty} \left[s(t) - \sum_{k=1}^{K} s_k f_k(t) \right]^2 dt \end{aligned} \tag{4-2-24}$$

The optimum coefficients in the series expansion of $s(t)$ may be found by differentiating (4-2-24) with respect to each of the coefficients $\{s_k\}$ and setting the first derivatives to zero. Alternatively, we may use a well-known result from estimation theory based on the mean-square-error criterion, which, simply stated, is that the minimum of $\mathscr{E}_e$ with respect to the $\{s_k\}$ is obtained when the error is orthogonal to each of the functions in the series expansion. Thus,

$$\int_{-\infty}^{\infty} \left[s(t) - \sum_{k=1}^{K} s_k f_k(t) \right] f_n(t)\, dt = 0, \qquad n = 1, 2, \ldots, K \tag{4-2-25}$$

Since the functions $\{f_n(t)\}$ are orthonormal, (4-2-25) reduces to

$$s_n = \int_{-\infty}^{\infty} s(t) f_n(t)\, dt, \qquad n = 1, 2, \ldots, K \tag{4-2-26}$$

Thus, the coefficients are obtained by projecting the signal $s(t)$ onto each of the functions $\{f_n(t)\}$. Consequently, $\hat{s}(t)$ is the projection of $s(t)$ onto the K-dimensional signal space spanned by the functions $\{f_n(t)\}$. The minimum mean square approximation error is

$$\begin{aligned} \mathscr{E}_{\min} &= \int_{-\infty}^{\infty} e(t) s(t)\, dt \\ &= \int_{-\infty}^{\infty} [s(t)]^2\, dt - \int_{-\infty}^{\infty} \sum_{k=1}^{K} s_k f_k(t) s(t)\, dt \\ &= \mathscr{E}_s - \sum_{k=1}^{k} s_k^2 \end{aligned} \tag{4-2-27}$$

which is nonnegative, by definition.

When the minimum mean square approximation error $\mathscr{E}_{\min} = 0$,

$$\mathscr{E}_s = \sum_{k=1}^{K} s_k^2 = \int_{-\infty}^{\infty} [s(t)]^2 \, dt \tag{4-2-28}$$

Under the condition that $\mathscr{E}_{\min} = 0$, we may express $s(t)$ as

$$s(t) = \sum_{k=1}^{K} s_k f_k(t) \tag{4-2-29}$$

where it is understood that equality of $s(t)$ to its series expansion holds in the sense that the approximation error has zero energy.

When every finite energy signal can be represented by a series expansion of the form in (4-2-29) for which $\mathscr{E}_{\min} = 0$, the set of orthonormal functions $\{f_n(t)\}$ is said to be *complete*.

Example 4-2-1: Trigonometric Fourier Series

A finite energy signal $s(t)$ that is zero everywhere except in the range $0 \leq t \leq T$ and has a finite number of discontinuities in this interval, can be represented in a Fourier series as

$$s(t) = \sum_{k=0}^{\infty} \left(a_k \cos \frac{2\pi k t}{T} + b_k \sin \frac{2\pi k t}{T} \right) \tag{4-2-30}$$

where the coefficients $\{a_k, b_k\}$ that minimize the mean square error are given by

$$\begin{aligned} a_k &= \frac{1}{\sqrt{T}} \int_0^T s(t) \cos \frac{2\pi k t}{T} \, dt \\ b_k &= \frac{1}{\sqrt{T}} \int_0^T s(t) \sin \frac{2\pi k t}{T} \, dt \end{aligned} \tag{4-2-31}$$

The set of trigonometric functions $\{\sqrt{2/T} \cos 2\pi k t/T, \sqrt{2/T} \sin 2\pi k t/T\}$ is complete, and, hence, the series expansion results in zero mean square error. These properties are easily established from the development given above.

Gram–Schmidt Procedure Now suppose that we have a set of finite energy signal waveforms $\{s_i(t), i = 1, 2, \ldots, M\}$ and we wish to construct a set of orthonormal waveforms. The Gram–Schmidt orthogonalization procedure allows us to construct such a set. We begin with the first waveform $s_1(t)$, which is assumed to have energy $\mathscr{E}_1$. The first waveform is simply constructed as

$$f_1(t) = \frac{s_1(t)}{\sqrt{\mathscr{E}_1}} \tag{4-2-32}$$

Thus, $f_1(t)$ is simply $s_1(t)$ normalized to unit energy.

The second waveform is constructed from $s_2(t)$ by first computing the projection of $f_1(t)$ onto $s_2(t)$, which is

$$c_{12} = \int_{-\infty}^{\infty} s_2(t) f_1(t)\, dt \tag{4-2-33}$$

Then, $c_{12} f_1(t)$ is subtracted from $s_2(t)$ to yield

$$f_2'(t) = s_2(t) - c_{12} f_1(t) \tag{4-2-34}$$

This waveform is orthogonal to $f_1(t)$ but it does not have unit energy. If $\mathscr{E}_2$ denotes the energy of $f_2'(t)$, the normalized waveform that is orthogonal to $f_1(t)$ is

$$f_2(t) = \frac{f_2'(t)}{\sqrt{\mathscr{E}_2}} \tag{4-2-35}$$

In general, the orthogonalization of the kth function leads to

$$f_k(t) = \frac{f_k'(t)}{\sqrt{\mathscr{E}_k}} \tag{4-2-36}$$

where

$$f_k'(t) = s_k(t) - \sum_{i=1}^{k-1} c_{ik} f_i(t) \tag{4-2-37}$$

and

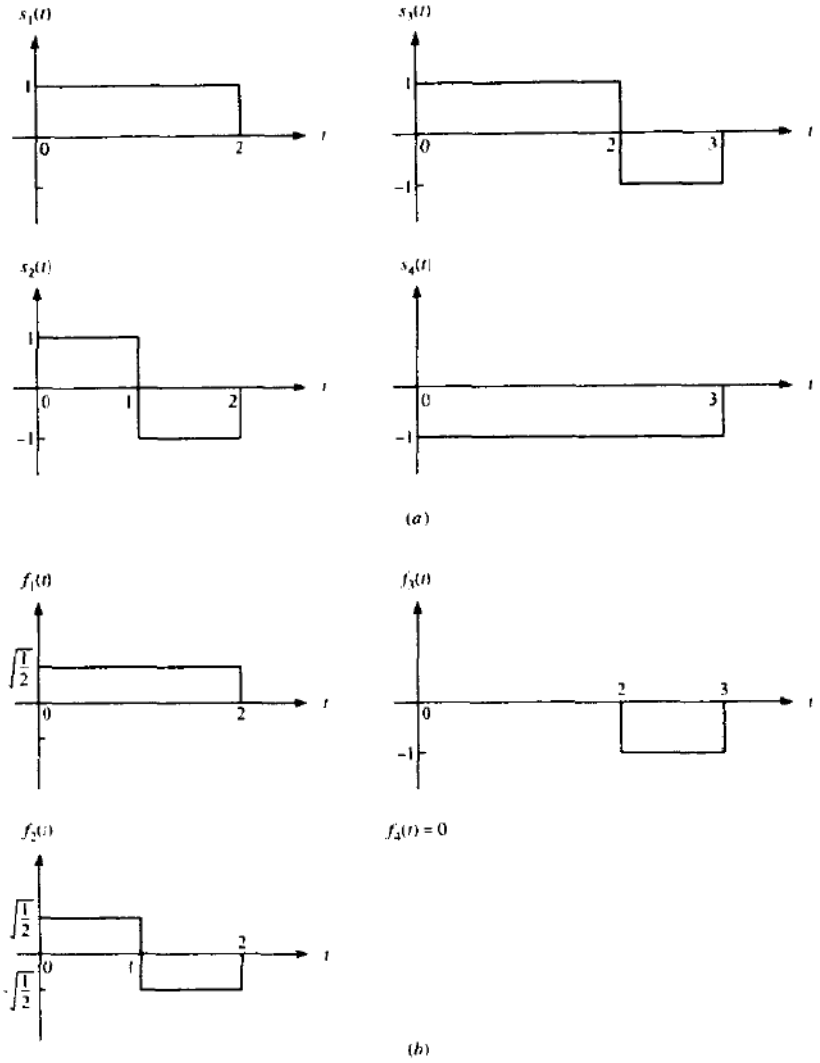
$$c_{ik} = \int_{-\infty}^{\infty} s_k(t) f_i(t)\, dt, \qquad i = 1, 2, \ldots, k-1 \tag{4-2-38}$$

Thus, the orthogonalization process is continued until all the M signal waveforms $\{s_i(t)\}$ have been exhausted and $N \leq M$ orthonormal waveforms have been constructed. The dimensionality N of the signal space will be equal to M if all the signal waveforms are linearly independent, i.e., none of the signals waveforms is a linear combination of the other signal waveforms.

Example 4-2-2

Let us apply the Gram–Schmidt procedure to the set of four waveforms illustrated in Fig. 4-2-1(*a*). The waveform $s_1(t)$ has energy $\mathscr{E}_1 = 2$, so that $f_1(t) = \sqrt{\frac{1}{2}}\, s_1(t)$. Next, we observe that $c_{12} = 0$; hence, $s_2(t)$ and $f_1(t)$ are orthogonal. Therefore, $f_2(t) = s_2(t)/\sqrt{\mathscr{E}_2} = \sqrt{\frac{1}{2}}\, s_2(t)$. To obtain $f_3(t)$, we compute c_{13} and c_{23}, which are $c_{13} = \sqrt{2}$ and $c_{23} = 0$. Thus,

$$f_3'(t) = s_3(t) - \sqrt{2}\, f_1(t) = \begin{cases} -1 & (2 \leq t \leq 3) \\ 0 & \text{(otherwise)} \end{cases}$$



Gram–Schmidt orthogonalization of the signals $\{s_i(t), i = 1, 2, 3, 4\}$ and the corresponding orthogonal signals.

Since $f_3'(t)$ has unit energy, it follows that $f_3(t) = f_3'(t)$. In determining $f_4(t)$, we find that $c_{14} = -\sqrt{2}$, $c_{24} = 0$, and $c_{34} = 1$. Hence,

$$f_4'(t) = s_4(t) + \sqrt{2}\, f_1(t) - f_3(t) = 0$$

Consequently, $s_4(t)$ is a linear combination of $f_1(t)$ and $f_3(t)$ and, hence, $f_4(t) = 0$. The three orthonormal functions are illustrated in Fig. 4-2-1(b).

Once we have constructed the set of orthonormal waveforms $\{f_n(t)\}$, we can express the M signals $\{s_m(t)\}$ as linear combinations of the $\{f_n(t)\}$. Thus, we may write

$$s_k(t) = \sum_{n=1}^{N} s_{kn} f_n(t), \qquad k = 1, 2, \ldots, M \tag{4-2-39}$$

and

$$\mathscr{E}_k = \int_0^T [s_k(t)]^2 dt = \sum_{n=1}^{N} s_{kn}^2 = \|\mathbf{s}_k\|^2 \tag{4-2-40}$$

Based on the expression in (4-2-39), each signal may be represented by the vector

$$\mathbf{s}_k = [s_{k1} \; s_{k2} \; \cdots \; s_{kN}] \tag{4-2-41}$$

or, equivalently, as a point in the N-dimensional signal space with coordinates $\{s_{ki}, i = 1, 2, \ldots, N\}$. The energy in the kth signal is simply the square of the length of the vector or, equivalently, the square of the Euclidean distance from the origin to the point in the N-dimensional space. Thus, any signal can be represented geometrically as a point in the signal space spanned by the orthonormal functions $\{f_n(t)\}$.

Example 4-2-3

Let us obtain the vector representation of the four signals shown in Fig. 4-2-1(a) by using the orthonormal set of functions in Fig. 4-2-1(b). Since the dimensionality of the signal space is $N = 3$, each signal is described by three components. The signal $s_1(t)$ is characterized by the vector $\mathbf{s}_1 = (\sqrt{2}, 0, 0)$. Similarly, the signals $s_2(t)$, $s_3(t)$, and $s_4(t)$ are characterized by the vectors $\mathbf{s}_2 = (0, \sqrt{2}, 0)$, $\mathbf{s}_3 = (\sqrt{2}, 0, 1)$, and $\mathbf{s}_4 = (-\sqrt{2}, 0, 1)$, respectively. These vectors are shown in Fig. 4-2-2. Their lengths are $|\mathbf{s}_1| = \sqrt{2}$, $|\mathbf{s}_2| = \sqrt{2}$,

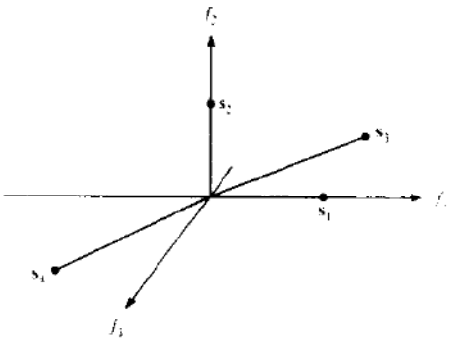


FIGURE 4-2-2 The four signal vectors represented as points in three dimensional function space.

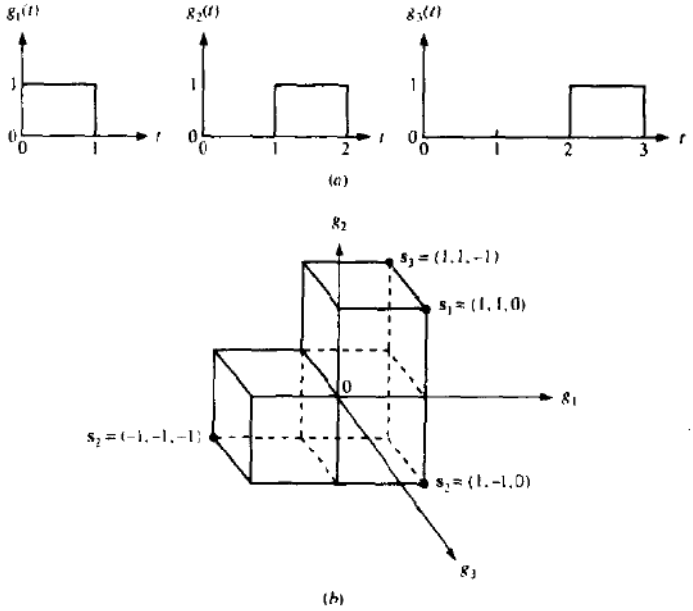
$|\mathbf{s}_3| = \sqrt{3}$, and $|\mathbf{s}_4| = \sqrt{3}$, and the corresponding signal energies are $\mathscr{E}_k = |\mathbf{s}_k|^2$, $k = 1, 2, 3, 4$.

We have demonstrated that a set of M finite energy waveforms $\{s_n(t)\}$ can be represented by a weighted linear combination of orthonormal functions $\{f_n(t)\}$ of dimensionality $N \leq M$. The functions $\{f_n(t)\}$ are obtained by applying the Gram–Schmidt orthogonalization procedure on $\{s_n(t)\}$. It should be emphasized, however, that the functions $\{f_n(t)\}$ obtained from the Gram–Schmidt procedure are not unique. If we alter the order in which the orthogonalization of the signals $\{s_n(t)\}$ is performed, the orthonormal waveforms will be different and the corresponding vector representation of the signals $\{s_n(t)\}$ will depend on the choice of the orthonormal functions $\{f_n(t)\}$. Nevertheless, the vectors $\{\mathbf{s}_n\}$ will retain their geometrical configuration and their lengths will be invariant to the choice of orthonormal functions $\{f_n(t)\}$.

Example 4-2-4

An alternative set of orthonormal functions for the four signals in Fig. 4-2-1 is illustrated in Fig. 4-2-3(a). By using these functions to expand $\{s_n(t)\}$, we

FIGURE 4-2-3 An alternative set of orthonormal functions for the four signals in Fig. 4-2-1(a) and the corresponding signal points.



obtain the corresponding vectors $\mathbf{s}_1 = (1, 1, 0)$, $\mathbf{s}_2 = (1, -1, 0)$, $\mathbf{s}_3 = (1, 1, -1)$, and $\mathbf{s}_4 = (-1, -1, -1)$, which are shown in Fig. 4-2-3(*b*). Note that the vector lengths are identical to those obtained from the orthonormal functions $\{f_n(t)\}$.

The orthogonal expansions described above were developed for real-valued signal waveforms. The extension to complex-valued signal waveforms is left as an exercise for the reader (see Problems 4-6 and 4-7).

Finally, let us consider the case in which the signal waveforms are bandpass and represented as

$$s_m(t) = \text{Re}\,[s_{lm}(t)e^{j2\pi f_c t}], \qquad m = 1, 2, \ldots, M \tag{4-2-42}$$

where $\{s_{lm}(t)\}$ denote the equivalent lowpass signals. Recall that the signal energies may be expressed either in terms of $s_m(t)$ or $s_{lm}(t)$, as

$$\begin{aligned} \mathcal{E}_m &= \int_{-\infty}^{\infty} s_m^2(t)\,dt \\ &= \frac{1}{2}\int_{-\infty}^{\infty} |s_{lm}(t)|^2\,dt \end{aligned} \tag{4-2-43}$$

The similarity between any pair of signal waveforms, say $s_m(t)$ and $s_k(t)$, is measured by the normalized cross-correlation

$$\frac{1}{\sqrt{\mathcal{E}_m \mathcal{E}_k}}\int_{-\infty}^{\infty} s_m(t)s_k(t)\,dt = \text{Re}\left\{\frac{1}{2\sqrt{\mathcal{E}_m \mathcal{E}_k}}\int_{-\infty}^{\infty} s_{lm}(t)s_{lk}^*(t)\,dt\right\} \tag{4-2-44}$$

We define the complex-valued cross-correlation coefficient ρ_{km} as

$$\rho_{km} = \frac{1}{2\sqrt{\mathcal{E}_m \mathcal{E}_k}}\int_{-\infty}^{\infty} s_{lm}^*(t)s_{lk}(t)\,dt \tag{4-2-45}$$

Then,

$$\text{Re}\,(\rho_{km}) = \frac{1}{\sqrt{\mathcal{E}_m \mathcal{E}_k}}\int_{-\infty}^{\infty} s_m(t)s_k(t)\,dt \tag{4-2-46}$$

or, equivalently,

$$\text{Re}\,(\rho_{km}) = \frac{\mathbf{s}_m \cdot \mathbf{s}_k}{\|\mathbf{s}_m\|\,\|\mathbf{s}_k\|} = \frac{\mathbf{s}_m \cdot \mathbf{s}_k}{\sqrt{\mathcal{E}_m \mathcal{E}_k}} \tag{4-2-47}$$

The cross-correlation coefficients between pairs of signal waveforms or signal vectors comprise one set of parameters that characterize the similarity

of a set of signals. Another related parameter is the Euclidean distance $d_{km}^{(e)}$ between a pair of signals, defined as

$$\begin{aligned} d_{km}^{(e)} &= \|\mathbf{s}_m - \mathbf{s}_k\| \\ &= \left\{\int_{-\infty}^{\infty} [s_m(t) - s_k(t)]^2\, dt\right\}^{1/2} \\ &= \{\mathscr{E}_m + \mathscr{E}_k - 2\sqrt{\mathscr{E}_m \mathscr{E}_k}\,\mathrm{Re}\,(\rho_{km})\}^{1/2} \end{aligned} \tag{4-2-48}$$

When $\mathscr{E}_m = \mathscr{E}_k = \mathscr{E}$ for all m and k, this expression simplifies to

$$d_{km}^{(e)} = \{2\mathscr{E}[1 - \mathrm{Re}\,(\rho_{km})]\}^{1/2} \tag{4-2-49}$$

Thus, the Euclidean distance is an alternative measure of the similarity (or dissimilarity) of the set of signal waveforms or the corresponding signal vectors.

In the following section, we describe digitally modulated signals and make use of the signal space representation for such signals. We shall observe that digitally modulated signals, which are classified as linear, are conveniently expanded in terms of two orthonormal basis functions of the form

$$\begin{aligned} f_1(t) &= \sqrt{\frac{2}{T}}\cos 2\pi f_c t \\ f_2(t) &= -\sqrt{\frac{2}{T}}\sin 2\pi f_c t \end{aligned} \tag{4-2-50}$$

Hence, if $s_{lm}(t)$ is expressed as $s_{lm}(t) = x_l(t) + jy_l(t)$, it follows that $s_m(t)$ in (4-2-42) may be expressed as

$$s_m(t) = x_l(t)f_1(t) + y_l(t)f_2(t) \tag{4-2-51}$$

where $x_l(t)$ and $y_l(t)$ represent the signal modulations.

4-3 REPRESENTATION OF DIGITALLY MODULATED SIGNALS

In the transmission of digital information over a communications channel, the modulator is the interface device that maps the digital information into analog waveforms that match the characteristics of the channel. The mapping is generally performed by taking blocks of $k = \log_2 M$ binary digits at a time from the information sequence $\{a_n\}$ and selecting one of $M = 2^k$ deterministic, finite energy waveforms $\{s_m(t), m = 1, 2, \ldots, M\}$ for transmission over the channel.

When the mapping from the digital sequence $\{a_n\}$ to waveforms is performed under the constraint that a waveform transmitted in any time interval depends on one or more previously transmitted waveforms, the modulator is said to have *memory*. On the other hand, when the mapping

from the sequence $\{a_n\}$ to the waveforms $\{s_m(t)\}$ is performed without any constraint on previously transmitted waveforms, the modulator is called *memoryless.*

In addition to classifying the modulator as either memoryless or having memory, we may classify it as either *linear* or *nonlinear.* Linearity of a modulation method requires that the principle of superposition applies in the mapping of the digital sequence into successive waveforms. In nonlinear modulation, the superposition principle does not apply to signals transmitted in successive time intervals. We shall begin by describing memoryless modulation methods.

4-3-1 Memoryless Modulation Methods

As indicated above, the modulator in a digital communication system maps a sequence of binary digits into a set of corresponding signal waveforms. These waveforms may differ in either amplitude or in phase or in frequency, or some combination of two or more signal parameters. We consider each of these signal types separately, beginning with digital pulse amplitude modulation (PAM). In all cases, we assume that the sequence of binary digits at the input to the modulator occurs at a rate of R bits/s.

Pulse Amplitude Modulated (PAM) Signals In digital PAM, the signal waveforms may be represented as

$$\begin{aligned} s_m(t) &= \text{Re}\,[A_m g(t) e^{j2\pi f_c t}] \\ &= A_m g(t) \cos 2\pi f_c t, \quad m = 1, 2, \ldots, M, \quad 0 \leq t \leq T \end{aligned} \tag{4-3-1}$$

where $\{A_m, 1 \leq m \leq M\}$ denote the set of M possible amplitudes corresponding to $M = 2^k$ possible k-bit blocks or *symbols.* The signal amplitudes A_m take the discrete values (levels)

$$A_m = (2m - 1 - M)d, \quad m = 1, 2, \ldots, M \tag{4-3-2}$$

where $2d$ is the distance between adjacent signal amplitudes. The waveform $g(t)$ is a real-valued signal pulse whose shape influences the spectrum of the transmitted signal, as we shall observe later. The symbol rate for the PAM signal is R/k. This is the rate at which changes occur in the amplitude of the carrier to reflect the transmission of new information. The time interval $T_b = 1/R$ is called the *bit interval* and the time interval $T = k/R = kT_b$ is called the *symbol interval.*

The M PAM signals have energies

$$\begin{aligned} \mathscr{E}_m &= \int_0^T s_m^2(t)\, dt \\ &= \tfrac{1}{2} A_m^2 \int_0^T g^2(t)\, dt \\ &= \tfrac{1}{2} A_m^2 \mathscr{E}_g \end{aligned} \tag{4-3-3}$$

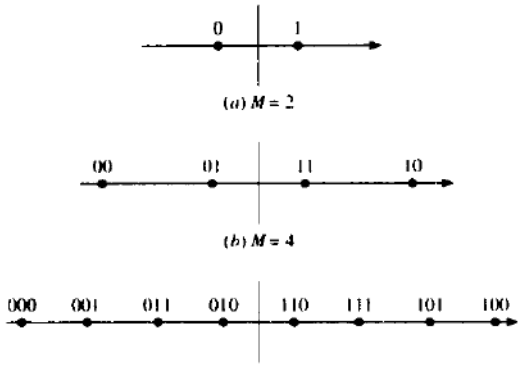


FIGURE 4-3-1 Signal space diagram for digital PAM signals.

where $\mathscr{E}_g$ denotes the energy in the pulse $g(t)$. Clearly, these signals are one-dimensional ($N = 1$), and, hence, are represented by the general form

$$s_m(t) = s_m f(t) \tag{4-3-4}$$

where $f(t)$ is defined as the unit-energy signal waveform given as

$$f(t) = \sqrt{\frac{2}{\mathscr{E}_g}}\, g(t) \cos 2\pi f_c t \tag{4-3-5}$$

and

$$s_m = A_m \sqrt{\tfrac{1}{2}\mathscr{E}_g}, \qquad m = 1, 2, \ldots, M \tag{4-3-6}$$

The corresponding signal space diagrams for $M = 2$, $M = 4$ and $M = 8$ are shown in Fig. 4-3-1. Digital PAM is also called *amplitude-shift keying* (ASK).

The mapping or assignment of k information bits to the $M = 2^k$ possible signal amplitudes may be done in a number of ways. The preferred assignment is one in which the adjacent signals amplitudes differ by one binary digit as illustrated in Fig. 4-3-1. This mapping is called *Gray encoding*. It is important in the demodulation of the signal because the most likely errors caused by noise involve the erroneous selection of an adjacent amplitude to the transmitted signal amplitude. In such a case, only a single bit error occurs in the k-bit sequence.

We note that the Euclidean distance between any pair of signal points is

$$\begin{aligned} d_{mn}^{(e)} &= \sqrt{(s_m - s_n)^2} \\ &= \sqrt{\tfrac{1}{2}\mathscr{E}_g}\,|A_m - A_n| \\ &= d\sqrt{2\mathscr{E}_g}\,|m - n| \end{aligned} \tag{4-3-7}$$

Hence, the distance between a pair of adjacent signal points, i.e., the minimum Euclidean distance, is

$$d_{\min}^{(e)} = d\sqrt{2\mathscr{E}_g} \tag{4-3-8}$$

The carrier-modulated PAM signal represented by (4-3-1) is a double-sideband (DSB) signal and requires twice the channel bandwidth of the equivalent lowpass signal for transmission. Alternatively, we may use single-sideband (SSB) PAM, which has the representation (lower or upper sideband).

$$s_m(t) = \text{Re}\{A_m[g(t) \pm j\hat{g}(t)]e^{j2\pi f_c t}\}, \quad m = 1, 2, \ldots, M \tag{4-3-9}$$

where $\hat{g}(t)$ is the Hilbert transform of $g(t)$. Thus, the bandwidth of the SSB signal is half that of the DSB signal.

The digital PAM signal is also appropriate for transmission over a channel that does not require carrier modulation. In this case, the signal waveform may be simply represented as

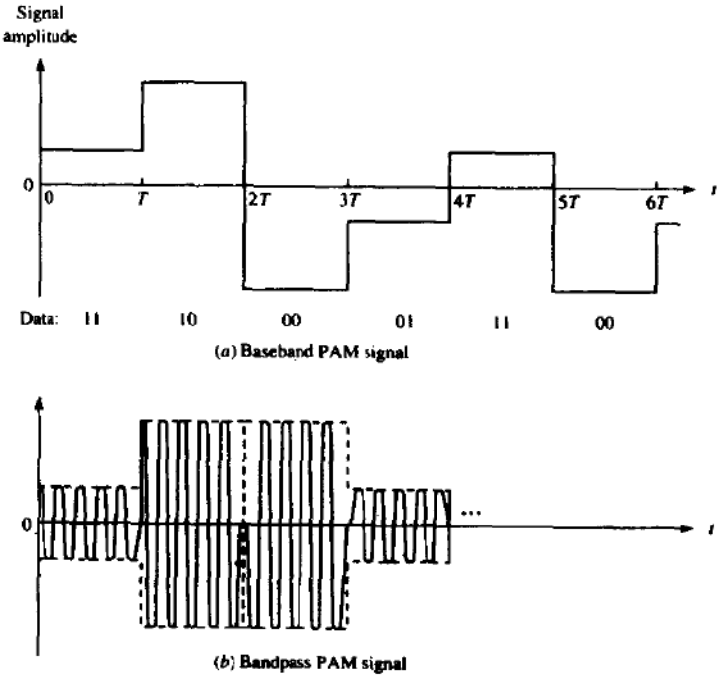
$$s_m(t) = A_m g(t), \quad m = 1, 2, \ldots, M \tag{4-3-10}$$

This is now called a *baseband* signal. For example a four-amplitude level baseband PAM signal is illustrated in Fig. 4-3-2(*a*). The carrier-modulated version of the signal is shown in Fig. 4-3-2(*b*).

In the special case of $M = 2$ signals, the binary PAM waveforms have the special property that

$$s_1(t) = -s_2(t)$$

FIGURE 4-3-2 Baseband and bandpass PAM signals.



(*a*) Baseband PAM signal

(*b*) Bandpass PAM signal

Hence, these two signals have the same energy and a cross-correlation coefficient of -1. Such signals are called *antipodal.*

Phase-Modulated Signals In digital phase modulation, the M signal waveforms are represented as

$$\begin{aligned} s_m(t) &= \text{Re}\left[g(t)e^{j2\pi(m-1)/M}e^{j2\pi f_c t}\right], \quad m = 1, 2, \ldots, M, \quad 0 \le t \le T \\ &= g(t)\cos\left[2\pi f_c t + \frac{2\pi}{M}(m-1)\right] \\ &= g(t)\cos\frac{2\pi}{M}(m-1)\cos 2\pi f_c t - g(t)\sin\frac{2\pi}{M}(m-1)\sin 2\pi f_c t \end{aligned} \tag{4-3-11}$$

where $g(t)$ is the signal pulse shape and $\theta_m = 2\pi(m-1)/M$, $m = 1, 2, \ldots, M$, are the M possible phases of the carrier that convey the transmitted information. Digital phase modulation is usually called *phase-shift keying* (PSK).

We note that these signal waveforms have equal energy, i.e.,

$$\begin{aligned} \mathscr{E} &= \int_0^T s_m^2(t)\,dt \\ &= \frac{1}{2}\int_0^T g^2(t)\,dt = \tfrac{1}{2}\mathscr{E}_g \end{aligned} \tag{4-3-12}$$

Furthermore, the signal waveforms may be represented as a linear combination of two-orthonormal signal waveforms, $f_1(t)$ and $f_2(t)$, i.e.,

$$s_m(t) = s_{m1}f_1(t) + s_{m2}f_2(t) \tag{4-3-13}$$

where

$$f_1(t) = \sqrt{\frac{2}{\mathscr{E}_g}}\,g(t)\cos 2\pi f_c t \tag{4-3-14}$$

$$f_2(t) = -\sqrt{\frac{2}{\mathscr{E}_g}}\,g(t)\sin 2\pi f_c t \tag{4-3-15}$$

and the two-dimensional vectors $\mathbf{s}_m = [s_{m1}\ \ s_{m2}]$ are given by

$$\mathbf{s}_m = \left[\sqrt{\frac{\mathscr{E}_g}{2}}\cos\frac{2\pi}{M}(m-1) \quad \sqrt{\frac{\mathscr{E}_g}{2}}\sin\frac{2\pi}{M}(m-1)\right], \quad m = 1, 2, \ldots, M \tag{4-3-16}$$

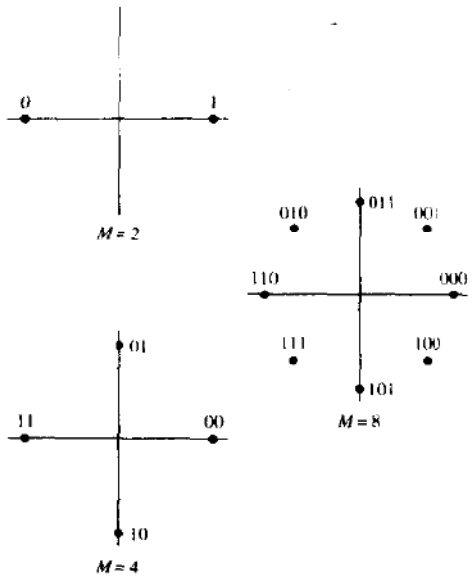


FIGURE 4-3-3 Signal space diagrams for PSK signals.

Signal space diagrams for $M = 2$, 4, and 8 are shown in Fig. 4-3-3. We note that $M = 2$ corresponds to one-dimensional signals, which are identical to binary PAM signals.

As is the case of PAM, the mapping or assignment of k information bits to the $M = 2^k$ possible phases may be done in a number of ways. The preferred assignment is Gray encoding, so that the most likely errors caused by noise will result in a single bit error in the k-bit symbol.

The Euclidean distance between signal points is

$$d_{mn}^{(c)} = |\mathbf{s}_m - \mathbf{s}_n| = \left\{ \mathscr{E}_g \left[1 - \cos \frac{2\pi}{M} (m - n) \right] \right\}^{1/2} \tag{4-3-17}$$

The minimum Euclidean distance corresponds to the case in which $|m - n| = 1$, i.e., adjacent signal phases. In this case,

$$d_{\min}^{(c)} = \sqrt{\mathscr{E}_g \left(1 - \cos \frac{2\pi}{M} \right)} \tag{4-3-18}$$

Quadrature Amplitude Modulation The bandwidth efficiency of PAM/SSB can also be obtained by simultaneously impressing two separate k-bit symbols from the information sequence $\{a_n\}$ on two quadrature carriers

$\cos 2\pi f_c t$ and $\sin 2\pi f_c t$. The resulting modulation technique is called quadrature PAM or QAM, and the corresponding signal waveforms may be expressed as

$$\begin{aligned} s_m(t) &= \text{Re}\,[(A_{mc} + jA_{ms})g(t)e^{j2\pi f_c t}], \quad m = 1, 2, \ldots, M, \quad 0 \le t \le T \\ &= A_{mc}g(t)\cos 2\pi f_c t - A_{ms}g(t)\sin 2\pi f_c t \end{aligned} \tag{4-3-19}$$

where A_{mc} and A_{ms} are the information-bearing signal amplitudes of the quadrature carriers and $g(t)$ is the signal pulse.

Alternatively, the QAM signal waveforms may be expressed as

$$\begin{aligned} s_m(t) &= \text{Re}\,[V_m e^{j\theta_m} g(t) e^{j2\pi f_c t}] \\ &= V_m g(t)\cos(2\pi f_c t + \theta_m) \end{aligned} \tag{4-3-20}$$

where $V_m = \sqrt{A_{mc}^2 + A_{ms}^2}$ and $\theta_m = \tan^{-1}(A_{ms}/A_{mc})$. From this expression, it is apparent that the QAM signal waveforms may be viewed as combined amplitude and phase modulation.

In fact, we may select any combination of M_1-level PAM and M_2-phase PSK to construct an $M = M_1 M_2$ combined PAM–PSK signal constellation. If $M_1 = 2^n$ and $M_2 = 2^m$, the combined PAM–PSK signal constellation results in the simultaneous transmission of $m + n = \log M_1 M_2$ binary digits occurring at a symbol rate $R/(m+n)$. Examples of signal space diagrams for combined PAM–PSK are shown in Fig. 4-3-4, for $M = 8$ and $M = 16$.

As in the case of PSK signals, the QAM signal waveforms may be represented as a linear combination of two orthonormal signal waveforms, $f_1(t)$ and $f_2(t)$, i.e.,

$$s_m(t) = s_{m1} f_1(t) + s_{m2} f_2(t) \tag{4-3-21}$$

where

$$\begin{aligned} f_1(t) &= \sqrt{\frac{2}{\mathscr{E}_g}}\, g(t)\cos 2\pi f_c t \\ f_2(t) &= -\sqrt{\frac{2}{\mathscr{E}_g}}\, g(t)\sin 2\pi f_c t \end{aligned} \tag{4-3-22}$$

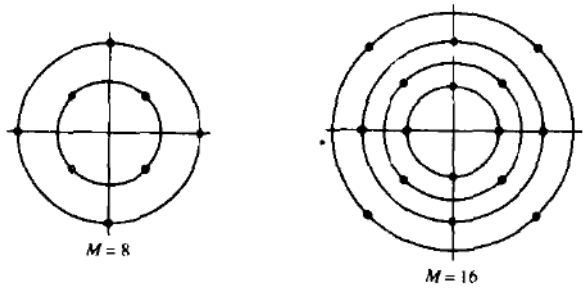


FIGURE 4-3-4 Examples of combined PAM–PSK signal space diagrams.

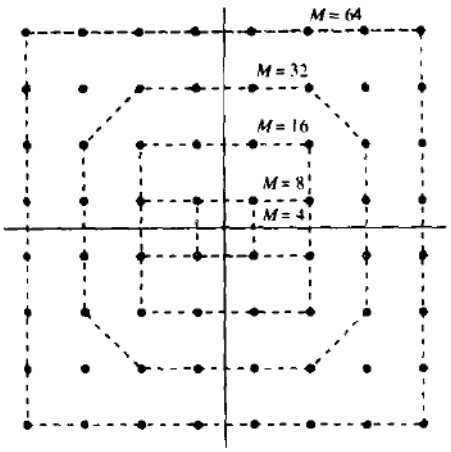


FIGURE 4-3-5 Several signal space diagrams for rectangular QAM.

and

$$\begin{aligned}\mathbf{s}_m &= [s_{m1} \quad s_{m2}] \\ &= [A_{mc}\sqrt{\tfrac{1}{2}\mathscr{E}_g} \quad A_{ms}\sqrt{\tfrac{1}{2}\mathscr{E}_g}]\end{aligned} \tag{4-3-23}$$

$\mathscr{E}_g$ is the energy of the signal pulse $g(t)$.

The Euclidean distance between any pair of signal vectors is

$$\begin{aligned}d_{mn}^{(e)} &= |\mathbf{s}_m - \mathbf{s}_n| \\ &= \sqrt{\tfrac{1}{2}\mathscr{E}_g[(A_{mc} - A_{nc})^2 + (A_{ms} - A_{ns})^2]}\end{aligned} \tag{4-3-24}$$

In the special case where the signal amplitudes takes the set of discrete values $\{(2m - 1 - M)d, \ m = 1, 2, \ldots, M\}$, the signal space diagram is rectangular, as shown in Fig. 4-3-5. In this case, the Euclidean distance between adjacent points, i.e., the minimum distance, is

$$d_{\min}^{(e)} = d\sqrt{2\mathscr{E}_g} \tag{4-3-25}$$

which is the same result as for PAM.

Multidimensional Signals It is apparent from the discussion above that the digital modulation of the carrier amplitude and phase allows us to construct signal waveforms that correspond to two-dimensional vectors and signal space diagrams. If we wish to construct signal waveforms corresponding to higher-dimensional vectors, we may use either the time domain or the frequency domain or both in order to increase the number of dimensions.

Suppose we have N-dimensional signal vectors. For any N, we may subdivide a time interval of length $T_1 = NT$ into N subintervals of length $T = T_1/N$. In each subinterval of length T, we may use binary PAM (a one-dimensional signal) to transmit an element of the N-dimensional signal

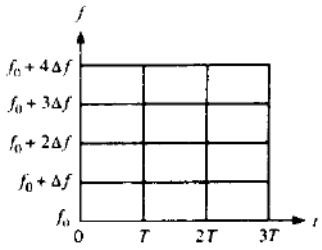


FIGURE 4-3-6 Subdivision of time and frequency axes into distinct slots.

vector. Thus, the N time slots are used to transmit the N-dimensional signal vector. If N is even, a time slot of length T may be used to simultaneously transmit two components of the N-dimensional vector by modulating the amplitude of quadrature carriers independently by the corresponding components. In this manner, the N-dimensional signal vector is transmitted in $\frac{1}{2}NT$ seconds ($\frac{1}{2}N$ time slots).

Alternatively, a frequency band of width $N\,\Delta f$ may be subdivided into N frequency slots each of width Δf. An N-dimensional signal vector can be transmitted over the channel by simultaneously modulating the amplitude of N carriers, one in each of the N frequency slots. Care must be taken to provide sufficient frequency separation Δf between successive carriers so that there is no cross talk interference among the signals on the N carriers. If quadrature carriers are used in each frequency slot, the N-dimensional vector (even N) may be transmitted in $\frac{1}{2}N$ frequency slots, thus reducing the channel bandwidth utilization by a factor of 2.

More generally, we may use both the time and frequency domains jointly to transmit an N-dimensional signal vector. For example, Fig. 4-3-6 illustrates a subdivision of the time and frequency axes into 12 slots. Thus, an $N = 12$-dimensional signal vector may be transmitted by PAM or an $N = 24$-dimensional signal vector may be transmitted by use of two quadrature carriers (QAM) in each slot.

Orthogonal Multidimensional Signals As a special case of the construction of multidimensional signals, let us consider the construction of M equal-energy orthogonal signal waveforms that differ in frequency, and are represented as

$$\begin{aligned} s_m(t) &= \text{Re}\,[s_{lm}(t)e^{j2\pi f_c t}], \quad m = 1, 2, \ldots, M, \quad 0 \leq t \leq T \\ &= \sqrt{\frac{2\mathcal{E}}{T}} \cos[2\pi f_c t + 2\pi m\,\Delta f\, t] \end{aligned} \tag{4-3-26}$$

where the equivalent lowpass signal waveforms are defined as

$$s_{lm}(t) = \sqrt{\frac{2\mathcal{E}}{T}}\, e^{j2\pi m\,\Delta f t}, \quad m = 1, 2, \ldots, M, \quad 0 \leq t \leq T \tag{4-3-27}$$

This type of frequency modulation is called *frequency-shift keying* (FSK).

These waveforms are characterized as having equal energy and cross-correlation coefficients

$$\rho_{km} = \frac{2\mathscr{E}/T}{2\mathscr{E}} \int_0^T e^{j2\pi(m-k)\Delta f t}\, dt$$

$$= \frac{\sin \pi T(m-k)\,\Delta f}{\pi T(m-k)\,\Delta f} e^{j\pi T(m-k)\Delta f} \tag{4-3-28}$$

The real part of ρ_{km} is

$$\rho_r \equiv \text{Re}\,(\rho_{km}) = \frac{\sin\,[\pi T(m-k)\,\Delta f]}{\pi T(m-k)\,\Delta f} \cos\,[\pi T(m-k)\,\Delta f]$$

$$= \frac{\sin\,[2\pi T(m-k)\,\Delta f]}{2\pi T(m-k)\,\Delta f} \tag{4-3-29}$$

First, we observe that $\text{Re}\,(\rho_{km}) = 0$ when $\Delta f = 1/2T$ and $m \neq k$. Since $|m-k| = 1$ corresponds to adjacent frequency slots, $\Delta f = 1/2T$ represents the minimum frequency separation between adjacent signals for orthogonality of the M signals. Plots of $\text{Re}\,(\rho_{km})$ versus Δf and $|\rho_{km}|$ versus Δf are shown in Fig. 4-3-7. Note that $|\rho_{km}| = 0$ for multiples of $1/T$ whereas $\text{Re}\,(\rho_{km}) = 0$ for multiples of $1/2T$.

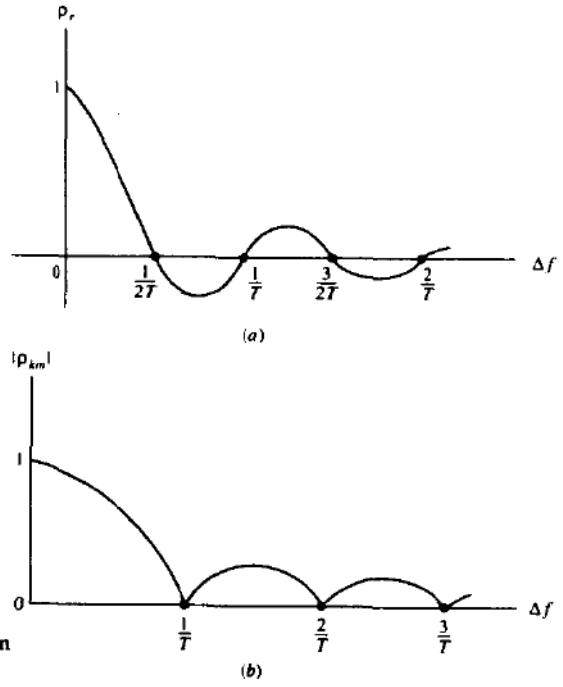


FIGURE 4-3-7 Cross-correlation coefficient as a function of frequency separation for FSK signals.

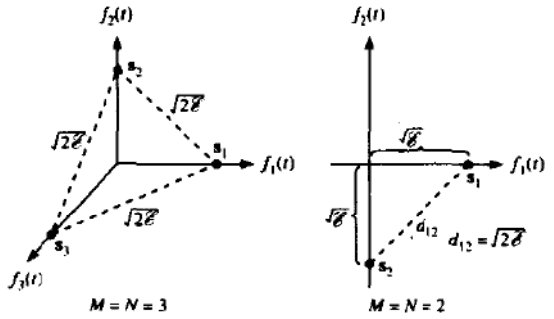


FIGURE 4-3-8 Orthogonal signals for $M = N = 3$ and $M = N = 2$.

For the case in which $\Delta f = 1/2T$, the M FSK signals are equivalent to the N-dimensional vectors

$$\begin{aligned} \mathbf{s}_1 &= [\sqrt{\mathscr{E}} \quad 0 \quad 0 \quad \ldots \quad 0 \quad 0] \\ \mathbf{s}_2 &= [0 \quad \sqrt{\mathscr{E}} \quad 0 \quad \ldots \quad 0 \quad 0] \\ &\vdots \quad \vdots \\ \mathbf{s}_N &= [0 \quad 0 \quad 0 \quad \ldots \quad 0, \quad \sqrt{\mathscr{E}}] \end{aligned} \tag{4-3-30}$$

where $N = M$. The distance between pairs of signals is

$$d_{km}^{(e)} = \sqrt{2\mathscr{E}} \quad \text{for all } m, k \tag{4-3-31}$$

which is also the minimum distance. Figure 4-3-8 illustrates the signal space diagram for $M = N = 2$ and $M = N = 3$.

Biorthogonal Signals A set of M biorthogonal signals can be constructed from $\frac{1}{2}M$ orthogonal signals by simply including the negatives of the orthogonal signals. Thus, we require $N = \frac{1}{2}M$ dimensions for the construction of a set of M biorthogonal signals. Figure 4-3-9 illustrates the biorthogonal signals for $M = 4$ and 6.

We note that the correlation between any pair of waveforms is either $\rho_r = -1$ or 0. The corresponding distances are $d = 2\sqrt{\mathscr{E}}$ or $\sqrt{2\mathscr{E}}$, with the latter being the minimum distance.

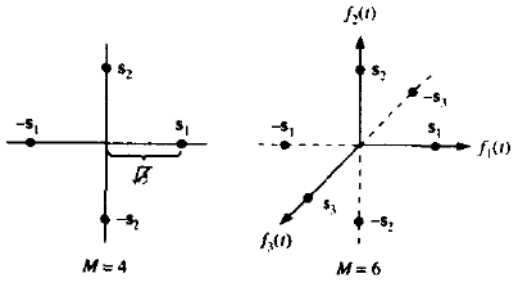


FIGURE 4-3-9 Signal space diagrams for $M = 4$ and $M = 6$ biorthogonal signals.

Simplex Signals Suppose we have a set of M orthogonal waveforms $\{s_m(t)\}$ or, equivalently, their vector representation $\{\mathbf{s}_m\}$. Their mean is

$$\bar{\mathbf{s}} = \frac{1}{M}\sum_{m=1}^{M} \mathbf{s}_m \tag{4-3-32}$$

Now, let us construct another set of M signals by subtracting the mean from each of the M orthogonal signals. Thus,

$$\mathbf{s}'_m = \mathbf{s}_m - \bar{\mathbf{s}}, \quad m = 1, 2, \ldots, M \tag{4-3-33}$$

The effect of the subtraction is to translate the origin of the m orthogonal signals to the point $\bar{\mathbf{s}}$.

The resulting signal waveforms are called *simplex signals* and have the following properties. First, the energy per waveform is

$$\begin{aligned} |\mathbf{s}'_m|^2 &= |\mathbf{s}_m - \bar{\mathbf{s}}|^2 \\ &= \mathscr{E} - \frac{2}{M}\mathscr{E} + \frac{1}{M}\mathscr{E} \\ &= \mathscr{E}\left(1 - \frac{1}{M}\right) \end{aligned} \tag{4-3-34}$$

Second, the cross-correlation of any pair of signals is

$$\begin{aligned} \mathrm{Re}\,(\rho_{mn}) &= \frac{\mathbf{s}'_m \cdot \mathbf{s}'_n}{|\mathbf{s}'_m|\,|\mathbf{s}'_n|} \\ &= \frac{-1/M}{1 - 1/M} = -\frac{1}{M-1} \end{aligned} \tag{4-3-35}$$

for all m, n. Hence, the set of simplex waveforms is *equally correlated* and requires less energy, by the factor $1 - 1/M$, than the set of orthogonal waveforms. Since only the origin was translated, the distance between any pair of signal points is maintained at $d = \sqrt{2\mathscr{E}}$, which is the same as the distance between any pair of orthogonal signals.

Figure 4-3-10 illustrates the simplex signals for $M = 2$, 3, and 4. Note that the signal dimensionality is $N = M - 1$.

Signal Waveforms from Binary Codes A set of M signaling waveforms can be generated from a set of M binary code words of the form

$$\mathbf{C}_m = [c_{m1} \quad c_{m2} \quad \ldots \quad c_{mN}], \quad m = 1, 2, \ldots, M \tag{4-3-36}$$

where $c_{mj} = 0$ or 1 for all m and j. Each component of a code word is mapped into an elementary binary PSK waveform as follows:

$$\begin{aligned} c_{mj} = 1 &\Rightarrow s_{mj}(t) = \sqrt{\frac{2\mathscr{E}_c}{T_c}}\cos 2\pi f_c t \quad (0 \le t \le T_c) \\ c_{mj} = 0 &\Rightarrow s_{mj}(t) = -\sqrt{\frac{2\mathscr{E}_c}{T_c}}\cos 2\pi f_c t \quad (0 \le t \le T_c) \end{aligned} \tag{4-3-37}$$

where $T_c = T/N$ and $\mathscr{E}_c = \mathscr{E}/N$. Thus, the M code words $\{C_m\}$ are mapped into a set of M waveforms $\{s_m(t)\}$.

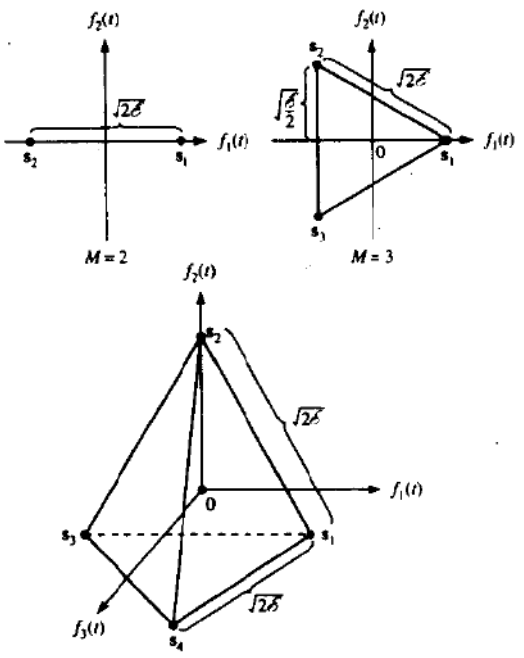


FIGURE 4-3-10 Signal space diagrams for M-ary simplex signals.

The waveforms can be represented in vector form as

$$\mathbf{s}_m = [s_{m1} \quad s_{m2} \quad \cdots \quad s_{mN}], \qquad m = 1, 2, \ldots, M \tag{4-3-38}$$

where $s_{mj} = \pm\sqrt{\mathcal{E}/N}$ for all m and j. N is called the block length of the code, and it is also the dimension of the M waveforms.

We note that there are 2^N possible waveforms that can be constructed from the 2^N possible binary code words. We may select a subset of $M < 2^N$ signal waveforms for transmission of the information. We also observe that the 2^N possible signal points correspond to the vertices of an N-dimensional hypercube with its center at the origin. Figure 4-3-11 illustrates the signal points in $N = 2$ and 3 dimensions.

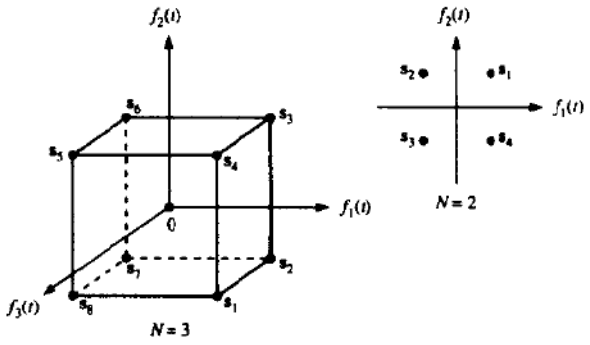


FIGURE 4-3-11 Signal space diagrams for signals generated from binary codes.

Each of the M waveforms has energy $\mathscr{E}$. The cross-correlation between any pair of waveforms depends on how we select the M waveforms from the 2^N possible waveforms. This topic is treated in Chapter 7. Clearly, any adjacent signal points have a cross-correlation coefficient

$$\rho_r = \frac{\mathscr{E}(1 - 2/N)}{\mathscr{E}} = \frac{N-2}{N} \tag{4-3-39}$$

and a corresponding distance of

$$\begin{aligned} d^{(e)} &= \sqrt{2\mathscr{E}(1-\rho_r)} \\ &= \sqrt{4\mathscr{E}/N} \end{aligned} \tag{4-3-40}$$

This concludes our discussion of memoryless modulation signals.

4-3-2 Linear Modulation with Memory

The modulation signals introduced in the previous section were classified as memoryless, because there was no dependence between signals transmitted in non-overlapping symbol intervals. In this section, we present some modulation signals in which there is dependence between the signals transmitted in successive symbol intervals. This signal dependence is usually introduced for the purpose of shaping the spectrum of the transmitted signal so that it matches the spectral characteristics of the channel. Signal dependence between signals transmitted in different signal intervals is generally accomplished by encoding the data sequence at the input to the modulator by means of a *modulation code*, as described in Chapter 9.

In this section, we shall present examples of modulation signals with memory and characterize their memory in terms of Markov chains. We shall confine our treatment to baseband signals. The generalization to bandpass signals is relatively straightforward.

Figure 4-3-12 illustrates three different baseband signals and the corresponding data sequence. The first signal, called NRZ, is the simplest. The binary information digit 1 is represented by a rectangular pulse of polarity A and the binary digit zero is represented by a rectangular pulse of polarity $-A$.

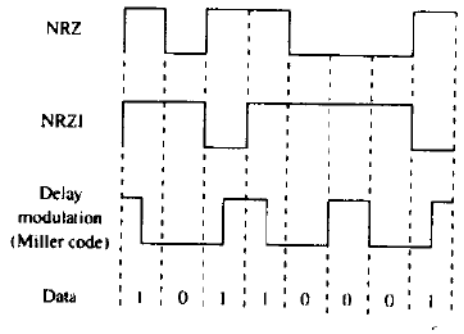


FIGURE 4-3-12 Baseband signals.

Hence, the NRZ modulation is memoryless and is equivalent to a binary PAM or a binary PSK signal in a carrier-modulated system.

The NRZI signal is different from the NRZ signal in that transitions from one amplitude level to another occur only when a 1 is transmitted. The amplitude level remains unchanged when a zero is transmitted. This type of signal encoding is called *differential encoding*. The encoding operation is described mathematically by the relation

$$b_k = a_k \oplus b_{k-1} \tag{4-3-41}$$

where $\{a_k\}$ is the binary information sequence into the encoder, $\{b_k\}$ is the output sequence of the encoder, and $\oplus$ denotes addition modulo 2. When $b_k = 1$, the transmitted waveform is a rectangular pulse of amplitude A, and when $b_k = 0$, the transmitted waveform is a rectangular pulse of amplitude $-A$. Hence, the output of the encoder is mapped into one of two waveforms in exactly the same manner as for the NRZ signal.

The differential encoding operation introduces memory in the signal. The combination of the encoder and the modulator operations may be represented by a state diagram (a Markov chain) as shown in Fig. 4-3-13. The state diagram may be described by two transition matrices corresponding to the two possible input bits {0, 1}. We note that when $a_k = 0$, the encoder stays in the same state. Hence, the state transition matrix for a zero is simply

$$\mathbf{T}_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{4-3-42}$$

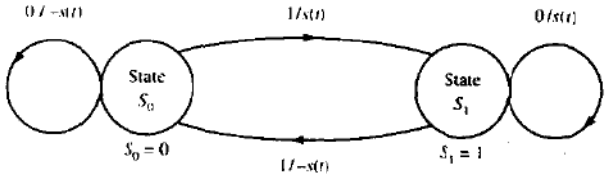
where $t_{ij} = 1$ if a_k results in a transition from state i to state j, $i = 1, 2$, and $j = 1, 2$; otherwise, $t_{ij} = 0$. Similarly, the state transition matrix for $a_k = 1$ is

$$\mathbf{T}_2 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \tag{4-3-43}$$

Thus, these two state transition matrices characterize the NRZI signal.

Another way to display the memory introduced by the precoding operation is by means of a trellis diagram. The trellis diagram for the NRZI signal is

FIGURE 4-3-13 *State diagram for the NRZI signal.*



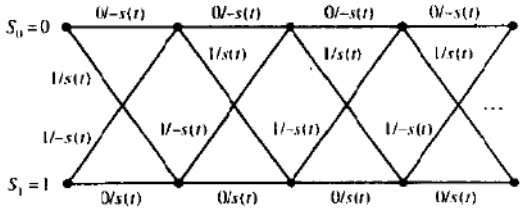


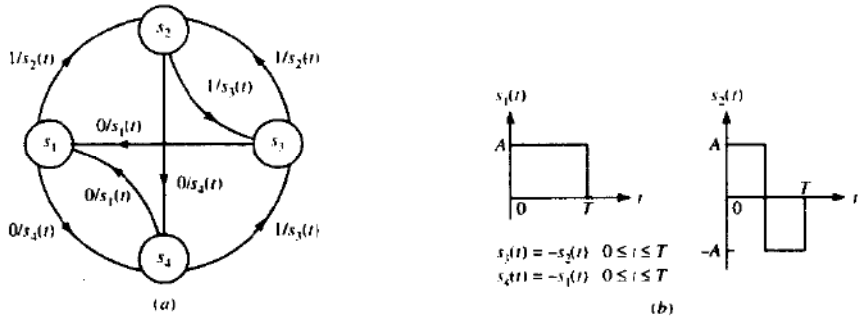
FIGURE 4-3-14 The trellis diagram for the NRZI signal.

illustrated in Fig. 4-3-14. The trellis provides exactly the same information concerning the signal dependence as the state diagram, but also depicts a time evolution of the state transitions.

The signal generated by delay modulation also has memory. As shown in Chapter 9, delay modulation is equivalent to encoding the data sequence by a run-length-limited code called a *Miller code* and using NRZI to transmit the encoded data. This type of digital modulation has been used extensively for digital magnetic recording and in carrier modulation systems employing binary PSK. The signal may be described by a state diagram that has four states as shown in Fig. 4-3-15(*a*). There are two elementary waveforms $s_1(t)$ and $s_2(t)$ and their negatives $-s_1(t)$ and $-s_2(t)$, which are used for transmitting the binary information. These waveforms are illustrated in Fig. 4-3-15(*b*). The mapping from bits to corresponding waveforms is illustrated in the state diagram. The state transition matrices that characterize the memory of this encoding and modulation method are easily obtained from the state diagram in Fig. 4-3-15. When $a_k = 0$, we have

$$\mathbf{T}_1 = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \tag{4-3-44}$$

FIGURE 4-3-15 State diagram (*a*) and basic waveforms (*b*) for delay modulated (Miller-encoded) signal.



and when $a_k = 1$, the transition matrix is

$$\mathbf{T}_2 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{4-3-45}$$

Thus, these two 4×4 state transition matrices characterize the state diagram for the Miller-encoded signal.

Modulation techniques with memory such as NRZI and Miller coding are generally characterized by a K-state Markov chain with *stationary state probabilities* $\{p_i, i = 1, 2, \ldots, K\}$ and *transition probabilities* $\{p_{ij}, i, j = 1, 2, \ldots, K\}$. Associated with each transition is a signal waveform $s_j(t)$, $j = 1, 2, \ldots, K$. Thus, the transition probability p_{ij} denotes the probability that signal waveform $s_j(t)$ is transmitted in a given signaling interval after the transmission of the signal waveform $s_i(t)$ in the previous signaling interval. The transition probabilities may be arranged in matrix form as

$$\mathbf{P} = \begin{bmatrix} p_{11} & p_{12} & \cdots & p_{1K} \\ p_{21} & p_{22} & \cdots & p_{2K} \\ \vdots & \vdots & & \vdots \\ p_{K1} & p_{K2} & \cdots & p_{KK} \end{bmatrix} \tag{4-3-46}$$

where $\mathbf{P}$ is called the *transition probability matrix.*

The transition probability matrix is easily obtained from the transition matrices $\{\mathbf{T}_i\}$ and the corresponding probabilities of occurrence of the input bits (or, equivalently, the stationary state transition probabilities $\{p_i\}$). The general relationship may be expressed as

$$\mathbf{P} = \sum_{i=1}^{2} q_i \mathbf{T}_i \tag{4-3-47}$$

where $q_1 = P(a_k = 0)$ and $q_2 = P(a_k = 1)$.

For the NRZI signal with equal state probabilities $p_1 = p_2 = \frac{1}{2}$ and transition matrices given by (4-3-42) and (4-3-43), the transition probability matrix is

$$\mathbf{P} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \end{bmatrix} \tag{4-3-48}$$

Similarly, the transition probability matrix for the Miller-coded signal with equally likely symbols ($q_1 = q_2 = \frac{1}{2}$ or, equivalently, $p_1 = p_2 = p_3 = p_4 = \frac{1}{4}$) is

$$\mathbf{P} = \begin{bmatrix} 0 & \frac{1}{2} & 0 & \frac{1}{2} \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 \end{bmatrix} \tag{4-3-49}$$

The transition probability matrix is useful in the determination of the spectral

characteristics of digital modulation techniques with memory, as we shall observe in Section 4-4.

4-3-3 Nonlinear Modulation Methods with Memory

In this section, we consider a class of digital modulation methods in which the phase of the signal is constrained to be continuous. This constraint results in a phase or frequency modulator that has memory. The modulation method is also nonlinear.

Continuous-Phase FSK (CPFSK) A conventional FSK signal is generated by shifting the carrier by an amount $f_n = \frac{1}{2}\Delta f I_n$, $I_n = \pm 1, \pm 3, \ldots, \pm(M-1)$, to reflect the digital information that is being transmitted. This type of FSK signal was described in Section 4-3-1, and it is memoryless. The switching from one frequency to another may be accomplished by having $M = 2^k$ separate oscillators tuned to the desired frequencies and selecting one of the M frequencies according to the particular k-bit symbol that is to be transmitted in a signal interval of duration $T = k/R$ seconds. However, such abrupt switching from one oscillator output to another in successive signaling intervals results in relatively large spectral side lobes outside of the main spectral band of the signal and, consequently, this method requires a large frequency band for transmission of the signal.

To avoid the use of signals having large spectral side lobes, the information-bearing signal frequency modulates a single carrier whose frequency is changed continuously. The resulting frequency-modulated signal is phase-continuous and, hence, it is called *continuous-phase* FSK (CPFSK). This type of FSK signal has memory because the phase of the carrier is constrained to be continuous.

In order to represent a CPFSK signal, we begin with a PAM signal

$$d(t) = \sum_n I_n g(t - nT) \tag{4-3-50}$$

where $\{I_n\}$ denotes the sequence of amplitudes obtained by mapping k-bit blocks of binary digits from the information sequence $\{a_n\}$ into the amplitude levels $\pm 1, \pm 3, \ldots, \pm(M-1)$ and $g(t)$ is a rectangular pulse of amplitude $1/2T$ and duration T seconds. The signal $d(t)$ is used to frequency-modulate the carrier. Consequently, the equivalent lowpass waveform $v(t)$ is expressed as

$$v(t) = \sqrt{\frac{2\mathscr{E}}{T}} \exp\left\{j\left[4\pi T f_d \int_{-\infty}^{t} d(\tau)\, d\tau + \phi_0\right]\right\} \tag{4-3-51}$$

where f_d is the *peak frequency deviation* and ϕ_0 is the initial phase of the carrier.

The carrier-modulated signal corresponding to (4-3-51) may be expressed as

$$s(t) = \sqrt{\frac{2\mathscr{E}}{T}} \cos\left[2\pi f_c t + \phi(t; \mathbf{I}) + \phi_0\right] \tag{4-3-52}$$

where $\phi(t; \mathbf{I})$ represents the time-varying phase of the carrier, which is defined as

$$\phi(t; \mathbf{I}) = 4\pi T f_d \int_{-\infty}^{t} d(\tau)\, d\tau$$
$$= 4\pi T f_d \int_{-\infty}^{t} \left[\sum_n I_n g(\tau - nT) \right] d\tau \tag{4-3-53}$$

Note that, although $d(t)$ contains discontinuities, the integral of $d(t)$ is continuous. Hence, we have a continuous-phase signal. The phase of the carrier in the interval $nT \leq t \leq (n+1)T$ is determined by integrating (4-3-53). Thus,

$$\phi(t; \mathbf{I}) = 2\pi f_d T \sum_{k=-\infty}^{n-1} I_k + 2\pi f_d (t - nT) I_n$$
$$= \theta_n + 2\pi h I_n q(t - nT) \tag{4-3-54}$$

where h, θ_n, and $q(t)$ are defined as

$$h = 2f_d T \tag{4-3-55}$$

$$\theta_n = \pi h \sum_{k=-\infty}^{n-1} I_k \tag{4-3-56}$$

$$q(t) = \begin{cases} 0 & (t < 0) \\ t/2T & (0 \leq t \leq T) \\ \frac{1}{2} & (t > T) \end{cases} \tag{4-3-57}$$

We observe that θ_n represents the accumulation (memory) of all symbols up to time $(n-1)T$. The parameter h is called the *modulation index*.

Continuous-Phase Modulation (CPM) When expressed in the form of (4-3-54), CPFSK becomes a special case of a general class of continuous-phase modulated (CPM) signals in which the carrier phase is

$$\phi(t; \mathbf{I}) = 2\pi \sum_{k=-\infty}^{n} I_k h_k q(t - kT), \quad nT \leq t \leq (n+1)T \tag{4-3-58}$$

where $\{I_k\}$ is the sequence of M-ary information symbols selected from the alphabet $\pm 1, \pm 3, \ldots, \pm(M-1)$, $\{h_k\}$ is a sequence of modulation indices, and $q(t)$ is some normalized waveform shape.

When $h_k = h$ for all k, the modulation index is fixed for all symbols. When the modulation index varies from one symbol to another, the CPM signal is called *multi-h*. In such a case, the $\{h_k\}$ are made to vary in a cyclic manner through a set of indices.

The waveform $q(t)$ may be represented in general as the integral of some pulse $g(t)$, i.e.,

$$q(t) = \int_0^t g(\tau)\, d\tau \tag{4-3-59}$$

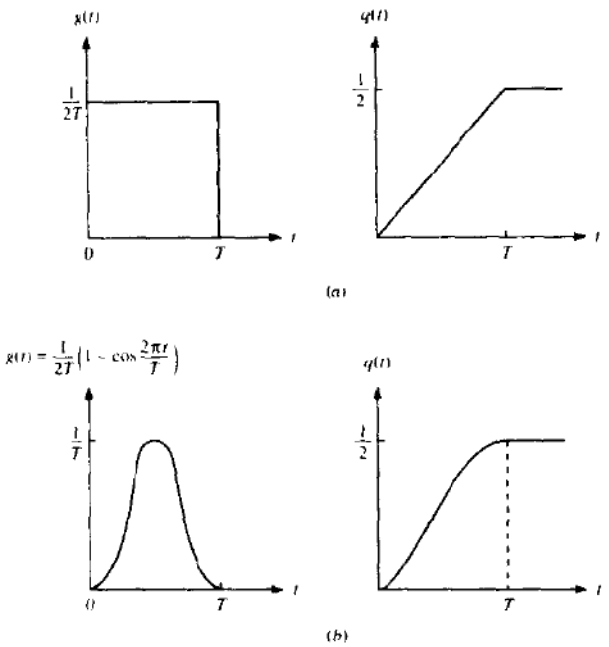


FIGURE 4-3-16 Pulse shapes for full response CPM (a, b) and partial response CPM (c, d).

If $g(t) = 0$ for $t > T$, the CPM signal is called *full response CPM*. If $g(t) \neq 0$ for $t > T$, the modulated signal is called *partial response CPM*. Figure 4-3-16 illustrates several pulse shapes for $g(t)$, and the corresponding $q(t)$. It is apparent that an infinite variety of CPM signals can be generated by choosing different pulse shapes $g(t)$ and by varying the modulation index h and the alphabet size M.

It is instructive to sketch the set of phase trajectories $\phi(t; \mathbf{I})$ generated by all possible values of the information sequence $\{I_n\}$. For example, in the case of CPFSK with binary symbols $I_n = \pm 1$, the set of phase trajectories beginning at time $t = 0$ is shown in Fig. 4-3-17. For comparison, the phase trajectories for quaternary CPFSK are illustrated in Fig. 4-3-18. These phase diagrams are called *phase trees*. We observe that the phase trees for CPFSK are piecewise linear as a consequence of the fact that the pulse $g(t)$ is rectangular. Smoother phase trajectories and phase trees are obtained by using pulses that do not contain discontinuities, such as the class of raised cosine pulses. For example, a phase trajectory generated by the sequence $(1, -1, -1, -1, 1, 1, -1, 1)$ for a partial response, raised cosine pulse of length $3T$ is illustrated in Fig. 4-3-19. For comparison, the corresponding phase trajectory generated by CPFSK is also shown.

The phase trees shown in these figures grow with time. However, the phase

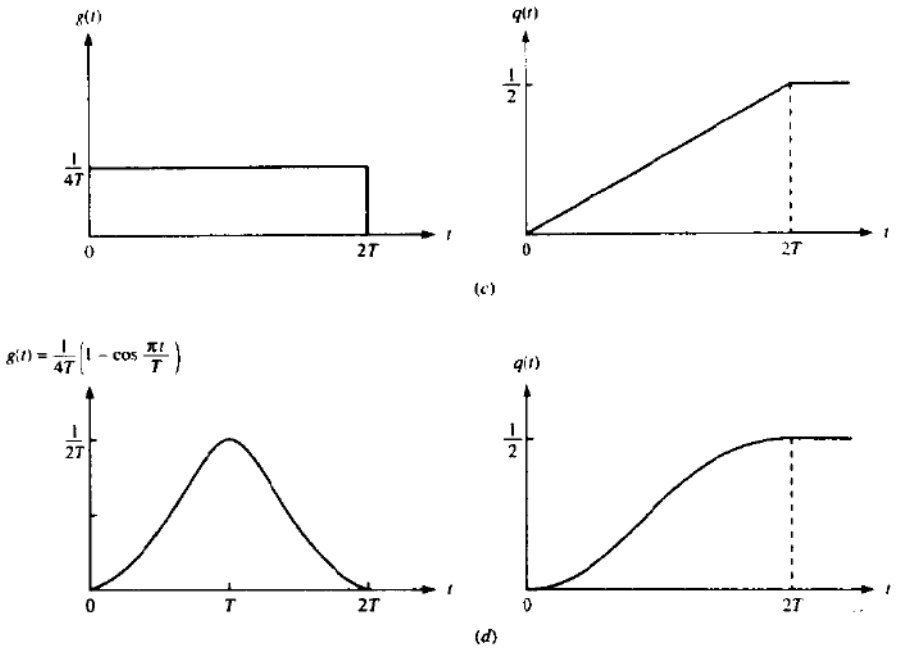


FIGURE 4-3-16 (*Continued*).

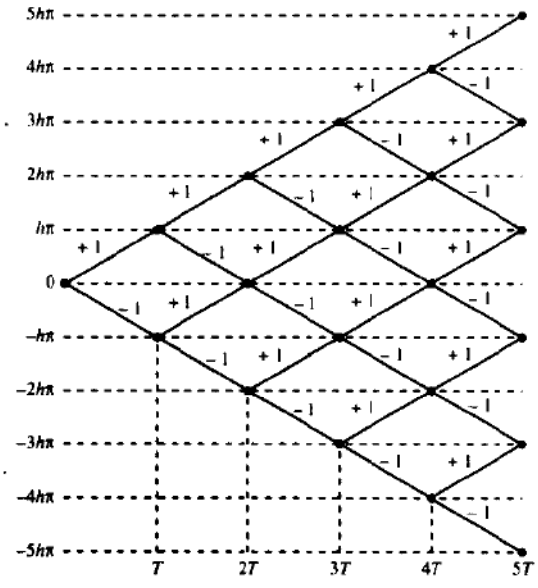


FIGURE 4-3-17 Phase trajectory for binary CPFSK.

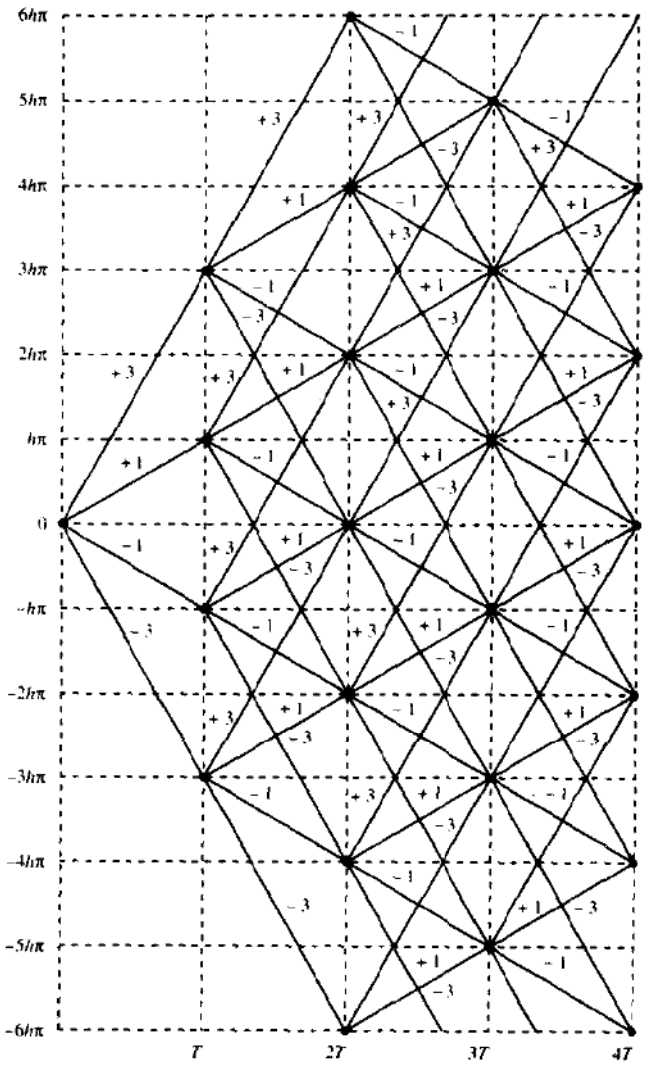


FIGURE 4-3-18 Phase trajectory for quaternary CPFSK.

of the carrier is unique only in the range from $\phi = 0$ to $\phi = 2\pi$ or, equivalently, from $\phi = -\pi$ to $\phi = \pi$. When the phase trajectories are plotted modulo 2π, say in the range $(-\pi, \pi)$, the phase tree collapses into a structure called a *phase trellis*. To properly view the phase trellis diagram, we may plot the two quadrature components $x_c(t; \mathbf{I}) = \cos \phi(t; \mathbf{I})$ and $x_s(t; \mathbf{I}) = \sin \phi(t; \mathbf{I})$ as functions of time. Thus, we generate a three-dimensional plot in which the quadrature components x_c and x_s appear on the surface of a cylinder of unit radius. For example, Fig. 4-3-20 illustrates the phase trellis or phase cylinder

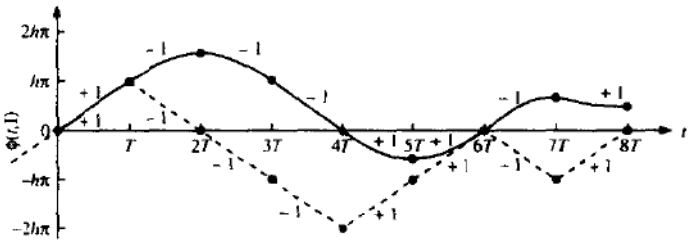


FIGURE 4-3-19 Phase trajectories for binary CPFSK (dashed) and binary, partial response CPM based on raised cosine pulse of length $3T$ (solid). [*From Sundberg (1986), © 1986 IEEE.*]

obtained with binary modulation, a modulation index $h = \frac{1}{2}$, and a raised cosine pulse of length $3T$.

Simpler representations for the phase trajectories can be obtained by displaying only the terminal values of the signal phase at the time instants $t = nT$. In this case, we restrict the modulation index of the CPM signal to be rational. In particular, let us assume that $h = m/p$, where m and p are relatively prime integers. Then, a full response CPM signal at the time instants $t = nT$ will have the terminal phase states

$$\Theta_s = \left\{0, \frac{\pi m}{p}, \frac{2\pi m}{p}, \ldots, \frac{(p-1)\pi m}{p}\right\} \tag{4-3-60}$$

when m is even and

$$\Theta_s = \left\{0, \frac{\pi m}{p}, \frac{2\pi m}{p}, \ldots, \frac{(2p-1)\pi m}{p}\right\} \tag{4-3-61}$$

when m is odd. Hence, there are p terminal phase states when m is even and $2p$ states when m is odd. On the other hand, when the pulse shape extends

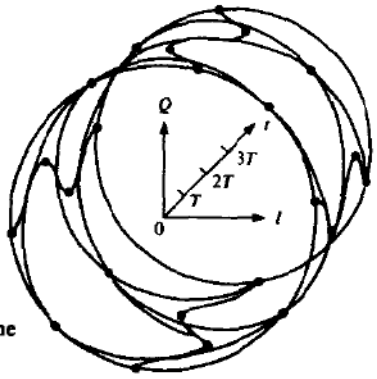


FIGURE 4-3-20 Phase cylinder for binary CPM with $h = \frac{1}{2}$ and a raised cosine pulse of length $3T$. [*From Sundberg (1986), © 1986 IEEE.*]

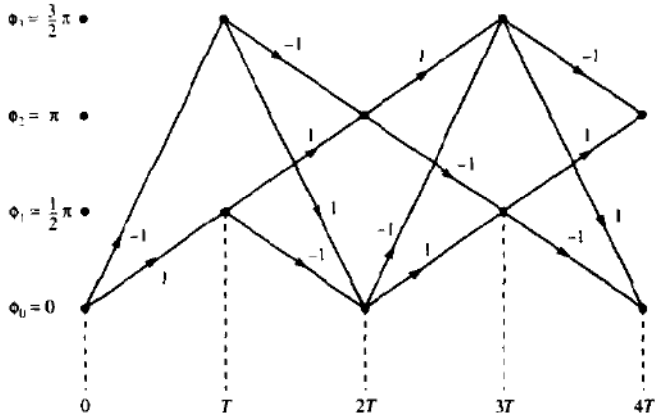


FIGURE 4-3-21 State trellis for binary CPFSK with $h = \frac{1}{2}$.

over L symbol intervals (partial response CPM), the number of phase states may increase up to a maximum of S_t, where

$$S_t = \begin{cases} pM^{L-1} & (\text{even } m) \\ 2pM^{L-1} & (\text{odd } m) \end{cases} \tag{4-3-62}$$

where M is the alphabet size. For example, the binary CPFSK signal (full response, rectangular pulse) with $h = \frac{1}{2}$, has $S_t = 4$ (terminal) phase states. The *state trellis* for this signal is illustrated in Fig. 4-3-21. We emphasize that the phase transitions from one state to another are not true phase trajectories. They represent phase transitions for the (terminal) states at the time instants $t = nT$.

An alternative representation to the state trellis is the state diagram, which also illustrates the state transitions at the time instants $t = nT$. This is an even more compact representation of the CPM signal characteristics. Only the possible (terminal) phase states and their transitions are displayed in the state diagram. Time does not appear explicitly as a variable. For example, the state diagram for the CPFSK signal with $h = \frac{1}{2}$ is shown in Fig. 4-3-22.

Minimum-Shift Keying (MSK) MSK is a special form of binary CPFSK

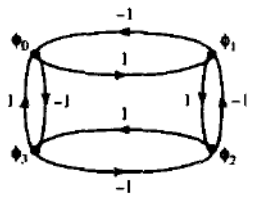


FIGURE 4-3-22 State diagram for binary CPFSK with $h = \frac{1}{2}$.

(and, therefore, CPM) in which the modulation index $h = \frac{1}{2}$. The phase of the carrier in the interval $nT \leq t \leq (n+1)T$ is

$$\begin{aligned}\phi(t; \mathbf{I}) &= \tfrac{1}{2}\pi \sum_{k=-\infty}^{n-1} I_k + \pi I_n q(t - nT) \\ &= \theta_n + \tfrac{1}{2}\pi I_n\left(\frac{t - nT}{T}\right), \quad nT \leq t \leq (n+1)T\end{aligned} \tag{4-3-63}$$

and the modulated carrier signal is

$$\begin{aligned}s(t) &= A\cos\left[2\pi f_c t + \theta_n + \tfrac{1}{2}\pi I_n\left(\frac{t - nT}{T}\right)\right] \\ &= A\cos\left[2\pi\left(f_c + \frac{1}{4T}I_n\right)t - \tfrac{1}{2}n\pi I_n + \theta_n\right], \quad nT \leq t \leq (n+1)T\end{aligned} \tag{4-3-64}$$

The expression (4-3-64) indicates that the binary CPFSK signal can be expressed as a sinusoid having one of two possible frequencies in the interval $nT \leq t \leq (n+1)T$. If we define these frequencies as

$$\begin{aligned}f_1 &= f_c - \frac{1}{4T} \\ f_2 &= f_c + \frac{1}{4T}\end{aligned} \tag{4-3-65}$$

then the binary CPFSK signal given by (4-3-64) may be written in the form

$$s_i(t) = A\cos[2\pi f_i t + \theta_n + \tfrac{1}{2}n\pi(-1)^{i-1}], \quad i = 1, 2 \tag{4-3-66}$$

The frequency separation $\Delta f = f_2 - f_1 = 1/2T$. Recall that $\Delta f = 1/2T$ is the minimum frequency separation that is necessary to ensure the orthogonality of the signals $s_1(t)$ and $s_2(t)$ over a signaling interval of length T. This explains why binary CPFSK with $h = \frac{1}{2}$ is called minimum-shift keying (MSK). The phase in the nth signaling interval is the phase state of the signal that results in phase continuity between adjacent intervals.

MSK may also be represented as a form of four-phase PSK. Specifically, we may express the equivalent lowpass digitally modulated signal in the form (see Problem 4-14)

$$v(t) = \sum_{n=-\infty}^{\infty} [I_{2n}g(t - 2nT) - jI_{2n+1}g(t - 2nT - T)] \tag{4-3-67}$$

where $g(t)$ is a sinusoidal pulse defined as

$$g(t) = \begin{cases} \sin\dfrac{\pi t}{2T} & (0 \leq t \leq 2T) \\ 0 & (\text{otherwise}) \end{cases} \tag{4-3-68}$$

Thus, this type of signal is viewed as a four-phase PSK signal in which the pulse shape is one-half cycle of a sinusoid. The even-numbered binary-valued (± 1) symbols $\{I_{2n}\}$ of the information sequence $\{I_n\}$ are transmitted via the cosine of the carrier, while the odd-numbered symbols $\{I_{2n+1}\}$ are transmitted via the sine of the carrier. The transmission rate on the two orthogonal carrier components is $1/2T$ bits per second so that the combined transmission rate is $1/T$ bits/s. Note that the bit transitions on the sine and cosine carrier components are staggered or offset in time by T seconds. For this reason, the signal

$$s(t) = A\left\{\left[\sum_{n=-\infty}^{\infty} I_{2n} g(t - 2nT)\right] \cos 2\pi f_c t + \left[\sum_{n=-\infty}^{\infty} I_{2n+1} g(t - 2nT - T)\right] \sin 2\pi f_c t\right\} \tag{4-3-69}$$

is called *offset quadrature PSK* (*OQPSK*) or *staggered quadrature PSK* (*SQPSK*).

Figure 4-3-23 illustrates the representation of the MSK signals as two staggered quadrature-modulated binary PSK signals. The corresponding sum of the two quadrature signals is a constant amplitude, frequency-modulated signal.

It is also interesting to compare the waveforms for MSK with offset QPSK in which the pulse $g(t)$ is rectangular for $0 \leq t \leq 2T$, and with conventional

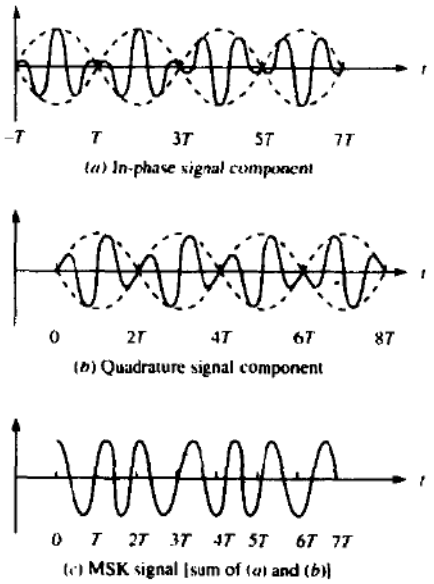


FIGURE 4-3-23 Representation of MSK signal as a form of two staggered binary PSK signals, each with a sinusoidal envelope.

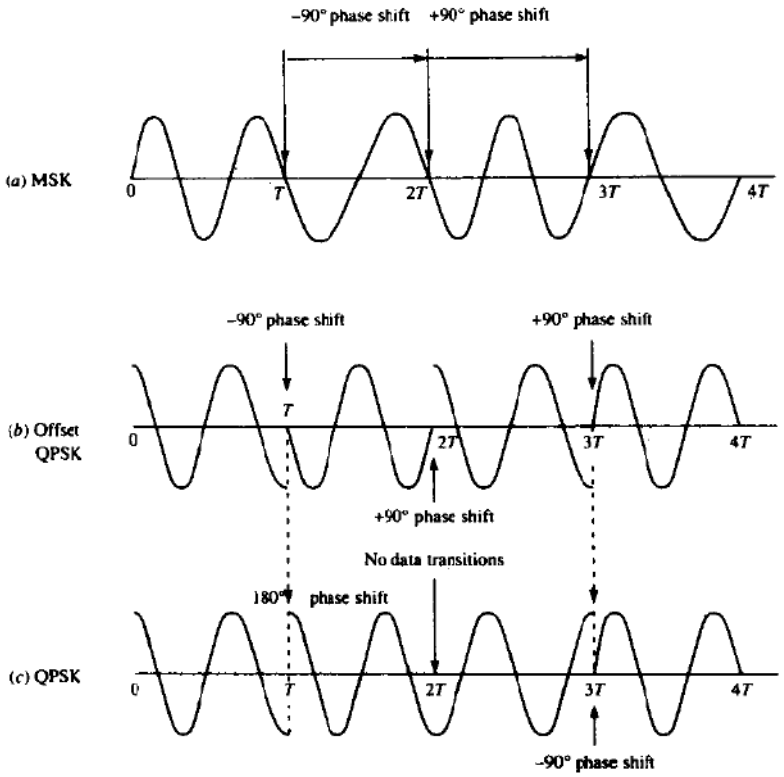


FIGURE 4-3-24 Signal waveforms for (*a*) MSK, (*b*) offset QPSK (rectangular pulse), and (*c*) conventional QPSK (rectangular pulse). [*From Gronemeyer and McBride (1976); © 1976 IEEE.*]

quadrature (four-phase) PSK (QPSK) in which the pulse $g(t)$ is rectangular for $0 \leq t \leq 2T$. Clearly, all three of the modulation methods result in identical data rates. The MSK signal has continuous phase. The offset QPSK signal with a rectangular pulse is basically two binary PSK signals for which the phase transitions are staggered in time by T seconds. Thus, the signal contains phase jumps of $\pm 90°$ that may occur as often as every T seconds. On the other hand, the conventional four-phase PSK signal with constant amplitude will contain phase jumps of $\pm 180°$ or $\pm 90°$ every $2T$ seconds. An illustration of these three signal types is given in Fig. 4-3-24.

Signal Space Diagrams for CPM In general, continuous-phase signals cannot be represented by discrete points in signal space as in the case of PAM, PSK, and QAM, because the phase of the carrier is time-variant. Instead, a continuous-phase signal is described by the various paths or trajectories from one phase state to another. For a constant-amplitude CPM signal, the various trajectories form a circle.

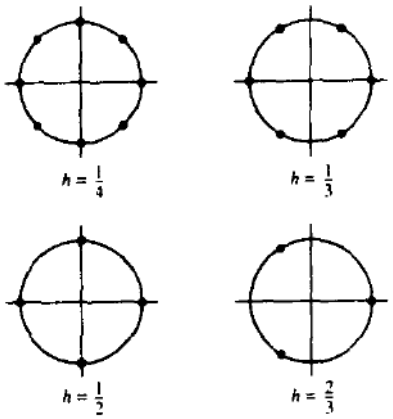


FIGURE 4-3-25 Signal space diagram for CPFSK.

For example, Fig. 4-3-25 illustrates the signal space (phase trajectory) diagram for CPFSK signals with $h = \frac{1}{4}$, $h = \frac{1}{3}$, $h = \frac{1}{2}$, and $h = \frac{2}{3}$. The beginning and ending points of these phase trajectories are marked in the figure by dots. Note that the length of the phase trajectory increases with an increase in h. An increase in h also results in an increase of the signal bandwidth, as demonstrated in the following section.

Multiamplitude CPM Multiamplitude CPM is a generalization of ordinary CPM in which the signal amplitude is allowed to vary over a set of amplitude values while the phase of the signal is constrained to be continuous. For example, let us consider a two-amplitude CPFSK signal, which may be represented as

$$s(t) = 2A \cos [2\pi f_c t + \phi_2(t; \mathbf{I})] + A \cos [2\pi f_c t + \phi_1(t; \mathbf{J})] \qquad (4\text{-}3\text{-}70)$$

where

$$\phi_2(t; \mathbf{I}) = \pi h \sum_{k=-\infty}^{n-1} I_k + \frac{\pi h I_n (t - nT)}{T}, \quad nT \leq t \leq (n+1)T \qquad (4\text{-}3\text{-}71)$$

$$\phi_1(t; \mathbf{J}) = \pi h \sum_{k=-\infty}^{n-1} J_k + \frac{\pi h J_n (t - nT)}{T}, \quad nT \leq t \leq (n+1)T \qquad (4\text{-}3\text{-}72)$$

The information is conveyed by the symbol sequences $\{I_n\}$ and $\{J_n\}$, which are related to two independent binary information sequences $\{a_n\}$ and $\{b_n\}$ that take values $\{0, 1\}$. We observe that the signal in (4-3-70) is a superposition of two CPFSK signals of different amplitude. However, the sequences $\{I_n\}$ and $\{J_n\}$ are not statistically independent, but are constrained in order to achieve phase continuity in the superposition of the two components.

To elaborate, let us consider the case where $h = \frac{1}{2}$, so that we have the superposition of two MSK signals. At the symbol transition points, the two

TABLE 4-3-1

a_n	b_n	I_n	J_n	Amplitude–phase relations
0	0	−1	−1	Amplitude is constant; phase decreases
0	1	−1	1	Amplitude changes; phase decreases
1	0	1	1	Amplitude is constant; phase increases
1	1	1	−1	Amplitude changes; phase increases

amplitude components are either in phase or 180° out of phase. The phase change in the signal is determined by the phase of the larger amplitude component, while the amplitude change is determined by the smaller component. Thus, the smaller component is constrained such that at the start and end of each symbol interval, it is either in phase or 180° out of phase with the larger component, independent of its phase. Under this constraint, the symbol sequences $\{I_n\}$ and $\{J_n\}$ may be expressed as

$$\begin{aligned} I_n &= 2a_n - 1 \\ J_n &= I_n(1 - 2b_n) = I_n\left(1 - \frac{b_n}{h}\right) \end{aligned} \tag{4-3-73}$$

These relationships are summarized in Table 4-3-1.

As a generalization, a multiamplitude CPFSK signal with n components may be expressed as

$$s(t) = 2^{N-1}\cos\left[2\pi f_c t + \phi_N(t; \mathbf{I})\right] + \sum_{m=1}^{N-1} 2^{m-1}\cos\left[2\pi f_c t + \phi_m(t; \mathbf{J}_m)\right] \tag{4-3-74}$$

where

$$\phi_N(t; \mathbf{I}) = \pi h I_n \frac{t - nT}{T} + \pi h \sum_{k=-\infty}^{n-1} I_k, \qquad nT \leq t \leq (n+1)T \tag{4-3-75}$$

and

$$\begin{aligned} \phi_m(t; \mathbf{J}_m) = {} & I_n \pi[h + \tfrac{1}{2}(J_{mn} + 1)]\frac{t - nT}{T} \\ & + \sum_{k=-\infty}^{n-1} \pi I_k[h + \tfrac{1}{2}(J_{mk} + 1)], \qquad nT \leq t \leq (n+1)T \end{aligned} \tag{4-3-76}$$

The sequences $\{I_n\}$ and $\{J_{mn}\}$ are statistically independent, binary-valued sequences that take values from the set $\{1, -1\}$.

From (4-3-75) and (4-3-76), we observe that each component in the sum

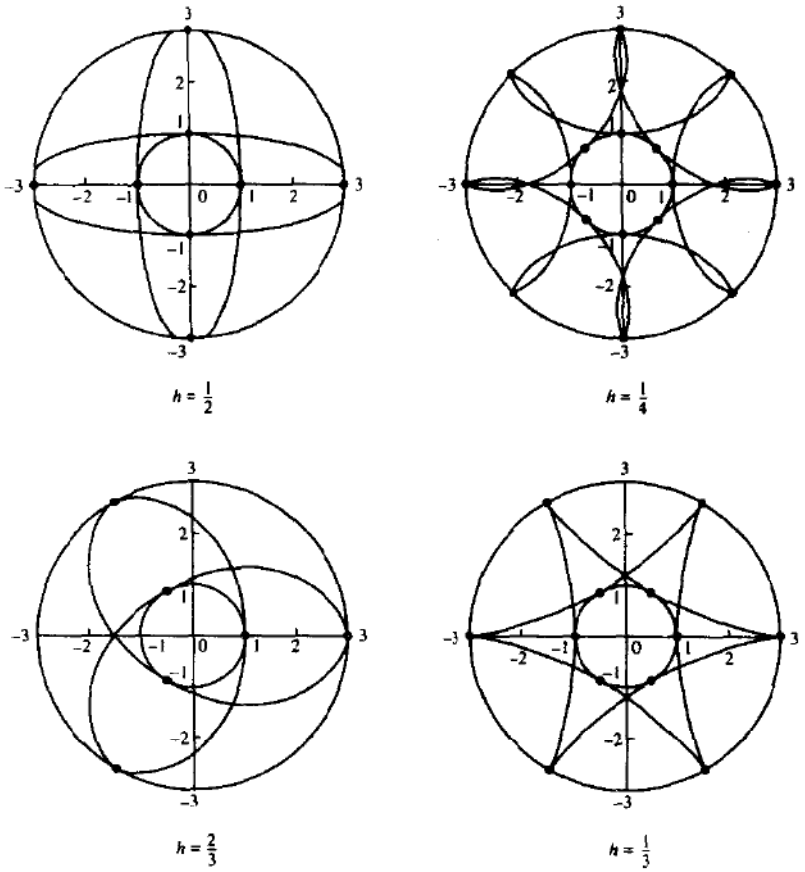


FIGURE 4-3-26 Signal space diagrams for two-component CPFSK.

will be either in phase or 180° out of phase with the largest component at the end of the nth symbol interval, i.e., at $t = (n + 1)T$. Thus, the signal states are specified by an amplitude level from the set of amplitudes $\{1, 3, 5, \ldots, 2^N - 1\}$ and a phase level from the set $\{0, \pi\theta, 2\pi\theta, \ldots, 2\pi - \pi h\}$. The phase constraint is required to maintain the phase continuity of the CPM signal.

Figure 4-3-26 illustrates the signal space diagrams for two-amplitude ($N = 2$) CPFSK with $h = \frac{1}{4}, \frac{1}{3}, \frac{1}{2}$, and $\frac{2}{3}$. The signal space diagrams for three-component ($N = 3$) CPFSK are shown in Fig. 4-3-27. In this case, there are *four* amplitude levels. The number of states depends on the modulation index h as well as N. Note that the beginning and ending points of the phase trajectories are marked by dots.

Additional multiamplitude CPM signal formats may be obtained by using

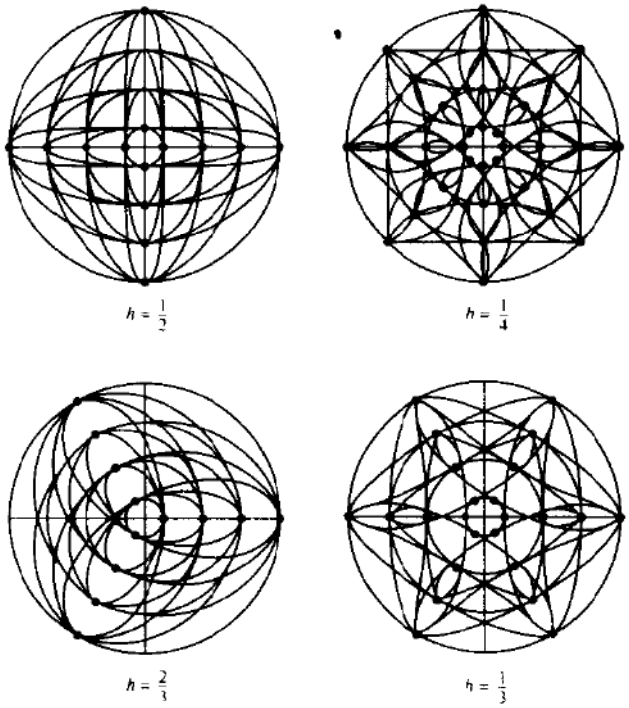


FIGURE 4-3-27 Signal space diagrams for three-component CPFSK.

pulse shapes other than rectangular, as well as signal pulses that span more than one symbol (partial response).

4-4 SPECTRAL CHARACTERISTICS OF DIGITALLY MODULATED SIGNALS

In most digital communications systems, the available channel bandwidth is limited. Consequently, the system designer must consider the constraints imposed by the channel bandwidth limitation in the selection of the modulation technique used to transmit the information. For this reason, it is important for us to determine the spectral content of the digitally modulated signals described in Section 4-3.

Since the information sequence is random, a digitally modulated signal is a stochastic process. We are interested in determining the power density spectrum of such a process. From the power density spectrum, we can determine the channel bandwidth required to transmit the information-bearing signal. Below, we first derive the spectral characteristics of the class of linearly

modulated signals. Then, we consider the nonlinear CPFSK, CPM, and baseband modulated signals with memory.

4-4-1 Power Spectra of Linearly Modulated Signals

Beginning with the form

$$s(t) = \text{Re}\,[v(t)e^{j2\pi f_c t}]$$

which relates the bandpass signal $s(t)$ to the equivalent lowpass signal $v(t)$, we may express the autocorrelation function of $s(t)$ as

$$\phi_{ss}(\tau) = \text{Re}\,[\phi_{vv}(\tau)e^{j2\pi f_c \tau}] \tag{4-4-1}$$

where $\phi_{vv}(\tau)$ is the autocorrelation function of the equivalent lowpass signal $v(t)$. The Fourier transform of (4-4-1) yields the desired expression for the power density spectrum $\Phi_{ss}(f)$ in the form

$$\Phi_{ss}(f) = \tfrac{1}{2}[\Phi_{vv}(f - f_c) + \Phi_{vv}(-f - f_c)] \tag{4-4-2}$$

where $\Phi_{vv}(f)$ is the power density spectrum of $v(t)$. It suffices to determine the autocorrelation function and the power density spectrum of the equivalent lowpass signal $v(t)$.

First we consider the linear digital modulation methods for which $v(t)$ is represented in the general form

$$v(t) = \sum_{n=-\infty}^{\infty} I_n g(t - nT) \tag{4-4-3}$$

where the transmission rate is $1/T = R/k$ symbols/s and $\{I_n\}$ represents the sequence of symbols that results from mapping k-bit blocks into corresponding signal points selected from the appropriate signal space diagram. Observe that in PAM, the sequence $\{I_n\}$ is real and corresponds to the amplitude values of the transmitted signal, but in PSK, QAM, and combined PAM–PSK, the sequence $\{I_n\}$ is complex-valued, since the signal points have a two-dimensional representation.

The autocorrelation function of $v(t)$ is

$$\begin{aligned}\phi_{vv}(t + \tau; t) &= \tfrac{1}{2}E[v^*(t)v(t + \tau)] \\ &= \tfrac{1}{2}\sum_{n=-\infty}^{\infty}\sum_{m=-\infty}^{\infty} E[I_n^* I_m]g^*(t - nT)g(t + \tau - mT)\end{aligned} \tag{4-4-4}$$

We assume that the sequence of information symbols $\{I_n\}$ is wide-sense stationary with mean μ_i and autocorrelation function

$$\phi_{ii}(m) = \tfrac{1}{2}E[I_n^* I_{n+m}] \tag{4-4-5}$$

Hence (4-4-4) can be expressed as

$$\phi_{vv}(t+\tau;t) = \sum_{n=-\infty}^{\infty} \sum_{m=-\infty}^{\infty} \phi_{ii}(m-n)g^*(t-nT)g(t+\tau-mT)$$

$$= \sum_{m=-\infty}^{\infty} \phi_{ii}(m) \sum_{n=-\infty}^{\infty} g^*(t-nT)g(t+\tau-nT-mT) \quad (4\text{-}4\text{-}6)$$

The second summation in (4-4-6), namely,

$$\sum_{n=-\infty}^{\infty} g^*(t-nT)g(t+\tau-nT-mT)$$

is periodic in the t variable with period T. Consequently, $\phi_{vv}(t+\tau;t)$ is also periodic in the t variable with period T. That is,

$$\phi_{vv}(t+T+\tau;t+T) = \phi_{vv}(t+\tau;t) \quad (4\text{-}4\text{-}7)$$

In addition, the mean value of $v(t)$, which is

$$E[v(t)] = \mu_i \sum_{n=-\infty}^{\infty} g(t-nT) \quad (4\text{-}4\text{-}8)$$

is periodic with period T. Therefore $v(t)$ is a stochastic process having a periodic mean and autocorrelation function. Such a process is called a *cyclostationary process* or a *periodically stationary process in the wide sense*, as described in Section 2-2-6.

In order to compute the power density spectrum of a cyclostationary process, the dependence of $\phi_{vv}(t+\tau;t)$ on the t variable must be eliminated. This can be accomplished simply by averaging $\phi_{vv}(t+\tau;t)$ over a single period. Thus,

$$\bar{\phi}_{vv}(\tau) = \frac{1}{T}\int_{-T/2}^{T/2} \phi_{vv}(t+\tau;t)\,dt$$

$$= \sum_{m=-\infty}^{\infty} \phi_{ii}(m) \sum_{n=-\infty}^{\infty} \frac{1}{T}\int_{-T/2}^{T/2} g^*(t-nT)g(t+\tau-nT-mT)\,dt$$

$$= \sum_{m=-\infty}^{\infty} \phi_{ii}(m) \sum_{n=-\infty}^{\infty} \frac{1}{T}\int_{-T/2-nT}^{T/2-nT} g^*(t)g(t+\tau-mT)\,dt \quad (4\text{-}4\text{-}9)$$

We interpret the integral in (4-4-9) as the time-autocorrelation function of $g(t)$ and define it as

$$\phi_{gg}(\tau) = \int_{-\infty}^{\infty} g^*(t)g(t+\tau)\,dt \quad (4\text{-}4\text{-}10)$$

Consequently (4-4-9) can be expressed as

$$\bar{\phi}_{vv}(\tau) = \frac{1}{T} \sum_{m=-\infty}^{\infty} \phi_{ii}(m)\phi_{gg}(\tau - mT) \tag{4-4-11}$$

The Fourier transform of the relation in (4-4-11) yields the (average) power density spectrum of $v(t)$ in the form

$$\Phi_{vv}(f) = \frac{1}{T}|G(f)|^2 \Phi_{ii}(f) \tag{4-4-12}$$

where $G(f)$ is the Fourier transform of $g(t)$, and $\Phi_{ii}(f)$ denotes the power density spectrum of the information sequence, defined as

$$\Phi_{ii}(f) = \sum_{m=-\infty}^{\infty} \phi_{ii}(m)e^{-j2\pi fmT} \tag{4-4-13}$$

The result (4-4-12) illustrates the dependence of the power density spectrum of $v(t)$ on the spectral characteristics of the pulse $g(t)$ and the information sequence $\{I_n\}$. That is, the spectral characteristics of $v(t)$ can be controlled by design of the pulse shape $g(t)$ and by design of the correlation characteristics of the information sequence.

Whereas the dependence of $\Phi_{vv}(f)$ on $G(f)$ is easily understood upon observation of (4-4-12), the effect of the correlation properties of the information sequence is more subtle. First of all, we note that for an arbitrary autocorrelation $\phi_{ii}(m)$ the corresponding power density spectrum $\Phi_{ii}(f)$ is periodic in frequency with period $1/T$. In fact, the expression (4-4-13) relating the spectrum $\Phi_{ii}(f)$ to the autocorrelation $\phi_{ii}(m)$ is in the form of an exponential Fourier series with the $\{\phi_{ii}(m)\}$ as the Fourier coefficients. As a consequence, the autocorrelation sequence $\phi_{ii}(m)$ is given by

$$\phi_{ii}(m) = T \int_{-1/2T}^{1/2T} \Phi_{ii}(f)e^{j2\pi fmT}\, df \tag{4-4-14}$$

Second, let us consider the case in which the information symbols in the sequence are real and mutually uncorrelated. In this case, the autocorrelation function $\phi_{ii}(m)$ can be expressed as

$$\phi_{ii}(m) = \begin{cases} \sigma_i^2 + \mu_i^2 & (m = 0) \\ \mu_i^2 & (m \neq 0) \end{cases} \tag{4-4-15}$$

where σ_i^2 denotes the variance of an information symbol. When (4-4-15) is used to substitute for $\phi_{ii}(m)$ in (4-4-13), we obtain

$$\Phi_{ii}(f) = \sigma_i^2 + \mu_i^2 \sum_{m=-\infty}^{\infty} e^{-j2\pi fmT} \tag{4-4-16}$$

The summation in (4-4-16) is periodic with period $1/T$. It may be viewed as

the exponential Fourier series of a periodic train of impulses with each impulse having an area $1/T$. Therefore (4-4-16) can also be expressed in the form

$$\Phi_{ii}(f) = \sigma_i^2 + \frac{\mu_i^2}{T} \sum_{m=-\infty}^{\infty} \delta\left(f - \frac{m}{T}\right) \tag{4-4-17}$$

Substitution of (4-4-17) into (4-4-12) yields the desired result for the power density spectrum of $v(t)$ when the sequence of information symbols is uncorrelated. That is,

$$\Phi_{vv}(f) = \frac{\sigma_i^2}{T}|G(f)|^2 + \frac{\mu_i^2}{T^2} \sum_{m=-\infty}^{\infty} \left|G\left(\frac{m}{T}\right)\right|^2 \delta\left(f - \frac{m}{T}\right) \tag{4-4-18}$$

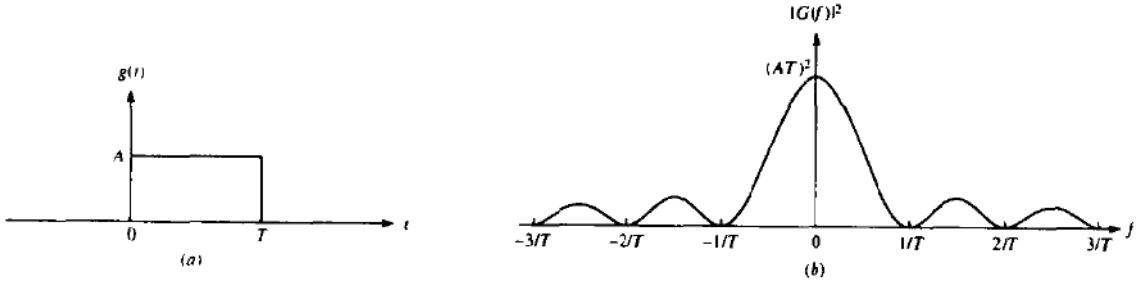
The expression (4-4-18) for the power density spectrum is purposely separated into two terms to emphasize the two different types of spectral components. The first term is the continuous spectrum, and its shape depends only on the spectral characteristic of the signal pulse $g(t)$. The second term consists of discrete frequency components spaced $1/T$ apart in frequency. Each spectral line has a power that is proportional to $|G(f)|^2$ evaluated at $f = m/T$. Note that the discrete frequency components vanish when the information symbols have zero mean, i.e., $\mu_i = 0$. This condition is usually desirable for the digital modulation techniques under consideration, and it is satisfied when the information symbols are equally likely and symmetrically positioned in the complex plane. Thus, the system designer can control the spectral characteristics of the digitally modulated signal by proper selection of the characteristics of the information sequence to be transmitted.

Example 4-4-1

To illustrate the spectral shaping resulting from $g(t)$, consider the rectangular pulse shown in Fig. 4-4-1(a). The Fourier transform of $g(t)$ is

$$G(f) = AT \frac{\sin \pi f T}{\pi f T} e^{-j\pi f T}$$

FIGURE 4-4-1 Rectangular pulse and its energy density spectrum $|G(f)|^2$.



Hence

$$|G(f)|^2 = (AT)^2\left(\frac{\sin \pi fT}{\pi fT}\right)^2 \tag{4-4-19}$$

This spectrum is illustrated in Fig. 4-4-1(*b*). Note that it contains zeros at multiples of $1/T$ in frequency and that it decays inversely as the square of the frequency variable. As a consequence of the spectral zeros in $G(f)$, all but one of the discrete spectral components in (4-4-18) vanish. Thus, upon substitution for $|G(f)|^2$ from (4-4-19), (4-4-18) reduces to

$$\Phi_{vv}(f) = \sigma_i^2 A^2 T\left(\frac{\sin \pi fT}{\pi fT}\right)^2 + A^2\mu_i^2\delta(f) \tag{4-4-20}$$

Example 4-4-2

As a second illustration of the spectral shaping resulting from $g(t)$, we consider the raised cosine pulse

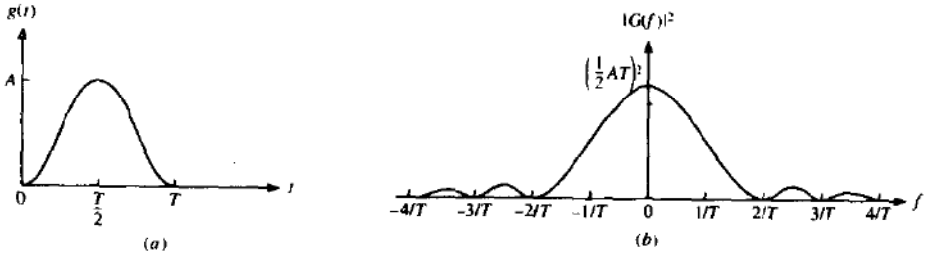
$$g(t) = \frac{A}{2}\left[1 + \cos\frac{2\pi}{T}\left(t - \frac{T}{2}\right)\right], \quad 0 \leq t \leq T \tag{4-2-21}$$

This pulse is graphically illustrated in Fig. 4-4-2(*a*). Its Fourier transform is easily derived and it may be expressed in the form

$$G(f) = \frac{AT}{2}\frac{\sin \pi fT}{\pi fT(1 - f^2T^2)}e^{-j\pi fT} \tag{4-4-22}$$

The square of the magnitude of $G(f)$ is shown in Fig. 4-4-2(*b*). It is interesting to note that the spectrum has zeros at $f = n/T$, $n = \pm 2, \pm 3, \pm 4, \ldots$. Consequently, all the discrete spectral components in (4-4-18), except the ones at $f = 0$ and $f = \pm 1/T$, vanish. When compared with the

FIGURE 4-4-2 Raised cosine pulse and its energy density spectrum $|G(f)|^2$.



spectrum of the rectangular pulse, the spectrum of the raised cosine pulse has a broader main lobe but the tails decay inversely as f^6.

Example 4-4-3

To illustrate that spectral shaping can also be accomplished by operations performed on the input information sequence, we consider a binary sequence $\{b_n\}$ from which we form the symbols

$$I_n = b_n + b_{n-1} \tag{4-4-23}$$

The $\{b_n\}$ are assumed to be uncorrelated random variables, each having zero mean and unit variance. Then the autocorrelation function of the sequence $\{I_n\}$ is

$$\begin{aligned}\phi_{ii}(m) &= E(I_n I_{n+m}) \\ &= \begin{cases} 2 & (m = 0) \\ 1 & (m = \pm 1) \\ 0 & \text{(otherwise)} \end{cases}\end{aligned} \tag{4-4-24}$$

Hence, the power density spectrum of the input sequence is

$$\begin{aligned}\Phi_{ii}(f) &= 2(1 + \cos 2\pi fT) \\ &= 4\cos^2 \pi fT\end{aligned} \tag{4-4-25}$$

and the corresponding power density spectrum for the (lowpass) modulated signal is

$$\Phi_{vv}(f) = \frac{4}{T}|G(f)|^2 \cos^2 \pi fT \tag{4-4-26}$$

4-4-2 Power Spectra of CPFSK and CPM Signals

In this section, we derive the power density spectrum for the class of constant amplitude CPM signals that were described in Section 4-3-3. We begin by computing the autocorrelation function and its Fourier transform, as was done in the case of linearly modulated signals.

The constant amplitude CPM signal is expressed as

$$s(t; \mathbf{I}) = A \cos [2\pi f_c t + \phi(t; \mathbf{I})] \tag{4-4-27}$$

where

$$\phi(t; \mathbf{I}) = 2\pi h \sum_{k=-\infty}^{\infty} I_k q(t - kT) \tag{4-4-28}$$

Each symbol in the sequence $\{I_n\}$ can take one of the M values $\{\pm 1, \pm 3, \ldots, \pm(M-1)\}$. These symbols are statistically independent and identically distributed with prior probabilities

$$P_n = P(I_k = n), \qquad n = \pm 1, \pm 3, \ldots, \pm(M-1) \tag{4-4-29}$$

where $\sum_n P_n = 1$. The pulse $g(t) = q'(t)$ is zero outside of the interval $[0, LT]$, $q(t) = 0$, $t < 0$, and $q(t) = \frac{1}{2}$ for $t > LT$.

The autocorrelation function of the equivalent lowpass signal

$$v(t) = e^{j\phi(t;\mathbf{I})}$$

is

$$\phi_{vv}(t+\tau; t) = \tfrac{1}{2}E\left[\exp\left(j2\pi h \sum_{k=-\infty}^{\infty} I_k[q(t+\tau-kT) - q(t-kT)]\right)\right] \tag{4-4-30}$$

First we express the sum in the exponent as a product of exponents. The result is

$$\phi_{vv}(t+\tau; t) = \tfrac{1}{2}E\left(\prod_{k=-\infty}^{\infty} \exp\{j2\pi h I_k[q(t+\tau-kT) - q(t-kT)]\}\right) \tag{4-4-31}$$

Next, we perform the expectation over the data symbols $\{I_k\}$. Since these symbols are statistically independent, we obtain

$$\phi_{vv}(t+\tau; t) = \tfrac{1}{2}\prod_{k=-\infty}^{\infty}\left(\sum_{\substack{n=-(M-1)\\ n \text{ odd}}}^{M-1} P_n \exp\{j2\pi hn[q(t+\tau-kT) - q(t-kT)]\}\right) \tag{4-4-32}$$

Finally, the average autocorrelation function is

$$\bar{\phi}_{vv}(\tau) = \frac{1}{T}\int_0^T \phi_{vv}(t+\tau; t)\, dt \tag{4-4-33}$$

Although (4-4-32) implies that there are an infinite number of factors in the product, the pulse $g(t) = q'(t) = 0$ for $t < 0$ and $t > LT$, and $q(t) = 0$ for $t < 0$. Consequently only a finite number of terms in the product have nonzero exponents. Thus, (4-4-32) can be simplified considerably. In addition, if we let $\tau = \xi + mT$, where $0 \le \xi < T$ and $m = 0, 1, \ldots$, the average autocorrelation in (4-4-33) reduces to

$$\bar{\phi}_{vv}(\xi + mT) = \frac{1}{2T}\int_0^T \prod_{k=1-L}^{m-1}\left(\sum_{\substack{n=-(M-1)\\ n \text{ odd}}}^{M-1} P_n \exp\{j2\pi hn[q(t+\xi-(k-m)T) - q(t-kT)]\}\right) \tag{4-4-34}$$

Let us focus on $\bar{\phi}_{vv}(\xi + mT)$ for $\xi + mT \geq LT$. In this case, (4-4-34) may be expressed as

$$\bar{\phi}_{vv}(\xi + mT) = [\psi(jh)]^{m-L}\lambda(\xi), \quad m \geq L, \quad 0 \leq \xi < T \tag{4-4-35}$$

where $\psi(jh)$ is the characteristic function of the random sequence $\{I_n\}$, defined as

$$\begin{aligned}\psi(jh) &= E(e^{jnhI_n}) \\ &= \sum_{\substack{n=-(M-1) \\ n \text{ odd}}}^{M-1} P_n e^{jnhn}\end{aligned} \tag{4-4-36}$$

and $\lambda(\xi)$ is the remaining part of the average autocorrelation function, which may be expressed as

$$\begin{aligned}\lambda(\xi) = \frac{1}{2T}\int_0^T \prod_{k=1-L}^{0}\left(\sum_{\substack{n=-(M-1) \\ n \text{ odd}}}^{M-1} P_n \exp\{j2\pi hn[\tfrac{1}{2} - q(t-kT)]\}\right) \\ \times \prod_{k=1-L}^{1}\left(\sum_{\substack{n=-(M-1) \\ n \text{ odd}}}^{M-1} P_n \exp[j2\pi hnq(t+\xi-kT)]\right)dt, \quad m \geq L\end{aligned} \tag{4-4-37}$$

Thus, $\bar{\phi}_{vv}(\tau)$ may be separated into a product of $\lambda(\xi)$ and $\psi(jh)$ as indicated in (4-4-35) for $\tau = \xi + mT \geq LT$ and $0 \leq \xi < T$. This property is used below.

The Fourier transform of $\bar{\phi}_{vv}(\tau)$ yields the average power density spectrum as

$$\begin{aligned}\Phi_{vv}(f) &= \int_{-\infty}^{\infty} \bar{\phi}_{vv}(\tau)e^{-j2\pi f\tau}\,d\tau \\ &= 2\,\mathrm{Re}\left[\int_0^{\infty} \bar{\phi}_{vv}(\tau)e^{-j2\pi f\tau}\,d\tau\right]\end{aligned} \tag{4-4-38}$$

But

$$\begin{aligned}\int_0^{\infty} \bar{\phi}_{vv}(\tau)e^{-j2\pi f\tau}\,d\tau &= \int_0^{LT} \bar{\phi}_{vv}(\tau)e^{-j2\pi f\tau}\,d\tau \\ &\quad + \int_{LT}^{\infty} \bar{\phi}_{vv}(\tau)e^{-j2\pi f\tau}\,d\tau\end{aligned} \tag{4-3-39}$$

With the aid of (4-4-35), the integral in the range $LT \leq \tau < \infty$ may be expressed as

$$\int_{LT}^{\infty} \bar{\phi}_{vv}(\tau)e^{-2\pi f\tau}\,d\tau = \sum_{m=L}^{\infty}\int_{mT}^{(m+1)T} \bar{\phi}_{vv}(\tau)e^{-j2\pi f\tau}\,d\tau \tag{4-4-40}$$

Now, let $\tau = \xi + mT$. Then (4-4-40) becomes

$$\begin{aligned}\int_{LT}^{\infty} \bar{\phi}_{vv}(\tau)e^{-j2\pi f\tau}\,dt &= \sum_{m=L}^{\infty}\int_0^T \bar{\phi}_{vv}(\xi+mT)e^{-j2\pi f(\xi+mT)}\,d\xi \\ &= \sum_{m=L}^{\infty}\int_0^T \lambda(\xi)[\psi(jh)]^{m-L}e^{-j2\pi f(\xi+mT)}\,d\xi \\ &= \sum_{n=0}^{\infty}\psi^n(jh)e^{-j2\pi fnT}\int_0^T \lambda(\xi)e^{-j2\pi f(\xi+LT)}\,d\xi \qquad (4\text{-}4\text{-}41)\end{aligned}$$

A property of the characteristic function is $|\psi(jh)| \leq 1$. For values of h for which $|\psi(jh)| < 1$, the summation in (4-4-41) converges and yields

$$\sum_{n=0}^{\infty}\psi^n(jh)e^{-j2\pi fnT} = \frac{1}{1-\psi(jh)e^{-j2\pi fT}} \qquad (4\text{-}4\text{-}42)$$

In this case, (4-4-41) reduces to

$$\int_{LT}^{\infty}\bar{\phi}_{vv}(\tau)e^{-j2\pi f\tau}\,dt = \frac{1}{1-\psi(jh)e^{-j2\pi fT}}\int_0^T \bar{\phi}_{vv}(\xi+LT)e^{-j2\pi f(\xi+LT)}\,d\xi \qquad (4\text{-}4\text{-}43)$$

By combining (4-4-38), (4-4-39), and (4-4-43), we obtain the power density spectrum of the CPM signal in the form

$$\Phi_{vv}(f) = 2\,\mathrm{Re}\left[\int_0^{LT}\bar{\phi}_{vv}(\tau)e^{-j2\pi f\tau}\,d\tau + \frac{1}{1-\psi(jh)e^{-j2\pi fT}}\int_{LT}^{(L+1)T}\bar{\phi}_{vv}(\tau)e^{-j2\pi f\tau}\,d\tau\right] \qquad (4\text{-}4\text{-}44)$$

This is the desired result when $|\psi(jh)| < 1$. In general, the power density spectrum is evaluated numerically from (4-4-44). The average autocorrelation function $\bar{\phi}_{vv}(\tau)$ for the range $0 \leq \tau \leq (L+1)T$ may be computed numerically from (4-4-34).

For values of h for which $|\psi(jh)| = 1$, e.g., $h = K$, where K is an integer, we can set

$$\psi(jh) = e^{j2\pi\nu}, \qquad 0 \leq \nu < 1 \qquad (4\text{-}4\text{-}45)$$

Then, the sum in (4-4-41) becomes

$$\sum_{n=0}^{\infty}e^{-j2\pi T(f-\nu/T)n} = \tfrac{1}{2} + \frac{1}{2T}\sum_{n=-\infty}^{\infty}\delta\left(f-\frac{\nu}{T}-\frac{n}{T}\right) - j\tfrac{1}{2}\cot\pi T\left(f-\frac{\nu}{T}\right) \qquad (4\text{-}4\text{-}46)$$

Thus, the power density spectrum now contains impulses located at frequencies

$$f_n = \frac{n+\nu}{T}, \quad 0 \leq \nu < 1, \quad n = 0, 1, 2, \ldots \tag{4-4-47}$$

The result (4-4-46) can be combined with (4-4-41) and (4-4-39) to obtain the entire power density spectrum, which includes both a continuous spectrum component and a discrete spectrum component.

Let us return to the case for which $|\psi(jh)| < 1$. When the symbols are equally probable, i.e.,

$$P_n = \frac{1}{M} \quad \text{for all } n$$

the characteristic function simplifies to the form

$$\begin{aligned} \psi(jh) &= \frac{1}{M} \sum_{\substack{n=-(M-1) \\ n \text{ odd}}}^{M-1} e^{j\pi hn} \\ &= \frac{1}{M} \frac{\sin M\pi h}{\sin \pi h} \end{aligned} \tag{4-4-48}$$

Note that in this case $\psi(jh)$ is real. The average autocorrelation function given by (4-4-34) also simplifies in this case to

$$\bar{\phi}_{vv}(\tau) = \frac{1}{2T} \int_0^T \prod_{k=1-L}^{\lfloor \tau/T \rfloor} \frac{1}{M} \frac{\sin 2\pi hM[q(t+\tau-kT) - q(t-kT)]}{\sin 2\pi h[q(t+\tau-kT) - q(t-kT)]} dt \tag{4-4-49}$$

The corresponding expression for the power density spectrum reduces to

$$\begin{aligned} \Phi_{vv}(f) = 2\Bigg[&\int_0^{LT} \bar{\phi}_{vv}(\tau) \cos 2\pi f\tau \, d\tau \\ &+ \frac{1 - \psi(jh)\cos 2\pi fT}{1 + \psi^2(jh) - 2\psi(jh)\cos 2\pi fT} \int_{LT}^{(L+1)T} \bar{\phi}_{vv}(\tau) \cos 2\pi f\tau \, d\tau \Bigg] \end{aligned} \tag{4-4-50}$$

Power Density Spectrum of CPFSK A closed-form expression for the power density spectrum can be obtained from (4-4-50) when the pulse shape $g(t)$ is rectangular and zero outside the interval $[0, T]$. In this case, $q(t)$ is linear for $0 \leq t \leq T$. The resulting power spectrum may be expressed as

$$\Phi_{vv}(f) = T\left[\frac{1}{M}\sum_{n=1}^{M} A_n^2(f) + \frac{2}{M^2}\sum_{n=1}^{M}\sum_{m=1}^{M} B_{nm}(f)A_n(f)A_m(f)\right] \tag{4-4-51}$$

where

$$A_n(f) = \frac{\sin \pi[fT - \frac{1}{2}(2n - 1 - M)h]}{\pi[fT - \frac{1}{2}(2n - 1 - M)h]}$$

$$B_{nm}(f) = \frac{\cos(2\pi fT - \alpha_{nm}) - \psi \cos \alpha_{nm}}{1 + \psi^2 - 2\psi \cos 2\pi fT} \tag{4-4-52}$$

$$\alpha_{nm} = \pi h(m + n - 1 - M)$$

$$\psi \equiv \psi(jh) = \frac{\sin M\pi h}{M \sin \pi h}$$

The power density spectrum of CPFSK for $M = 2$, 4, and 8 is plotted in Figs 4-4-3, 4-4-4, and 4-4-5 as a function of the normalized frequency fT, with the modulation index $h = 2f_dT$ as a parameter. Note that only one-half of the bandwidth occupancy is shown in these graphs. The origin corresponds to the carrier f_c. The graphs illustrate that the spectrum of CPFSK is relatively smooth and well confined for $h < 1$. As h approaches unity, the spectra become very peaked and, for $h = 1$ when $|\psi| = 1$, we find that impulses occur at M frequencies. When $h > 1$ the spectrum becomes much broader. In communication systems where CPFSK is used, the modulation index is designed to conserve bandwidth, so that $h < 1$.

The special case of binary CPFSK with $h = \frac{1}{2}$ (or $f_d = 1/4T$) and $\psi = 0$ corresponds to MSK. In this case, the spectrum of the signal is

$$\Phi_{vv}(f) = \frac{16A^2T}{\pi^2}\left(\frac{\cos 2\pi fT}{1 - 16f^2T^2}\right)^2 \tag{4-4-53}$$

where the signal amplitude $A = 1$ in (4-4-52). In contrast the spectrum of four-phase offset (quadrature) PSK (OQPSK) with a rectangular pulse $g(t)$ of duration T is

$$\Phi_{vv}(f) = A^2T\left(\frac{\sin \pi fT}{\pi fT}\right)^2 \tag{4-4-54}$$

If we compare these spectral characteristics, we should normalize the frequency variable by the bit rate or the bit interval T_b. Since MSK is binary FSK, it follows that $T = T_b$ in (4-4-53). On the other hand, in OQPSK, $T = 2T_b$ so that (4-4-54) becomes

$$\Phi_{vv}(f) = 2A^2T_b\left(\frac{\sin 2\pi fT_b}{2\pi fT_b}\right)^2 \tag{4-4-55}$$

The spectra of the MSK and OQPSK signals are illustrated in Fig. 4-4-6. Note that the main lobe of MSK is 50% wider than that for OQPSK. However, the side lobes in MSK fall off considerably faster. For example, if we compare the bandwidth W that contains 99% of the total power, we find that $W = 1.2/T_b$ for MSK and $W \approx 8/T_b$ for OQPSK. Consequently, MSK has a narrower spectral

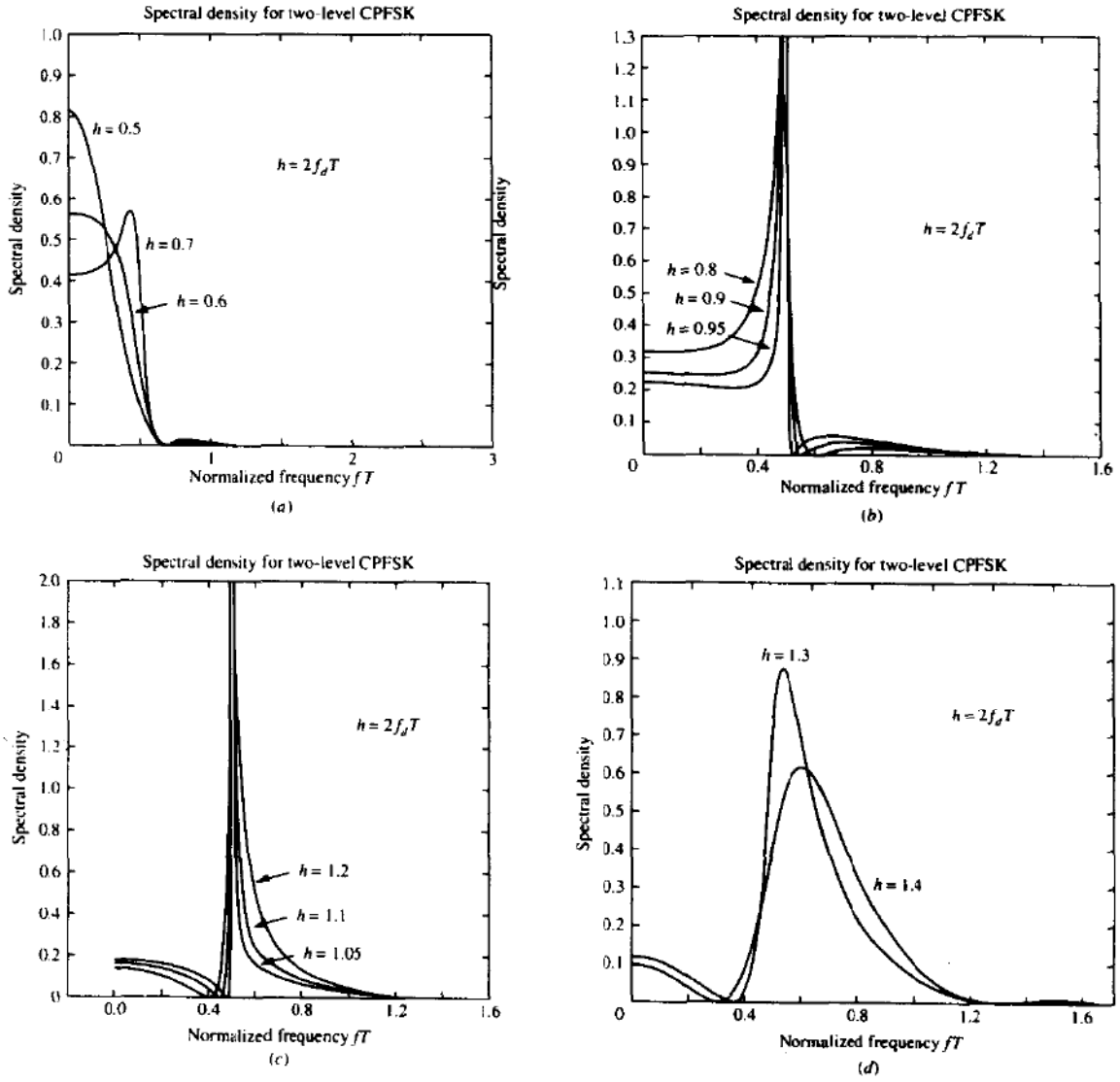


FIGURE 4-4-3 Power density spectrum of binary CPFSK.

occupancy when viewed in terms of fractional out-of-band power above $fT_b = 1$. Graphs for the fractional out-of-band power for OQPSK and MSK are shown in Fig. 4-4-7. Note that MSK is significantly more bandwidth-efficient than QPSK. This efficiency accounts for the popularity of MSK in many digital communications systems.

Even greater bandwidth efficiency than MSK can be achieved by reducing

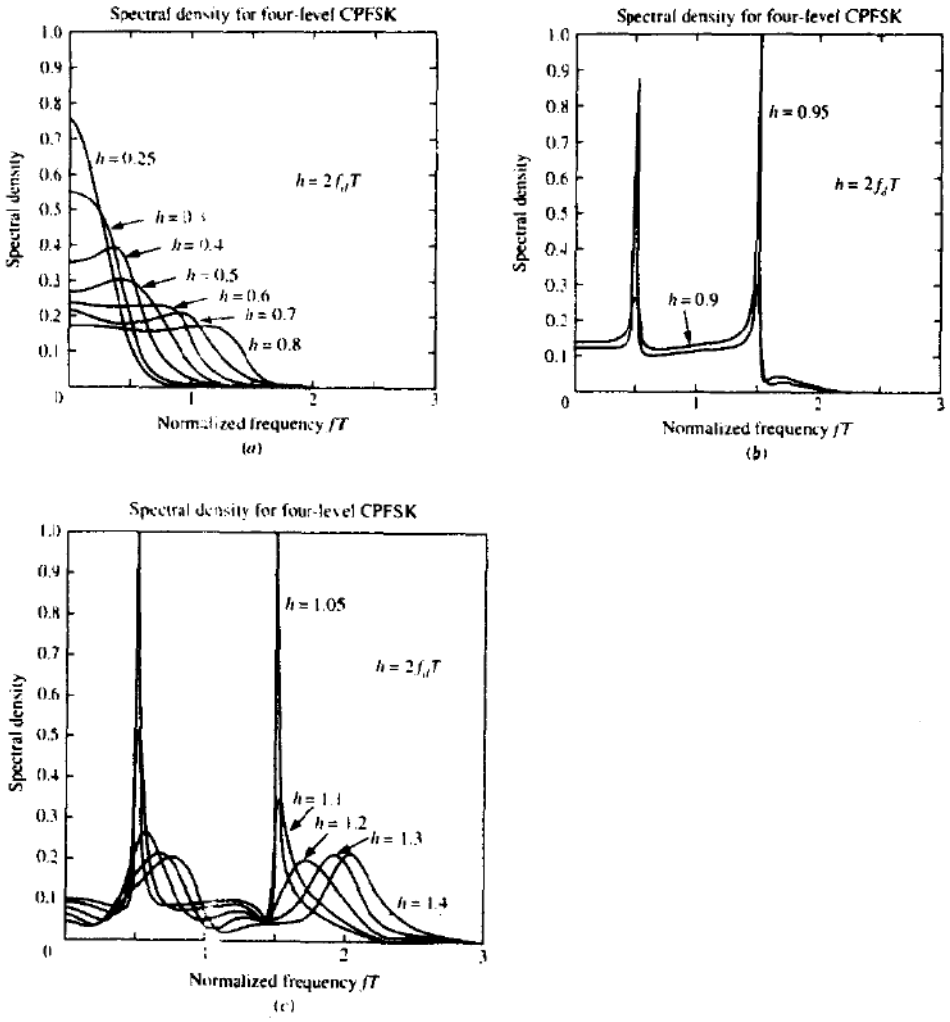


FIGURE 4-4-4 Power density spectrum of quaternary CPFSK.

the modulation index. However, the FSK signals will no longer be orthogonal and there will be an increase in the error probability.

Spectral Characteristics of CPM In general, the bandwidth occupancy of CPM depends on the choice of the modulation index h, the pulse shape $g(t)$, and the number of signals M. As we have observed for CPFSK, small values of h result in CPM signals with relatively small bandwidth occupancy, while large values of h result in signals with large bandwidth occupancy. This is also the case for the more general CPM signals.

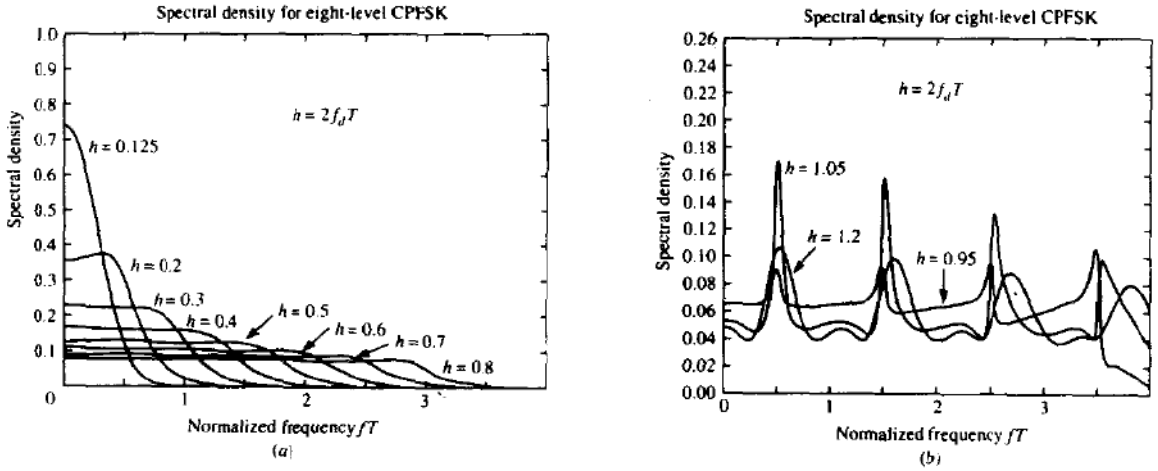
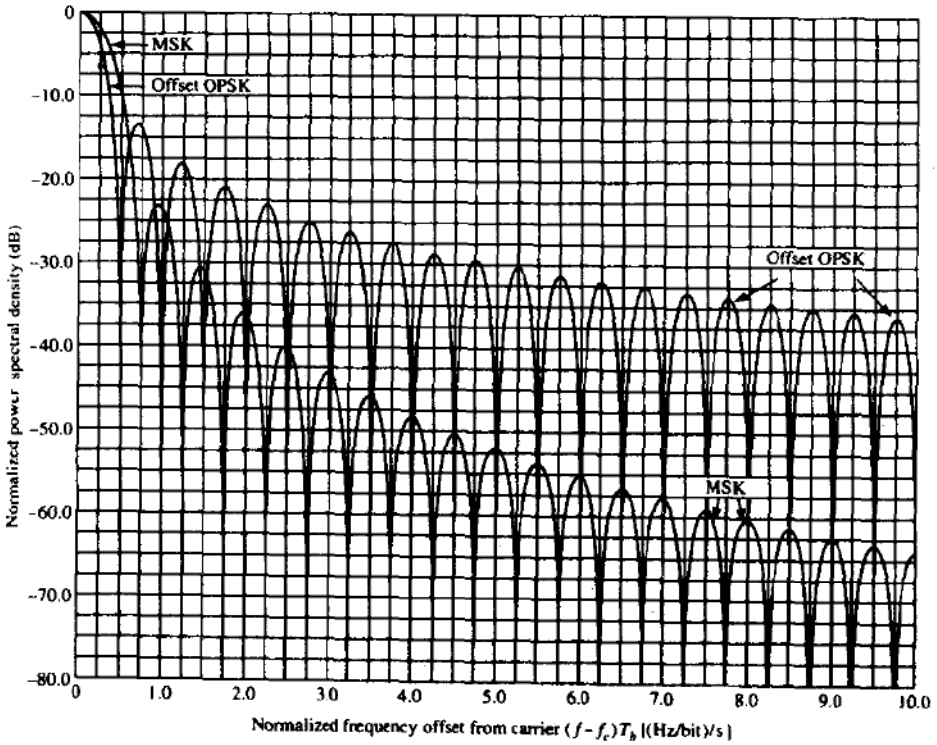


FIGURE 4-4-5 Power density spectrum of octal CPFSK.

FIGURE 4-4-6 Power density spectra of MSK and offset QPSK. [*From Gronemeyer and McBride (1976), © IEEE.*]



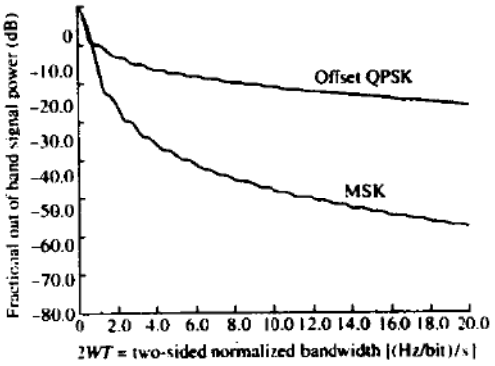


FIGURE 4-4-7 Fractional out-of-band power (normalized two-sided bandwidth = 2 BT). [*From Gronemeyer and McBride (1976); © 1976 IEEE.*]

The use of smooth pulses such as raised cosine pulses of the form

$$g(t) = \begin{cases} \dfrac{1}{2LT}\left(1 - \cos\dfrac{2\pi t}{LT}\right) & (0 \le t \le LT) \\ 0 & (\text{otherwise}) \end{cases} \tag{4-4-56}$$

where $L = 1$ for full response and $L > 1$ for partial response, result in smaller bandwidth occupancy and, hence, greater bandwidth efficiency than the use of rectangular pulses. For example, Fig. 4-4-8 illustrates the power density spectrum for binary CPM with different partial response raised cosine (LRC) pulses when $h = \frac{1}{2}$. For comparison, the spectrum of binary CPFSK is also shown. Note that as L increases the pulse $g(t)$ becomes smoother and the corresponding spectral occupancy of the signal is reduced.

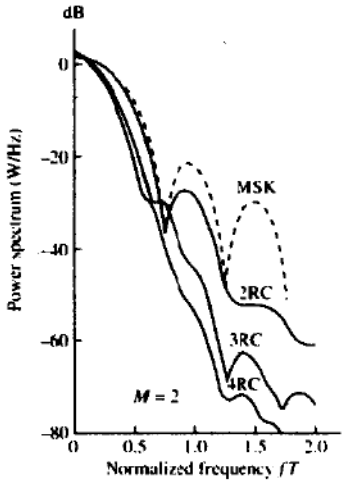


FIGURE 4-4-8 Power density spectrum for binary CPM with $h = \frac{1}{2}$ and different pulse shapes. [*From Aulin* et al. *(1981); © 1981 IEEE.*]

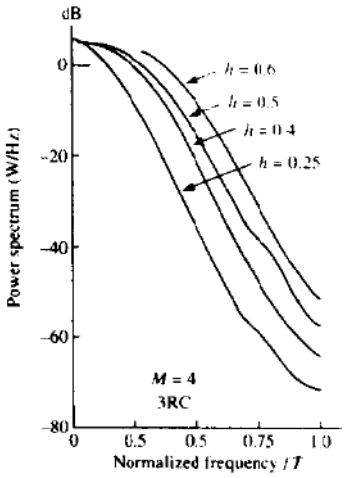
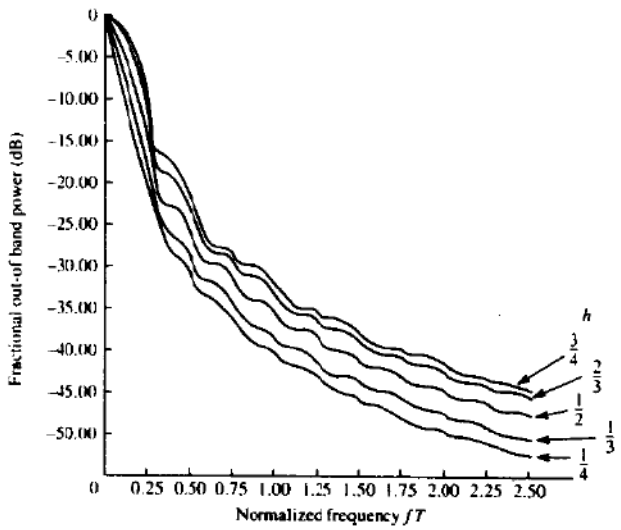


FIGURE 4-4-9 Power density spectrum for $M = 4$ CPM with $3RC$ and different modulation indices. [*From Aulin* et al. *(1981);* © *1981 IEEE.*]

The effect of varying the modulation index in a CPM signal is illustrated in Fig. 4-4-9 for the case of $M = 4$ and a raised cosine pulse of the form given in (4-4-56) with $L = 3$. Note that these spectral characteristics are similar to the ones illustrated previously for CPFSK, except that these spectra are narrower due to the use of a smoother pulse shape.

Finally, in Fig. 4-4-10, we illustrate the fractional out-of-band power for two-amplitude CPFSK with several different values of h.

FIGURE 4-4-10 Fractional out-of-band power for two-component CPFSK. (*Mulligan, 1988.*)



4-4-3 Power Spectra of Modulated Signals with Memory

In the last two sections, we have determined the spectral characteristics for the class of linearly modulated signals without memory and for the class of angle-modulated signals such as CPFSK and CPM, which are nonlinear and possess memory. In this section, we consider the spectral characteristics of linearly modulated signals that have memory that can be modeled by a Markov chain. We have already encountered such signals in Section 4-3-2, where we described several types of baseband signals.

The power density spectrum of a digitally modulated signal that is generated by a Markov chain may be derived by following the basic procedure given in the previous section. Thus, we can determine the autocorrelation function and then evaluate its Fourier transform to obtain the power density spectrum. For signals that are generated by a Markov chain with transition probability matrix $\mathbf{P}$, the power density spectrum of the modulated signal may be expressed in the general form (see Titsworth and Welch, 1961)

$$\Phi(f) = \frac{1}{T^2}\sum_{n=-\infty}^{\infty}\left|\sum_{i=1}^{K} p_i S_i\left(\frac{n}{T}\right)\right|^2 \delta\left(f-\frac{n}{T}\right) + \frac{1}{T}\sum_{i=1}^{K} p_i |S_i'(f)|^2 + \frac{2}{T}\mathrm{Re}\left[\sum_{i=1}^{K}\sum_{j=1}^{K} p_i S_i'^*(f) S_j'(f) P_{ij}(f)\right] \tag{4-4-57}$$

where $S_i(f)$ is the Fourier transform of the signal waveform $s_i(t)$,

$$s_i'(t) = s_i(t) - \sum_{k=1}^{K} p_k s_k(t)$$

$P_{ij}(f)$ is the Fourier transform of the discrete-time sequence $p_{ij}(n)$, defined as

$$P_{ij}(f) = \sum_{n=1}^{\infty} p_{ij}(n) e^{-j2\pi nfT} \tag{4-4-58}$$

and K is the number of states of the modulator. The term $p_{ij}(n)$ denotes the probability that the signal $s_j(t)$ is transmitted n signaling intervals after the transmission of $s_i(t)$. Hence, $\{p_{ij}(n)\}$ are the transition probabilities in the transition probability matrix $\mathbf{P}^n$. Note that $p_{ij}(1) = p_{ij}$.

When there is no memory in the modulation method, the signal waveform transmitted on each signaling interval is independent of the waveforms transmitted in previous signaling intervals. The power density spectrum of the resultant signal may still be expressed in the form of (4-4-57), if the transition probability matrix is replaced by

$$\mathbf{P} = \begin{bmatrix} p_1 & p_2 & \cdots & p_K \\ p_1 & p_2 & \cdots & p_K \\ \vdots & \vdots & & \vdots \\ p_1 & p_2 & \cdots & p_K \end{bmatrix} \tag{4-4-59}$$

and we impose the condition that $\mathbf{P}^n = \boldsymbol{P}$ for all $n \geq 1$. Under these conditions, the expression for the power density spectrum becomes a function of the stationary state probabilities $\{p_i\}$ only, and, hence, it reduces to the simpler form

$$\Phi(f) = \frac{1}{T^2} \sum_{n=-\infty}^{\infty} \left| \sum_{i=1}^{K} p_i S_i\left(\frac{n}{T}\right) \right|^2 \delta\left(f - \frac{n}{T}\right) + \frac{1}{T} \sum_{i=1}^{K} p_i(1-p_i) |S_i(f)|^2 - \frac{2}{T} \sum_{i=1}^{K} \sum_{j=1}^{K} p_i p_j \operatorname{Re}[S_i(f) S_j^*(f)] \quad (4\text{-}4\text{-}60)$$

$(i<j)$

We observe that our previous result for the power density spectrum of memoryless linear modulation given by (4-4-18) may be viewed as a special case of (4-4-60) in which all waveforms are identical except for a set of scale factors that convey the digital information (Problem 4-30).

We also make the observation that the first term in the expression for the power density spectrum given by either (4-4-57) or (4-4-60) consists of discrete frequency components. This line spectrum vanishes when

$$\sum_{i=1}^{K} p_i S_i\left(\frac{n}{T}\right) = 0 \quad (4\text{-}4\text{-}61)$$

The condition (4-4-61) is usually imposed in the design of practical digital communications systems and is easily satisfied by an appropriate choice of signaling waveforms (Problem 4-31).

Now, let us determine the power density spectrum of the baseband-modulated signals described in Section 4-3-2. First, the NRZ signal is characterized by the two waveforms $s_1(t) = g(t)$ and $s_2(t) = -g(t)$, where $g(t)$ is a rectangular pulse of amplitude A. For $K = 2$, (4-4-60) reduces to

$$\Phi(f) = \frac{(2p-1)^2}{T^2} \sum_{n=-\infty}^{\infty} \left| G\left(\frac{n}{T}\right) \right|^2 \delta\left(f - \frac{n}{T}\right) + \frac{4p(1-p)}{T} |G(f)|^2 \quad (4\text{-}4\text{-}62)$$

where

$$|G(f)|^2 = (AT)^2 \left(\frac{\sin \pi f T}{\pi f T}\right)^2 \quad (4\text{-}4\text{-}63)$$

Observe that when $p = \frac{1}{2}$, the line spectrum vanishes and $\Phi(f)$ reduces to

$$\Phi(f) = \frac{1}{T} |G(f)|^2 \quad (4\text{-}4\text{-}64)$$

The NRZI signal is characterized by the transition probability matrix

$$\mathbf{P} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \end{bmatrix} \tag{4-4-65}$$

Notice that in this case $\mathbf{P}^n = \mathbf{P}$ for all $n \geq 1$. Hence, the special form for the power density spectrum given by (4-4-62) applies to this modulation format as well. Consequently, the power density spectrum for the NRZI signal is identical to the spectrum of the NRZ signal.

Delay modulation has a transition probability matrix

$$\mathbf{P} = \begin{bmatrix} 0 & \frac{1}{2} & 0 & \frac{1}{2} \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 \end{bmatrix} \tag{4-4-66}$$

and stationary state probabilities $p_i = \frac{1}{4}$ for $i = 1, 2, 3, 4$. Powers of $\mathbf{P}$ may be obtained by use of the relation

$$\mathbf{P}^4\boldsymbol{\rho} = -\tfrac{1}{4}\boldsymbol{\rho} \tag{4-4-67}$$

where $\boldsymbol{\rho}$ is the signal correlation matrix with elements

$$\rho_{ij} = \frac{1}{T}\int_0^T s_i(t)s_j(t)\,dt \tag{4-4-68}$$

and where the four signals $\{s_i(t),\ i = 1, 2, 3, 4\}$ are shown in Fig. 4-3-15. It is easily seen that

$$\boldsymbol{\rho} = \begin{bmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & -1 & 0 \\ 0 & -1 & 1 & 0 \\ -1 & 0 & 0 & 1 \end{bmatrix} \tag{4-4-69}$$

Consequently, powers of $\mathbf{P}$ can be generated from the relation

$$\mathbf{P}^{k+4}\boldsymbol{\rho} = -\tfrac{1}{4}\mathbf{P}^k\boldsymbol{\rho}, \qquad k > 1 \tag{4-4-70}$$

Use of (4-4-66), (4-4-69), and (4-4-70) in (4-4-57) yields the power density spectrum of delay modulation. It may be expressed in the form

$$\Phi(f) = \frac{1}{2\psi^2(17 + 8\cos 8\psi)}[23 - 2\cos\psi - 22\cos 2\psi - 12\cos 3\psi + 5\cos 4\psi + 12\cos 5\psi + 2\cos 6\psi - 8\cos 7\psi + 2\cos 8\psi] \tag{4-4-71}$$

where $\psi = \pi f T$.

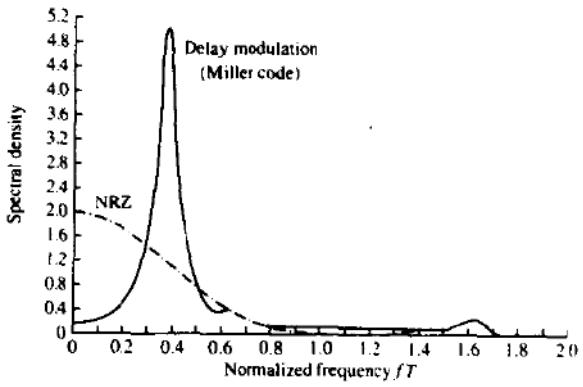


FIGURE 4-4-11 Power spectral density (one-sided) of Miller code (delay modulation) and NRZ/NRZI baseband signals. [*From Hecht and Guida (1969); © 1969 IEEE.*]

The spectra of these baseband signals are illustrated in Fig. 4-4-11. Observe that the spectra of the NRZ and NRZI signals peak at $f = 0$. Delay modulation has a narrower spectrum and a relatively small zero-frequency content. Its bandwidth occupancy is significantly smaller than that of the NRZ signal. These two characteristics make delay modulation an attractive choice for channels that do not pass dc, such as magnetic recording media.

4-5 BIBLIOGRAPHICAL NOTES AND REFERENCES

The characteristics of signals and systems given in this chapter are very useful in the design of optimum modulation/demodulation and coding/decoding techniques for a variety of channel models. In particular, the digital modulation methods introduced in this chapter are widely used in digital communication systems. The next chapter is concerned with optimum demodulation techniques for these signals and their performance in an additive, white gaussian noise channel. A general reference for signal characterization is the book by Franks (1969).

Of particular importance in the design of digital communications systems are the spectral characteristics of the digitally modulated signals, which are presented in this chapter in some depth. Of these modulation techniques, CPM is one of the most important due to its efficient use of bandwidth. For this reason, it has been widely investigated by many researchers, and a large number of papers have been published in the technical literature. The most comprehensive treatment of CPM, including its performance and its spectral characteristics, can be found in the book by Anderson *et al.* (1986). In addition to this text, the tutorial paper by Sundberg (1986) presents the basic concepts and an overview of the performance characteristics of various CPM techniques. This paper also contains over 100 references to published papers on this topic.

There are a large number of references dealing with the spectral characteristics of CPFSK and CPM. As a point of reference, we should mention that MSK was invented by Doelz and Heald in 1961. The early work on the power

spectral density of CPFSK and CPM was done by Bennett and Rice (1963), Anderson and Salz (1965), and Bennett and Davey (1965). The book by Lucky *et al.* (1968) also contains a treatment of the spectral characteristics of CPFSK. Most of the recent work is referenced in the paper by Sundberg (1986). We should also cite the special issue on bandwidth-efficient modulation and coding published by the *IEEE Transactions on Communications* (March 1981), which contains several papers on the spectral characteristics and performance of CPM.

The generalization of MSK to multiple amplitudes was investigated by Weber *et al.* (1978). The combination of multiple amplitudes with general CPM was proposed by Mulligan (1988) who investigated its spectral characteristics and its error probability performance in gaussian noise with and without coding.

4-1 Prove the following properties of Hilbert transforms:
- **a** If $x(t) = x(-t)$ then $\hat{x}(t) = -\hat{x}(-t)$;
- **b** If $x(t) = -x(-t)$ then $\hat{x}(t) = \hat{x}(-t)$;
- **c** If $x(t) = \cos \omega_0 t$ then $\hat{x}(t) = \sin \omega_0 t$;
- **d** If $x(t) = \sin \omega_0 t$ then $\hat{x}(t) = -\cos \omega_0 t$;
- **e** $\hat{\hat{x}}(t) = -x(t)$;
- **f** $\int_{-\infty}^{\infty} x^2(t)\,dt = \int_{-\infty}^{\infty} \hat{x}^2(t)\,dt$;
- **g** $\int_{-\infty}^{\infty} x(t)\hat{x}(t)\,dt = 0$.

4-2 If $x(t)$ is a stationary random process with autocorrelation function $\phi_{xx}(\tau) = E[x(t)x(t+\tau)]$ and spectral density $\Phi_{xx}(f)$ then show that $\phi_{\hat{x}\hat{x}}(\tau) = \phi_{xx}(\tau)$, $\phi_{x\hat{x}}(\tau) = -\hat{\phi}_{xx}(\tau)$, and $\Phi_{\hat{x}\hat{x}}(f) = \Phi_{xx}(f)$.

4-3 Suppose that $n(t)$ is a zero-mean stationary narrowband process represented by either (4-1-37), (4-1-38), or (4-1-39). The autocorrelation function of the equivalent lowpass process $z(t) = x(t) + jy(t)$ is defined as

$$\phi_{zz}(\tau) = \tfrac{1}{2}E[z^*(t)z(t+\tau)]$$

- **a** Show that

$$E[z(t)z(t+\tau)] = 0$$

- **b** Suppose $\phi_{zz}(\tau) = N_0\delta(\tau)$, and let

$$V = \int_0^t z(t)\,dt$$

Determine $E(V^2)$ and $E(VV^*) = E(|V|^2)$.

4-4 Determine the autocorrelation function of the stochastic process

$$x(t) = A \sin(2\pi f_c t + \theta)$$

where f_c is a constant and θ is a uniformly distributed phase, i.e.,

$$p(\theta) = \frac{1}{2\pi}, \quad 0 \le \theta \le 2\pi$$

4-5 Prove that $s_l(t)$ is generally a complex-valued signal and give the condition under which it is real. Assume that $s(t)$ is a real-valued bandpass signal.

4-6 Suppose that $s(t)$ is either a real- or complex-valued signal that is represented as a linear combination of orthonormal functions $\{f_n(t)\}$, i.e.,

$$\hat{s}(t) = \sum_{k=1}^{K} s_k f_k(t)$$

where

$$\int_{-\infty}^{\infty} f_n(t) f_m^*(t)\, dt = \begin{cases} 0 & (m \neq n) \\ 1 & (m = n) \end{cases}$$

Determine the expressions for the coefficients $\{s_k\}$ in the expansion $\hat{s}(t)$ that minimize the energy

$$\mathscr{E}_e = \int_{-\infty}^{\infty} |s(t) - \hat{s}(t)|^2\, dt$$

and the corresponding residual error $\mathscr{E}_e$.

4-7 Suppose that a set of M signal waveforms $\{s_m(t)\}$ are complex-valued. Derive the equations for the Gram–Schmidt procedure that will result in a set of $N \leq M$ orthonormal signal waveforms.

4-8 Determine the correlation coefficients ρ_{km} among the four signal waveforms $\{s_i(t)\}$ shown in Fig. 4-2-1, and the corresponding Euclidean distances.

4-9 Consider a set of M orthogonal signal waveforms $s_m(t)$, $1 \leq m \leq M$, $0 \leq t \leq T$, all of which have the same energy $\mathscr{E}$. Define a new set of M waveforms as

$$s'_m(t) = s_m(t) - \frac{1}{M} \sum_{k=1}^{M} s_k(t), \qquad 1 \leq m \leq M, \quad 0 \leq t \leq T$$

Show that the M signal waveforms $\{s'_m(t)\}$ have equal energy, given by

$$\mathscr{E}' = (M-1)\mathscr{E}/M$$

and are equally correlated, with correlation coefficient

$$\rho_{mn} = \frac{1}{\mathscr{E}'} \int_0^T s'_m(t) s'_n(t)\, dt = -\frac{1}{M-1}$$

4-10 Consider the three waveforms $f_n(t)$ shown in Fig. P4-10.

a Show that these waveforms are orthonormal.

b Express the waveform $x(t)$ as a weighted linear combination of $f_n(t)$, $n = 1, 2, 3$, if

$$x(t) = \begin{cases} -1 & (0 \leq t < 1) \\ 1 & (1 \leq t < 3) \\ -1 & (3 \leq t < 4) \end{cases}$$

and determine the weighting coefficients.

4-11 Consider the four waveforms shown in Fig. P4-11.

a Determine the dimensionality of the waveforms and a set of basis functions.

b Use the basis functions to represent the four waveforms by vectors $\mathbf{s}_1$, $\mathbf{s}_2$, $\mathbf{s}_3$, and $\mathbf{s}_4$.

c Determine the minimum distance between any pair of vectors.

4-12 Determine a set of orthonormal functions for the four signals shown in Fig. P4-12.

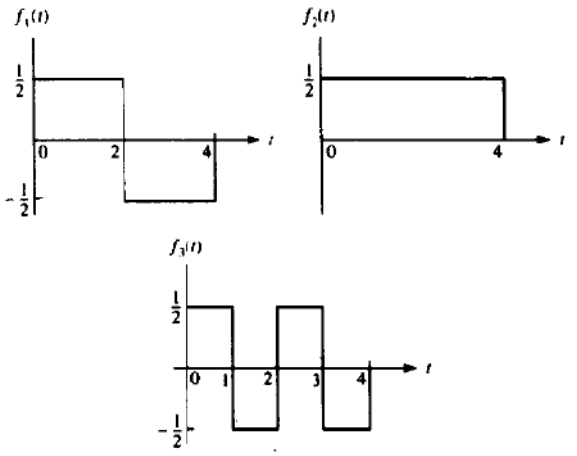


FIGURE P4-10

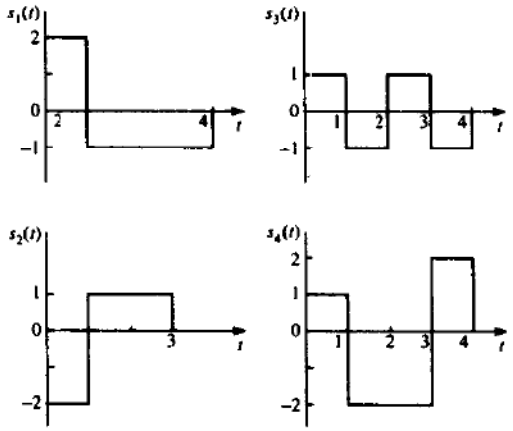


FIGURE P4-11

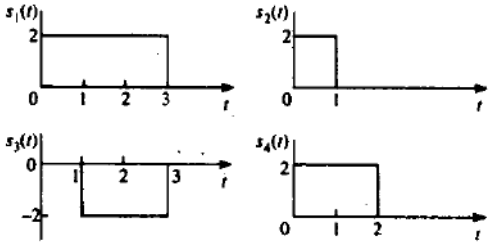


FIGURE P4-12

4-13 A lowpass gaussian stochastic process $x(t)$ has a power spectral density

$$\Phi(f) = \begin{cases} N_0 & (|f| < B) \\ 0 & (|f| > B) \end{cases}$$

Determine the power spectral density and the autocorrelation function of $y(t) = x^2(t)$.

4-14 Consider an equivalent lowpass digitally modulated signal of the form

$$u(t) = \sum_n [a_n g(t - 2nT) - jb_n g(t - 2nT - T)]$$

where $\{a_n\}$ and $\{b_n\}$ are two sequences of statistically independent binary digits and $g(t)$ is a sinusoidal pulse defined as

$$g(t) = \begin{cases} \sin(\pi t/2T) & (0 < t < 2T) \\ 0 & (\text{otherwise}) \end{cases}$$

This type of signal is viewed as a four-phase PSK signal in which the pulse shape is one-half cycle of a sinusoid. Each of the information sequences $\{a_n\}$ and $\{b_n\}$ is transmitted at a rate of $1/2T$ bits/s and, hence, the combined transmission rate is $1/T$ bits/s. The two sequences are staggered in time by T seconds in transmission. Consequently, the signal $u(t)$ is called *staggered four-phase PSK*.

a Show that the envelope $|u(t)|$ is a constant, independent of the information a_n on the in-phase component and information b_n on the quadrature component. In other words, the amplitude of the carrier used in transmitting the signal is constant.

b Determine the power density spectrum of $u(t)$.

c Compare the power density spectrum obtained from (b) with the power density spectrum of the MSK signal. What conclusion can you draw from this comparison?

4-15 Consider a four-phase PSK signal represented by the equivalent lowpass signal

$$u(t) = \sum_n I_n g(t - nT)$$

where I_n takes on one of the four possible values $\sqrt{\frac{1}{2}}(\pm 1 \pm j)$ with equal probability. The sequence of information symbols $\{I_n\}$ is statistically independent.

a Determine and sketch the power density spectrum of $u(t)$ when

$$g(t) = \begin{cases} A & (0 \leq t \leq T) \\ 0 & (\text{otherwise}) \end{cases}$$

b Repeat (a) when

$$g(t) = \begin{cases} A \sin(\pi t/T) & (0 \leq t \leq T) \\ 0 & (\text{otherwise}) \end{cases}$$

c Compare the spectra obtained in (a) and (b) in terms of the 3 dB bandwidth and the bandwidth to the first spectral zero.

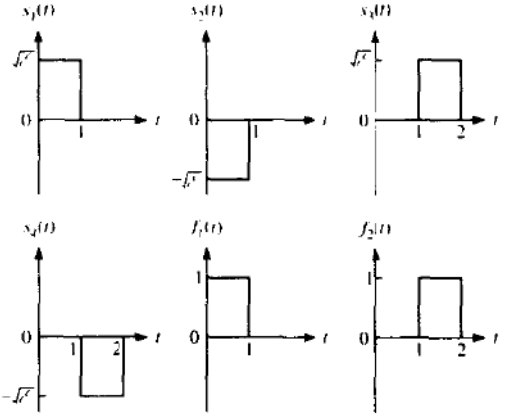


FIGURE P4-18

4-16 The random process $v(t)$ is defined as

$$v(t) = X \cos 2\pi f_c t - Y \sin 2\pi f_c t$$

where X and Y are random variables. Show that $v(t)$ is wide-sense stationary if and only if $E(X) = E(Y) = 0$, $E(X^2) = E(Y^2)$, and $E(XY) = 0$.

4-17 Carry out the Gram–Schmidt orthogonalization of the signals in Fig. 4-2-1(*a*) in the order $s_4(t)$, $s_3(t)$, $s_1(t)$, and, thus, obtain a set of orthonormal functions $\{f_m(t)\}$. Then, determine the vector representation of the signals $\{s_n(t)\}$ by using the orthonormal functions $\{f_m(t)\}$. Also, determine the signal energies.

4-18 Determine the signal space representation of the four signals $s_k(t)$, $k = 1, 2, 3, 4$, shown in Fig. P4-18, by using as basis functions the orthonormal functions $f_1(t)$ and $f_2(t)$. Plot the signal space diagram and show that this signal set is equivalent to that for a four-phase PSK signal.

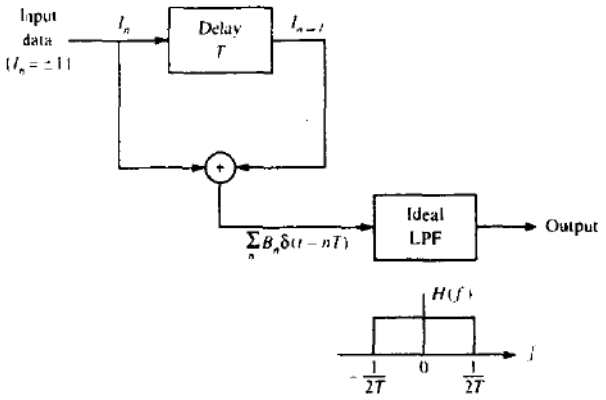
4-19 The power density spectrum of the cyclostationary process

$$v(t) = \sum_{n=-\infty}^{\infty} I_n g(t - nT)$$

was derived in Section 4-4-1 by averaging the autocorrelation function $\phi_{vv}(t + \tau, t)$ over the period T of the process and then evaluating the Fourier transform of the average autocorrelation function. An alternative approach is to change the cyclostationary process into a stationary process $v_\Delta(t)$ by adding a random variable Δ, uniformly distributed over $0 \leq \Delta < T$, so that

$$v_\Delta(t) = \sum_{n=-\infty}^{\infty} I_n g(t - nT - \Delta)$$

and defining the spectral density of $v(t)$ as the Fourier transform of the autocorrelation function of the stationary process $v_\Delta(t)$. Derive the result in (4-4-11), by evaluating the autocorrelation function of $v_\Delta(t)$ and its Fourier transform.



4-20 A PAM partial response signal (PRS) is generated as shown in Fig. P4-20 by exciting an ideal lowpass filter of bandwidth W by the sequence

$$B_n = I_n + I_{n-1}$$

at a rate $1/T = 2W$ symbols/s. The sequence $\{I_n\}$ consists of binary digits selected independently from the alphabet $\{1, -1\}$ with equal probability. Hence, the filtered signal has the form

$$v(t) = \sum_{n=-\infty}^{\infty} B_n g(t - nT), \qquad T = \frac{1}{2W}$$

a Sketch the signal space diagram for $v(t)$ and determine the probability of occurrence of each symbol.
b Determine the autocorrelation and power density spectrum of the three-level sequence $\{B_n\}$.
c The signal points of the sequence $\{B_n\}$ form a Markov chain. Sketch this Markov chain and indicate the transition probabilities among the states.

4-21 The lowpass equivalent representation of a PAM signal is

$$u(t) = \sum_n I_n g(t - nT)$$

Suppose $g(t)$ is a rectangular pulse and

$$I_n = a_n - a_{n-2}$$

where $\{a_n\}$ is a sequence of uncorrelated binary-valued $(1, -1)$ random variables that occur with equal probability.
a Determine the autocorrelation function of the sequence $\{I_n\}$.
b Determine the power density spectrum of $u(t)$.
c Repeat (b) if the possible values of the a_n are $(0, 1)$.

4-22 Show that $x(t) = s(t)\cos 2\pi f_c t \pm \hat{s}(t)\sin 2\pi f_c t$ is a single-sideband signal, where $s(t)$ is band-limited to $B \leq f_c$ Hz and $\hat{s}(t)$ is its Hilbert transform.

4-23 Use the results in Section 4-4-3 to determine the power density spectrum of the binary FSK signals in which the waveforms are

$$s_i(t) = \sin \omega_i t, \quad i = 1, 2, \quad 0 \leq t \leq T$$

where $\omega_1 = n\pi/T$ and $\omega_2 = m\pi/T$, $n \neq m$, and m and n are arbitrary positive integers. Assume that $p_1 = p_2 = \frac{1}{2}$. Sketch the spectrum and compare this result with the spectrum of the MSK signal.

4-24 Use the results in Section 4-4-3 to determine the power density spectrum of multitone FSK (MFSK) signals for which the signal waveforms are

$$s_n(t) = \sin \frac{2\pi n t}{T}, \quad n = 1, 2, \ldots, M, \quad 0 \leq t \leq T$$

Assume that the probabilities $p_i = 1/M$ for all i. Sketch the power spectral density.

4-25 A quadrature partial response signal (QPRS) is generated by two separate partial response signals of the type described in Problem 4-20 placed in phase quadrature. Hence, the QPRS is represented as

$$s(t) = \text{Re}\,[v(t)e^{j2\pi f_c t}]$$

where

$$v(t) = v_c(t) + jv_s(t)$$
$$= \sum_n B_n u(t - nT) + j \sum_n C_n u(t - nT)$$

and $B_n = I_n + I_{n-1}$ and $C_n = J_n + J_{n-1}$. The sequences $\{B_n\}$ and $\{C_n\}$ are uncorrelated and $I_n = \pm 1$, $J_n = \pm 1$ with equal probability.

a Sketch the signal space diagram for the QPRS signal and determine the probability of occurrence of each symbol.

b Determine the autocorrelations and power spectra density of $v_c(t)$, $v_s(t)$, and $v(t)$.

c Sketch the Markov chain model and indicate the transition probabilities for the QPRS.

4-26 Determine the autocorrelation functions for the MSK and offset QPSK modulated signals based on the assumption that the information sequences for each of the two signals are uncorrelated and zero-mean.

4-27 Sketch the phase tree, the state trellis, and the state diagram for partial response CPM with $h = \frac{1}{2}$ and

$$u(t) = \begin{cases} 1/4T & (0 \leq t \leq 2T) \\ 0 & (\text{otherwise}) \end{cases}$$

4-28 Determine the number of terminal phase states in the state trellis diagram for

a a full response binary CPFSK with either $h = \frac{2}{3}$ or $\frac{3}{4}$;

b a partial response $L = 3$ binary CPFSK with either $h = \frac{2}{3}$ or $\frac{3}{4}$.

4-29 Show that 16 QAM can be represented as a superposition of two four-phase constant envelope signals where each component is amplified separately before summing, i.e.

$$s(t) = G[A_n \cos 2\pi f_c t + B_n \sin 2\pi f_c t] + [C_n \cos 2\pi f_c t + D_n \sin 2\pi f_c t]$$

where $\{A_n\}$, $\{B_n\}$, $\{C_n\}$, and $\{D_n\}$ are statistically independent binary sequences

with elements from the set $\{+1, -1\}$ and G is the amplifier gain. Thus, show that the resulting signal is equivalent to

$$s(t) = I_n \cos 2\pi f_c t + Q_n \sin 2\pi f_c t$$

and determine I_n and Q_n in terms of A_n, B_n, C_n, and D_n.

4-30 Use the result in (4-4-60) to derive the expression for the power density spectrum of memoryless linear modulation given by (4-4-18) under the condition that

$$s_k(t) = I_k s(t), \quad k = 1, 2, \ldots, K$$

where I_k is one of the K possible transmitted symbols that occur with equal probability.

4-31 Show that a sufficient condition for the absence of the line spectrum component in (4-4-60) is

$$\sum_{i=1}^{K} p_i s_i(t) = 0$$

Is this condition necessary? Justify your answer.

4-32 The information sequence $\{a_n\}_{n=-\infty}^{\infty}$ is a sequence of iid random variables, each taking values +1 and −1 with equal probability. This sequence is to be transmitted at baseband by a biphase coding scheme, described by

$$s(t) = \sum_{n=-\infty}^{\infty} a_n g(t - nT)$$

where $g(t)$ is shown in Fig. P4-32.

a Find the power spectral density of $s(t)$.

b Assume that it is desirable to have a zero in the power spectrum at $f = 1/T$. To this end, we use a precoding scheme by introducing $b_n = a_n + ka_{n-1}$, where k is some constant, and then transmit the $\{b_n\}$ sequence using the same $g(t)$. Is it possible to choose k to produce a frequency null at $f = 1/T$? If yes, what are the appropriate value and the resulting power spectrum?

c Now assume we want to have zeros at all multiples of $f_0 = 1/4T$. Is it possible to have these zeros with an appropriate choice of k in the previous part? If not then what kind of precoding do you suggest to result in the desired nulls?

4-33 Starting with the definition of the transition probability matrix for delay modulation given in (4-4-66), demonstrate that the relation

$$\mathbf{P}^4\boldsymbol{\rho} = -\tfrac{1}{4}\boldsymbol{\rho}$$

holds, and, hence,

$$\mathbf{P}^{k+4}\boldsymbol{\rho} = -\tfrac{1}{4}\mathbf{P}^k\boldsymbol{\rho}, \quad k \geq 1$$

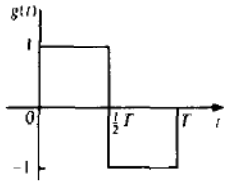


FIGURE P4-32

4-34 The two signal waveforms for binary FSK signal transmission with discontinuous phase are

$$s_0(t) = \sqrt{\frac{2\mathscr{E}_b}{T_b}} \cos\left[2\pi\left(f - \frac{\Delta f}{2}\right)t + \theta_0\right], \quad 0 \leq t < T$$

$$s_1(t) = \sqrt{\frac{2\mathscr{E}_b}{T_b}} \cos\left[2\pi\left(f + \frac{\Delta f}{2}\right)t + \theta_1\right], \quad 0 \leq t \leq T$$

where $\Delta f = 1/T \ll f_c$, and θ_0 and θ_1 are uniformly distributed random variables on the interval $(0, 2\pi)$. The signals $s_0(t)$ and $s_1(t)$ are equally probable.

a Determine the power spectral density of the FSK signal.

b Show that the power spectral density decays as $1/f^2$ for $f \gg f_c$.

5

OPTIMUM RECEIVERS FOR THE ADDITIVE WHITE GAUSSIAN NOISE CHANNEL

In Chapter 4, we described various types of modulation methods that may be used to transmit digital information through a communication channel. As we have observed, the modulator at the transmitter performs the function of mapping the digital sequence into signal waveforms.

This chapter deals with the design and performance characteristics of optimum receivers for the various modulation methods, when the channel corrupts the transmitted signal by the addition of gaussian noise. In Section 5-1, we first treat memoryless modulation signals, followed by modulation signals with memory. We evaluate the probability of error of the various modulation methods in Section 5-2. We treat the optimum receiver for CPM signals and its performance in Section 5-3. In Section 5-4, we derive the optimum receiver when the carrier phase of the signals is unknown at the receiver and is treated as a random variable. Finally, in Section 5-5, we consider the use of regenerative repeaters in signal transmission and carry out a link budget analysis for radio channels.

5-1 OPTIMUM RECEIVER FOR SIGNALS CORRUPTED BY ADDITIVE WHITE GAUSSIAN NOISE

Let us begin by developing a mathematical model for the signal at the input to the receiver. We assume that the transmitter sends digital information by use of M signal waveforms $\{s_m(t), m = 1, 2, \ldots, M\}$. Each waveform is transmitted within the symbol (signaling) interval of duration T. To be specific, we consider the transmission of information over the interval $0 \leq t \leq T$.

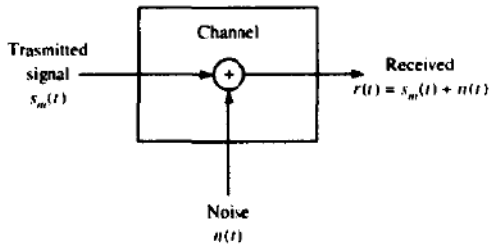


FIGURE 5-1-1 Model for received signal passed through an AWGN channel.

The channel is assumed to corrupt the signal by the addition of white gaussian noise, as illustrated in Fig. 5-1-1. Thus, the received signal in the interval $0 \leq t \leq T$ may be expressed as

$$r(t) = s_m(t) + n(t), \qquad 0 \leq t \leq T \tag{5-1-1}$$

where $n(t)$ denotes a sample function of the additive white gaussian noise (AWGN) process with power spectral density $\Phi_{nn}(f) = \frac{1}{2}N_0$ W/Hz. Based on the observation of $r(t)$ over the signal interval, we wish to design a receiver that is optimum in the sense that it minimizes the probability of making an error.

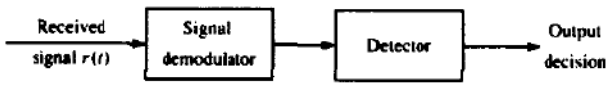
It is convenient to subdivide the receiver into two parts—the signal demodulator and the detector—as shown in Fig. 5-1-2. The function of the signal demodulator is to convert the received waveform $r(t)$ into an N-dimensional vector $\mathbf{r} = [r_1 \; r_2 \; \cdots \; r_N]$, where N is the dimension of the transmitted signal waveforms. The function of the detector is to decide which of the M possible signal waveforms was transmitted based of the vector $\mathbf{r}$.

Two realizations of the signal demodulator are described in the next two sections. One is based on the use of signal correlators. The second is based on the use of matched filters. The optimum detector that follows the signal demodulator is designed to minimize the probability of error.

5-1-1 Correlation Demodulator

In this section, we describe a correlation demodulator that decomposes the received signal and the noise into N-dimensional vectors. In other words, the signal and the noise are expanded into a series of linearly weighted orthonormal basis functions $\{f_n(t)\}$. It is assumed that the N basis functions $\{f_n(t)\}$ span the signal space, so that every one of the possible transmitted

FIGURE 5-1-2 Receiver configuration.



signals of the set $\{s_m(t), 1 \leq m \leq M\}$ can be represented as a weighted linear combination of $\{f_n(t)\}$. In the case of the noise, the functions $\{f_n(t)\}$ do not span the noise space. However, we show below that the noise terms that fall outside the signal space are irrelevant to the detection of the signal.

Suppose the received signal $r(t)$ is passed through a parallel bank of N crosscorrelators which basically compute the projection of $r(t)$ onto the N basis functions $\{f_n(t)\}$, as illustrated in Fig. 5-1-3. Thus, we have

$$\int_0^T r(t)f_k(t)\,dt = \int_0^T [s_m(t) + n(t)]f_k(t)\,dt$$
$$r_k = s_{mk} + n_k, \quad k = 1, 2, \ldots, N \tag{5-1-2}$$

where

$$s_{mk} = \int_0^T s_m(t)f_k(t)\,dt, \quad k = 1, 2, \ldots, N$$
$$n_k = \int_0^T n(t)f_k(t)\,dt, \quad k = 1, 2, \ldots, N \tag{5-1-3}$$

The signal is now represented by the vector $\mathbf{s}_m$ with components s_{mk}, $k = 1, 2, \ldots, N$. Their values depend on which of the M signals was transmitted. The components $\{n_k\}$ are random variables that arise from the presence of the additive noise.

In fact, we can express the received signal $r(t)$ in the interval $0 \leq t \leq T$ as

$$r(t) = \sum_{k=1}^{N} s_{mk}f_k(t) + \sum_{k=1}^{N} n_k f_k(t) + n'(t)$$
$$= \sum_{k=1}^{N} r_k f_k(t) + n'(t) \tag{5-1-4}$$

The term $n'(t)$, defined as

$$n'(t) = n(t) - \sum_{k=1}^{N} n_k f_k(t) \tag{5-1-5}$$

is a zero-mean gaussian noise process that represents the difference between the original noise process $n(t)$ and the part corresponding to the projection of $n(t)$ onto the basis functions $\{f_k(t)\}$. We shall show below that $n'(t)$ is irrelevant to the decision as to which signal was transmitted. Consequently, the decision may be based entirely on the correlator output signal and noise components $r_k = s_{mk} + n_k$, $k = 1, 2, \ldots, N$.

Since the signals $\{s_m(t)\}$ are deterministic, the signal components are deterministic. The noise components $\{n_k\}$ are gaussian. Their mean values are

$$E(n_k) = \int_0^T E[n(t)]f_k(t)\,dt = 0 \tag{5-1-6}$$

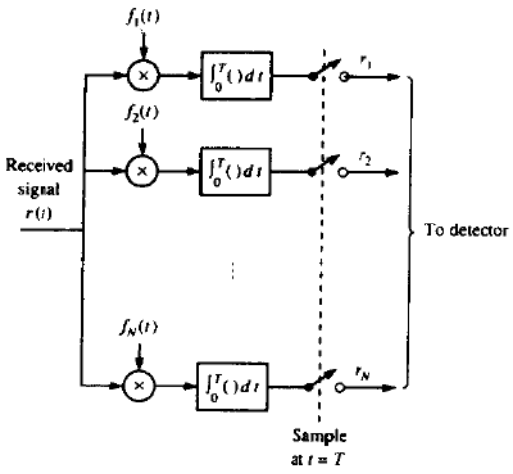


FIGURE 5-1-3 Correlation-type demodulator.

for all n. Their covariances are

$$\begin{aligned} E(n_k n_m) &= \int_0^T \int_0^T E[n(t)n(\tau)] f_k(t) f_m(\tau)\, dt\, d\tau \\ &= \tfrac{1}{2}N_0 \int_0^T \int_0^T \delta(t-\tau) f_k(t) f_m(\tau)\, dt\, d\tau \\ &= \tfrac{1}{2}N_0 \int_0^T f_k(t) f_m(t)\, dt \\ &= \tfrac{1}{2}N_0 \delta_{mk} \end{aligned} \tag{5-1-7}$$

where $\delta_{mk} = 1$ when $m = k$ and zero otherwise. Therefore, the N noise components $\{n_k\}$ are zero-mean uncorrelated gaussian random variables with a common variance $\sigma_n^2 = \frac{1}{2}N_0$.

From the above development, it follows that the correlator outputs $\{r_k\}$ conditioned on the mth signal being transmitted are gaussian random variables with mean

$$E(r_k) = E(s_{mk} + n_k) = s_{mk} \tag{5-1-8}$$

and equal variance

$$\sigma_r^2 = \sigma_n^2 = \tfrac{1}{2}N_0 \tag{5-1-9}$$

Since the noise components $\{n_k\}$ are uncorrelated gaussian random variables, they are also statistically independent. As a consequence, the correlator outputs $\{r_k\}$ conditioned on the mth signal being transmitted are statistically independent gaussian variables. Hence, the conditional probability density functions of the random variables $[r_1\ r_2 \cdots r_N] = \mathbf{r}$ are simply

$$p(\mathbf{r} \mid \mathbf{s}_m) = \prod_{k=1}^{N} p(r_k \mid s_{mk}), \qquad m = 1, 2, \ldots, M \tag{5-1-10}$$

where

$$p(r_k \mid s_{mk}) = \frac{1}{\sqrt{\pi N_0}} \exp\left[-\frac{(r_k - s_{mk})^2}{N_0} \right], \quad k = 1, 2, \ldots, N \qquad (5\text{-}1\text{-}11)$$

By substituting (5-1-11) into (5-1-10), we obtain the joint conditional pdfs

$$p(\mathbf{r} \mid \mathbf{s}_m) = \frac{1}{(\pi N_0)^{N/2}} \exp\left[-\sum_{k=1}^{N} \frac{(r_k - s_{mk})^2}{N_0} \right], \quad m = 1, 2, \ldots, M \qquad (5\text{-}1\text{-}12)$$

As a final point we wish to show that the correlator outputs $(r_1, r_2, \ldots, r_N)$ are *sufficient statistics* for reaching a decision on which of the M signals was transmitted, i.e., that no additional relevant information can be extracted from the remaining noise process $n'(t)$. Indeed, $n'(t)$ is uncorrelated with the N correlator outputs $\{r_k\}$, i.e.,

$$\begin{aligned} E[n'(t)r_k] &= E[n'(t)]s_{mk} + E[n'(t)n_k] \\ &= E[n'(t)n_k] \\ &= E\left\{ \left[n(t) - \sum_{j=1}^{N} n_j f_j(t) \right] n_k \right\} \\ &= \int_0^T E[n(t)n(\tau)] f_k(\tau)\, d\tau - \sum_{j=1}^{N} E(n_j n_k) f_j(t) \\ &= \tfrac{1}{2} N_0 f_k(t) - \tfrac{1}{2} N_0 f_k(t) = 0 \end{aligned} \qquad (5\text{-}1\text{-}13)$$

Since $n'(t)$ and $\{r_k\}$ are gaussian and uncorrelated, they are also statistically independent. Consequently, $n'(t)$ does not contain any information that is relevant to the decision as to which signal waveform was transmitted. All the relevant information is contained in the correlator outputs $\{r_k\}$. Hence, $n'(t)$ may be ignored.

Example 5-1-1

Consider an M-ary baseband PAM signal set in which the basic pulse shape $g(t)$ is rectangular as shown in Fig. 5-1-4. The additive noise is a zero-mean white gaussian noise process. Let us determine the basis function $f(t)$ and the output of the correlation-type demodulator. The energy in the rectangular pulse is

$$\mathcal{E}_g = \int_0^T g^2(t)\, dt = \int_0^T a^2\, dt = a^2 T$$

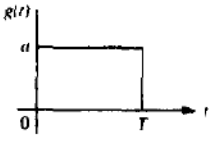


FIGURE 5-1-4 Signal pulse for Example 5-1-1.

Since the PAM signal set has dimension $N = 1$, there is only one basis function $f(t)$. This is given as

$$f(t) = \frac{1}{\sqrt{a^2 T}} g(t)$$

$$= \begin{cases} 1/\sqrt{T} & (0 \leq t \leq T) \\ 0 & (\text{otherwise}) \end{cases}$$

The output of the correlation-type demodulator is

$$r = \int_0^T r(t) f(t)\, dt = \frac{1}{\sqrt{T}} \int_0^T r(t)\, dt$$

It is interesting to note that the correlator becomes a simple integrator when $f(t)$ is rectangular. If we substitute for $r(t)$, we obtain

$$r = \frac{1}{\sqrt{T}} \left\{ \int_0^T [s_m(t) + n(t)] \right\} dt$$

$$= \frac{1}{\sqrt{T}} \left[\int_0^T s_m(t)\, dt + \int_0^T n(t)\, dt \right]$$

$$r = s_m + n$$

where the noise term $E(n) = 0$ and

$$\sigma_n^2 = E\left[\frac{1}{T} \int_0^T \int_0^T n(t) n(\tau)\, dt\, d\tau \right]$$

$$= \frac{1}{T} \int_0^T \int_0^T E[n(t) n(\tau)]\, dt\, d\tau$$

$$= \frac{N_0}{2T} \int_0^T \int_0^T \delta(t - \tau)\, dt\, d\tau = \tfrac{1}{2} N_0$$

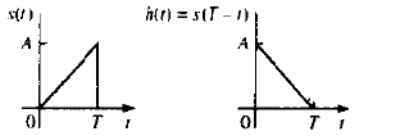
The probability density function for the sampled output is

$$p(r \mid s_m) = \frac{1}{\sqrt{\pi N_0}} \exp\left[-\frac{(r - s_m)^2}{N_0} \right]$$

5-1-2 Matched-Filter Demodulator

Instead of using a bank of N correlators to generate the variables $\{r_k\}$, we may use a bank of N linear filters. To be specific, let us suppose that the impulse responses of the N filters are

$$h_k(t) = f_k(T - t), \quad 0 \leq t \leq T \tag{5-1-14}$$



(a) Signal $s(t)$ (b) Impulse response of filter matched to $s(t)$

FIGURE 5-1-5 Signal $s(t)$ and filter matched to $s(t)$.

where $\{f_k(t)\}$ are the N basis functions and $h_k(t) = 0$ outside of the interval $0 \leq t \leq T$. The outputs of these filters are

$$y_k(t) = \int_0^t r(\tau)h_k(t-\tau)\,d\tau$$

$$= \int_0^t r(\tau)f_k(T-t+\tau)\,d\tau, \qquad k = 1, 2, \ldots, N \tag{5-1-15}$$

Now, if we sample the outputs of the filters at $t = T$, we obtain

$$y_k(T) = \int_0^T r(\tau)f_k(\tau)\,d\tau = r_k, \qquad k = 1, 2, \ldots, N \tag{5-1-16}$$

Hence, the sampled outputs of the filters at time $t = T$ are exactly the set of values $\{r_k\}$ obtained from the N linear correlators.

A filter whose impulse response $h(t) = s(T-t)$, where $s(t)$ is assumed to be confined to the time interval $0 \leq t \leq T$, is called the *matched filter* to the signal $s(t)$. An example of a signal and its matched filter are shown in Fig. 5-1-5. The response of $h(t) = s(T-t)$ to the signal $s(t)$ is

$$y(t) = \int_0^t s(\tau)s(T-t+\tau)\,d\tau \tag{5-1-17}$$

which is basically the time-autocorrelation function of the signal $s(t)$. Figure 5-1-6 illustrates $y(t)$ for the triangular signal pulse shown in Fig. 5-1-5. Note that the autocorrelation function $y(t)$ is an even function of t, which attains a peak at $t = T$.

In the case of the demodulator described above, the N matched filters are

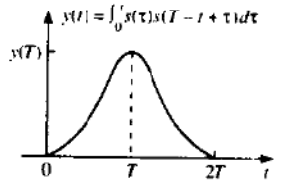


FIGURE 5-1-6 The matched filter output is the autocorrelation function of $s(t)$.

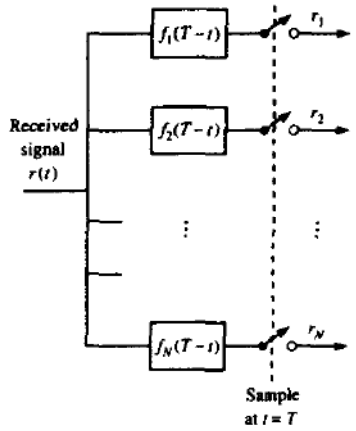


FIGURE 5-1-7 Matched filter demodulator.

matched to the basis functions $\{f_k(t)\}$. Figure 5-1-7 illustrates the matched filter demodulator that generates the observed variables $\{r_k\}$.

Properties of the Matched Filter A matched filter has some interesting properties. Let us prove the most important property, which may be stated as follows: If a signal $s(t)$ is corrupted by AWGN, the filter with impulse response matched to $s(t)$ maximizes the output signal-to-noise ratio (SNR).

To prove this property, let us assume that the received signal $r(t)$ consists of the signal $s(t)$ and AWGN $n(t)$ which has zero-mean and power spectral density $\Phi_{nn}(f) = \frac{1}{2}N_0$ W/Hz. Suppose the signal $r(t)$ is passed through a filter with impulse response $h(t)$, $0 \leq t \leq T$, and its output is sampled at time $t = T$. The filter response to the signal and noise components is

$$\begin{aligned} y(t) &= \int_0^t r(\tau)h(t-\tau)\,d\tau \\ &= \int_0^t s(\tau)h(t-\tau)\,d\tau + \int_0^t n(\tau)h(t-\tau)\,d\tau \end{aligned} \tag{5-1-18}$$

At the sampling instant $t = T$, the signal and noise components are

$$\begin{aligned} y(T) &= \int_0^T s(\tau)h(T-\tau)\,d\tau + \int_0^T n(\tau)h(t-\tau)\,d\tau \\ &= y_s(T) + y_n(T) \end{aligned} \tag{5-1-19}$$

where $y_s(T)$ represents the signal component and $y_n(T)$ the noise component. The problem is to select the filter impulse response that maximizes the output signal-to-noise ratio (SNR_0) defined as

$$\text{SNR}_0 = \frac{y_s^2(T)}{E[y_n^2(T)]} \tag{5-1-20}$$

The denominator in (5-1-20) is simply the variance of the noise term at the output of the filter. Let us evaluate $E[y_n^2(T)]$. We have

$$\begin{aligned} E[y_n^2(T)] &= \int_0^T \int_0^T E[n(\tau)n(t)]h(T-\tau)h(T-t)\,dt\,d\tau \\ &= \tfrac{1}{2}N_0 \int_0^T \int_0^T \delta(t-\tau)h(T-\tau)h(T-t)\,dt\,d\tau \\ &= \tfrac{1}{2}N_0 \int_0^T h^2(T-t)\,dt \end{aligned} \tag{5-1-21}$$

Note that the variance depends on the power spectral density of the noise and the energy in the impulse response $h(t)$.

By substituting for $y_s(T)$ and $E[y_n^2(T)]$ into (5-1-20), we obtain the expression for the output SNR as

$$\mathrm{SNR}_0 = \frac{[\int_0^T s(\tau)h(T-\tau)\,d\tau]^2}{\frac{1}{2}N_0 \int_0^T h^2(T-t)\,dt} = \frac{[\int_0^T h(\tau)s(T-\tau)\,d\tau]^2}{\frac{1}{2}N_0 \int_0^T h^2(T-t)\,dt} \tag{5-1-22}$$

Since the denominator of the SNR depends on the energy in $h(t)$, the maximum output SNR over $h(t)$ is obtained by maximizing the numerator subject to the constraint that the demoninator is held constant. The maximization of the numerator is most easily performed by use of the Cauchy–Schwarz inequality, which states, in general, that if $g_1(t)$ and $g_2(t)$ are finite-energy signals then

$$\left[\int_{-\infty}^{\infty} g_1(t)g_2(t)\,dt\right]^2 \leq \int_{-\infty}^{\infty} g_1^2(t)\,dt \int_{-\infty}^{\infty} g_2^2(t)\,dt \tag{5-1-23}$$

with equality when $g_1(t) = Cg_2(t)$ for any arbitrary constant C. If we set $g_1(t) = h(t)$ and $g_2(t) = s(T-t)$, it is clear that the SNR is maximized when $h(t) = Cs(T-t)$, i.e., $h(t)$ is matched to the signal $s(t)$. The scale factor C^2 drops out of the expression for the SNR, since it appears in both the numerator and the denominator.

The output (maximum) SNR obtained with the matched filter is

$$\begin{aligned} \mathrm{SNR}_0 &= \frac{2}{N_0}\int_0^T s^2(t)\,dt \\ &= 2\mathscr{E}/N_0 \end{aligned} \tag{5-1-24}$$

Note that the output SNR from the matched filter depends on the energy of the waveform $s(t)$ but not on the detailed characteristics of $s(t)$. This is another interesting property of the matched filter.

Frequency-Domain Interpretation of the Matched Filter The matched filter has an interesting frequency-domain interpretation. Since $h(t) = s(T-t)$,

the Fourier transform of this relationship is

$$\begin{aligned} H(f) &= \int_0^T s(T-t)e^{-j2\pi ft}\,dt \\ &= \left[\int_0^T s(\tau)e^{j2\pi f\tau}\,d\tau\right]e^{-j2\pi fT} \\ &= S^*(f)e^{-j2\pi fT} \end{aligned} \tag{5-1-25}$$

We observe that the matched filter has a frequency response that is the complex conjugate of the transmitted signal spectrum multiplied by the phase factor $e^{-j2\pi fT}$, which represents the sampling delay of T. In other words, $|H(f)| = |S(f)|$, so that the magnitude response of the matched filter is identical to the transmitted signal spectrum. On the other hand, the phase of $H(f)$ is the negative of the phase of $S(f)$.

Now, if the signal $s(t)$ with spectrum $S(f)$ is passed through the matched filter, the filter output has a spectrum $Y(f) = |S(f)|^2 e^{-j2\pi fT}$. Hence, the output waveform is

$$\begin{aligned} y_s(t) &= \int_{-\infty}^{\infty} Y(f)e^{j2\pi ft}\,df \\ &= \int_{-\infty}^{\infty} |S(f)|^2 e^{-j2\pi fT}e^{j2\pi ft}\,df \end{aligned} \tag{5-1-26}$$

By sampling the output of the matched filter at $t = T$, we obtain

$$y_s(T) = \int_{-\infty}^{\infty} |S(f)|^2\,df = \int_0^T s^2(t)\,dt = \mathcal{E} \tag{5-1-27}$$

where the last step follows from Parseval's relation.

The noise at the output of the matched filter has a power spectral density

$$\Phi_0(f) = \tfrac{1}{2}|H(f)|^2 N_0 \tag{5-1-28}$$

Hence, the total noise power at the output of the matched filter is

$$\begin{aligned} P_n &= \int_{-\infty}^{\infty} \Phi_0(f)\,df \\ &= \tfrac{1}{2}N_0 \int_{-\infty}^{\infty} |H(f)|^2\,df = \tfrac{1}{2}N_0 \int_{-\infty}^{\infty} |S(f)|^2\,df = \tfrac{1}{2}\mathcal{E}N_0 \end{aligned} \tag{5-1-29}$$

The output SNR is simply the ratio of the signal power P_s, given by

$$P_s = y_s^2(T) \tag{5-1-30}$$

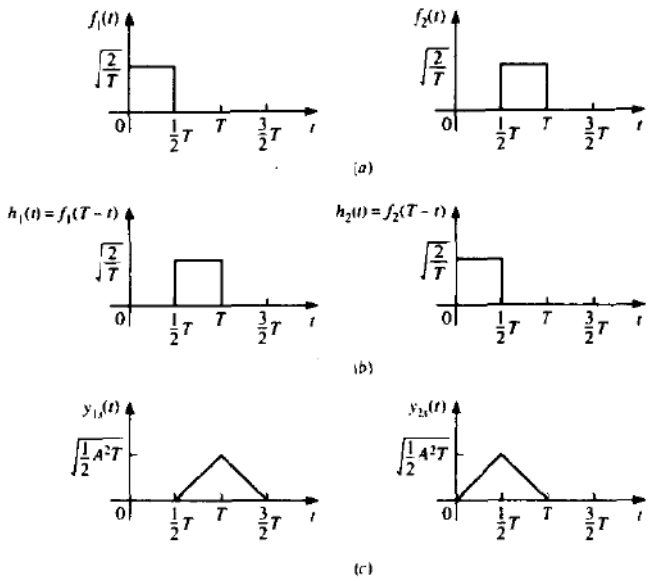


FIGURE 5-1-8 Basis functions and matched filter responses for Example 5-1-2.

to the noise power P_n. Hence,

$$\text{SNR}_0 = \frac{P_s}{P_n} = \frac{\mathscr{E}^2}{\frac{1}{2}\mathscr{E}N_0} = \frac{2\mathscr{E}}{N_0} \tag{5-1-31}$$

which agrees with the result given by (5-1-24).

Example 5-1-2

Consider the $M = 4$ biorthogonal signals shown in Fig. 5-1-8 for transmitting information over an AWGN channel. The noise is assumed to have zero mean and power spectral density $\frac{1}{2}N_0$. Let us determine the basis functions for this signal set, the impulse responses of the matched-filter demodulators, and the output waveforms of the matched-filter demodulators when the transmitted signal is $s_1(t)$.

The $M = 4$ biorthogonal signals have dimension $N = 2$. Hence, two basis functions are needed to represent the signals. From Fig. 5-1-8, we choose $f_1(t)$ and $f_2(t)$ as

$$\begin{aligned} f_1(t) &= \begin{cases} \sqrt{2/T} & (0 \leq t \leq \frac{1}{2}T) \\ 0 & (\text{otherwise}) \end{cases} \\ f_2(t) &= \begin{cases} \sqrt{2/T} & (\frac{1}{2}T \leq t \leq T) \\ 0 & (\text{otherwise}) \end{cases} \end{aligned} \tag{5-1-32}$$

These waveforms are illustrated in Fig. 5-1-8(*a*). The impulse responses of the two matched filters are

$$h_1(t) = f_1(T-t) = \begin{cases} \sqrt{2/T} & (\frac{1}{2}T \le t \le T) \\ 0 & (\text{otherwise}) \end{cases}$$
$$h_2(t) = f_2(T-t) = \begin{cases} \sqrt{2/T} & (0 \le t \le \frac{1}{2}T) \\ 0 & (\text{otherwise}) \end{cases} \tag{5-1-33}$$

and are illustrated in Fig. 5-1-8(*b*).

If $s_1(t)$ is transmitted, the (noise-free) responses of the two matched filters are as shown in Fig. 5-1-8(*c*). Since $y_1(t)$ and $y_2(t)$ are sampled at $t = T$, we observe that $y_{1s}(T) = \sqrt{\frac{1}{2}A^2T}$ and $y_{2s}(T) = 0$. Note that $\frac{1}{2}A^2T = \mathscr{E}$, the signal energy. Hence, the received vector formed from the two matched filter outputs at the sampling instant $t = T$ is

$$\mathbf{r} = [r_1 \quad r_2] = [\sqrt{\mathscr{E}} + n_1 \quad n_2] \tag{5-1-34}$$

where $n_1 = y_{1n}(T)$ and $n_2 = y_{2n}(T)$ are the noise components at the outputs of the matched filters, given by

$$y_{kn}(T) = \int_0^T n(t) f_k(t)\, dt, \qquad k = 1, 2 \tag{5-1-35}$$

Clearly, $E(n_k) = E[y_{kn}(T)] = 0$. Their variance is

$$\begin{aligned} \sigma_n^2 = E[y_{kn}^2(T)] &= \int_0^T \int_0^T E[n(t)n(\tau)] f_k(t) f_k(\tau)\, dt\, d\tau \\ &= \tfrac{1}{2}N_0 \int_0^T \int_0^T \delta(t-\tau) f_k(\tau) f_k(t)\, dt\, d\tau \\ &= \tfrac{1}{2}N_0 \int_0^T f_k^2(t)\, dt = \tfrac{1}{2}N_0 \end{aligned} \tag{5-1-36}$$

Observe that the SNR_0 for the first matched filter is

$$\text{SNR}_0 = \frac{(\sqrt{\mathscr{E}})^2}{\frac{1}{2}N_0} = \frac{2\mathscr{E}}{N_0} \tag{5-1-37}$$

which agrees with our previous result. Also note that the four possible outputs of the two matched filters, corresponding to the four possible transmitted signals in Fig. 5-1-8 are $(r_1, r_2) = (\sqrt{\mathscr{E}} + n_1, n_2)$, $(n_1, \sqrt{\mathscr{E}} + n_2)$, $(-\sqrt{\mathscr{E}} + n_1, n_2)$ and $(n_1, -\sqrt{\mathscr{E}} + n_2)$.

5-1-3 The Optimum Detector

We have demonstrated that, for a signal transmitted over an AWGN channel, either a correlation demodulator or a matched filter demodulator produces the vector $\mathbf{r} = [r_1 \quad r_2 \quad \cdots \quad r_N]$, which contains all the relevant information in the received signal waveform. In this section, we describe the optimum decision

rule based on the observation vector $\mathbf{r}$. For this development, we assume that there is no memory in signals transmitted in successive signal intervals.

We wish to design a signal detector that makes a decision on the transmitted signal in each signal interval based on the observation of the vector $\mathbf{r}$ in each interval such that the probability of a correct decision is maximized. With this goal in mind, we consider a decision rule based on the computation of the *posterior probabilities* defined as

$$P(\text{signal } \mathbf{s}_m \text{ was transmitted} \mid \mathbf{r}), \quad m = 1, 2, \ldots, M$$

which we abbreviate as $P(\mathbf{s}_m \mid \mathbf{r})$. The decision criterion is based on selecting the signal corresponding to the maximum of the set of posterior probabilities $\{P(\mathbf{s}_m \mid \mathbf{r})\}$. Later, we show that this criterion maximizes the probability of a correct decision and, hence, minimizes the probability of error. This decision criterion is called the *maximum a posteriori probability* (MAP) criterion.

Using Bayes' rule, the posterior probabilities may be expressed as

$$P(\mathbf{s}_m \mid \mathbf{r}) = \frac{p(\mathbf{r} \mid \mathbf{s}_m)P(\mathbf{s}_m)}{p(\mathbf{r})} \tag{5-1-38}$$

where $p(\mathbf{r} \mid \mathbf{s}_m)$ is the conditional pdf of the observed vector given $\mathbf{s}_m$, and $P(\mathbf{s}_m)$ is the *a priori probability* of the mth signal being transmitted. The denominator of (5-1-38) may be expressed as

$$p(\mathbf{r}) = \sum_{m=1}^{M} p(\mathbf{r} \mid \mathbf{s}_m)P(\mathbf{s}_m) \tag{5-1-39}$$

From (5-1-38) and (5-1-39), we observe that the computation of the posterior probabilities $P(\mathbf{s}_m \mid \mathbf{r})$ requires knowledge of the *a priori probabilities* $P(\mathbf{s}_m)$ and the conditional pdfs $p(\mathbf{r} \mid \mathbf{s}_m)$ for $m = 1, 2, \ldots, M$.

Some simplification occurs in the MAP criterion when the M signals are equally probable a priori, i.e., $P(\mathbf{s}_m) = 1/M$ for all M. Furthermore, we note that the denominator in (5-1-38) is independent of which signal is transmitted. Consequently, the decision rule based on finding the signal that maximizes $P(\mathbf{s}_m \mid \mathbf{r})$ is equivalent to finding the signal that maximizes $p(\mathbf{r} \mid \mathbf{s}_m)$.

The conditional pdf $p(\mathbf{r} \mid \mathbf{s}_m)$ or any monotonic function of it is usually called the *likelihood function*. The decision criterion based on the maximum of $p(\mathbf{r} \mid \mathbf{s}_m)$ over the M signals is called the *maximum-likelihood* (ML) *criterion*. We observe that a detector based on the MAP criterion and one that is based on the ML criterion make the same decisions as long as the a priori probabilities $P(\mathbf{s}_m)$ are all equal, i.e., the signals $\{\mathbf{s}_m\}$ are equiprobable.

In the case of an AWGN channel, the likelihood function $p(\mathbf{r} \mid \mathbf{s}_m)$ is given by (5-1-12). To simplify the computations, we may work with the natural logarithm of $p(\mathbf{r} \mid \mathbf{s}_m)$, which is a monotonic function. Thus,

$$\ln p(\mathbf{r} \mid \mathbf{s}_m) = -\tfrac{1}{2}N \ln (\pi N_0) - \frac{1}{N_0} \sum_{k=1}^{N} (r_k - s_{mk})^2 \tag{5-1-40}$$

The maximum of $\ln p(\mathbf{r} \mid \mathbf{s}_m)$ over $\mathbf{s}_m$ is equivalent to finding the signal $\mathbf{s}_m$ that minimizes the Euclidean distance

$$D(\mathbf{r}, \mathbf{s}_m) = \sum_{k=1}^{N} (r_k - s_{mk})^2 \tag{5-1-41}$$

We call $D(\mathbf{r}, \mathbf{s}_m)$, $m = 1, 2, \ldots, M$, the *distance metrics*. Hence, for the AWGN channel, the decision rule based on the ML criterion reduces to finding the signal $\mathbf{s}_m$ that is closest in distance to the received signal vector $\mathbf{r}$. We shall refer to this decision rule as *minimum distance detection*.

Another interpretation of the optimum decision rule based on the ML criterion is obtained by expanding the distance metrics in (5-1-41) as

$$\begin{aligned} D(\mathbf{r}, \mathbf{s}_m) &= \sum_{n=1}^{N} r_n^2 - 2 \sum_{n=1}^{N} r_n s_{mn} + \sum_{n=1}^{N} s_{mn}^2 \\ &= |\mathbf{r}|^2 - 2\mathbf{r} \cdot \mathbf{s}_m + |\mathbf{s}_m|^2, \qquad m = 1, 2, \ldots, M \end{aligned} \tag{5-1-42}$$

The term $|\mathbf{r}|^2$ is common to all decision metrics, and, hence, it may be ignored in the computations of the metrics. The result is a set of modified distance metrics

$$D'(\mathbf{r}, \mathbf{s}_m) = -2\mathbf{r} \cdot \mathbf{s}_m + |\mathbf{s}_m|^2 \tag{5-1-43}$$

Note that selecting the signal $\mathbf{s}_m$ that minimizes $D'(\mathbf{r}, \mathbf{s}_m)$ is equivalent to selecting the signal that maximizes the metric $C(\mathbf{r}, \mathbf{s}_m) = -D'(\mathbf{r}, \mathbf{s}_m)$, i.e.,

$$C(\mathbf{r}, \mathbf{s}_m) = 2\mathbf{r} \cdot \mathbf{s}_m - |\mathbf{s}_m|^2 \tag{5-1-44}$$

The term $\mathbf{r} \cdot \mathbf{s}_m$ represents the projection of the received signal vector onto each of the M possible transmitted signal vectors. The value of each of these projections is a measure of the correlation between the received vector and the mth signal. For this reason, we call $C(\mathbf{r}, \mathbf{s}_m)$, $m = 1, 2, \ldots, M$, the *correlation metrics* for deciding which of the M signals was transmitted. Finally, the terms $|\mathbf{s}_m|^2 = \mathcal{E}_m$, $m = 1, 2, \ldots, M$, may be viewed as bias terms that serve as compensation for signal sets that have unequal energies, such as PAM. If all signals have the same energy, $|\mathbf{s}_m|^2$ may also be ignored in the computation of the correlation metrics $C(\mathbf{r}, \mathbf{s}_m)$ and the distance metrics $D(\mathbf{r}, \mathbf{s}_m)$ or $D'(\mathbf{r}, \mathbf{s}_m)$.

It is easy to show (see Problem 5-5) that the correlation metrics $C(\mathbf{r}, \mathbf{s}_m)$ can also be expressed as

$$C(\mathbf{r}, \mathbf{s}_m) = 2 \int_0^T r(t) s_m(t)\, dt - \mathcal{E}_m, \qquad m = 0, 1, \ldots, M \tag{5-1-45}$$

Therefore, these metrics can be generated by a demodulator that cross-correlates the received signal $r(t)$ with each of the M possible transmitted signals and adjusts each correlator output for the bias in the case of unequal signal energies. Equivalently, the received signal may be passed through a bank of M filters matched to the possible transmitted signals $\{s_m(t)\}$ and

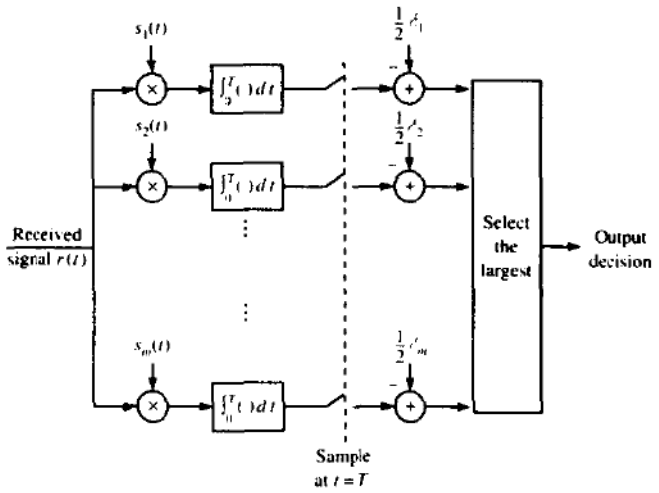


FIGURE 5-1-9 An alternative realization of the optimum AWGN receiver.

sampled at $t = T$, the end of the symbol interval. Consequently, the optimum receiver (demodulator and detector) can be implemented in the alternative configuration illustrated in Fig. 5-1-9.

In summary, we have demonstrated that the optimum ML detector computes a set of M distances $D(\mathbf{r}, \mathbf{s}_m)$ or $D'(\mathbf{r}, \mathbf{s}_m)$ and selects the signal corresponding to the smallest (distance) metric. Equivalently, the optimum ML detector computes a set of M correlation metrics $C(\mathbf{r}, \mathbf{s}_m)$ and selects the signal corresponding to the largest correlation metric.

The above development for the optimum detector treated the important case in which all signals are equally probable. In this case, the MAP criterion is equivalent to the ML criterion. However, when the signals are not equally probable, the optimum MAP detector bases its decision on the probabilities $P(\mathbf{s}_m \mid \mathbf{r})$, $m = 1, 2, \ldots, M$, given by (5-1-38) or, equivalently, on the *metrics*,

$$PM(\mathbf{r}, \mathbf{s}_m) = p(\mathbf{r} \mid \mathbf{s}_m)P(\mathbf{s}_m)$$

The following example illustrates this computation for binary PAM signals.

Example 5-1-3

Consider the case of binary PAM signals in which the two possible signal points are $s_1 = -s_2 = \sqrt{\mathscr{E}_b}$, where $\mathscr{E}_b$ is the energy per bit. The prior probabilities are $P(s_1) = p$ and $P(s_2) = 1 - p$. Let us determine the metrics for the optimum MAP detector when the transmitted signal is corrupted with AWGN.

The received signal vector (one-dimensional) for binary PAM is

$$r = \pm\sqrt{\mathscr{E}_b} + y_n(T) \tag{5-1-46}$$

where $y_n(T)$ is a zero-mean Gaussian random variable with variance $\sigma_n^2 = \frac{1}{2}N_0$. Consequently, the conditional pdfs $p(r \mid s_m)$ for the two signals are

$$p(r \mid s_1) = \frac{1}{\sqrt{2\pi}\,\sigma_n} \exp\left[-\frac{(r - \sqrt{\mathscr{E}_b})^2}{2\sigma_n^2}\right] \tag{5-1-47}$$

$$p(r \mid s_2) = \frac{1}{\sqrt{2\pi}\,\sigma_n} \exp\left[-\frac{(r + \sqrt{\mathscr{E}_b})^2}{2\sigma_n^2}\right] \tag{5-1-48}$$

Then the metrics $PM(\mathbf{r}, \mathbf{s}_1)$ and $PM(\mathbf{r}, \mathbf{s}_2)$ are

$$\begin{aligned} PM(\mathbf{r}, \mathbf{s}_1) &= pp(r \mid s_1) \\ &= \frac{p}{\sqrt{2\pi}\,\sigma_n} \exp\left[-\frac{(r - \sqrt{\mathscr{E}_b})^2}{2\sigma_n^2}\right] \end{aligned} \tag{5-1-49}$$

$$PM(\mathbf{r}, \mathbf{s}_2) = \frac{1-p}{\sqrt{2\pi}\,\sigma_n} \exp\left[-\frac{(r + \sqrt{\mathscr{E}_b})^2}{2\sigma_n^2}\right] \tag{5-1-50}$$

If $PM(\mathbf{r}, \mathbf{s}_1) > PM(\mathbf{r}, \mathbf{s}_2)$, we select s_1 as the transmitted signal; otherwise, we select s_2. This decision rule may be expressed as

$$\frac{PM(\mathbf{r}, \mathbf{s}_1)}{PM(\mathbf{r}, \mathbf{s}_2)} \underset{s_2}{\overset{s_1}{\gtrless}} 1 \tag{5-1-51}$$

But

$$\frac{PM(\mathbf{r}, \mathbf{s}_1)}{PM(\mathbf{r}, \mathbf{s}_2)} = \frac{p}{1-p} \exp\left[\frac{(r + \sqrt{\mathscr{E}_b})^2 - (r - \sqrt{\mathscr{E}_b})^2}{2\sigma_n^2}\right] \tag{5-1-52}$$

so that (5-1-51) may be expressed as

$$\frac{(r + \sqrt{\mathscr{E}_b})^2 - (r - \sqrt{\mathscr{E}_b})^2}{2\sigma_n^2} \underset{s_2}{\overset{s_1}{\gtrless}} \ln \frac{1-p}{p} \tag{5-1-53}$$

or equivalently,

$$\sqrt{\mathscr{E}_b}\, r \underset{s_2}{\overset{s_1}{\gtrless}} \tfrac{1}{2}\sigma_n^2 \ln \frac{1-p}{p} = \tfrac{1}{4}N_0 \ln \frac{1-p}{p} \tag{5-1-54}$$

This is the final form for the optimum detector. It computes the correlation metric $C(\mathbf{r}, \mathbf{s}_1) = r\sqrt{\mathscr{E}_b}$ and compares it with threshold $\frac{1}{4}N_0 \ln[(1-p)/p]$. Figure 5-1-10 illustrates the two signal points s_1 and s_2. The threshold, denoted by τ_h, divides the real line into two regions, say R_1 and R_2, where R_1 consists of the set of points that are greater than τ_h and

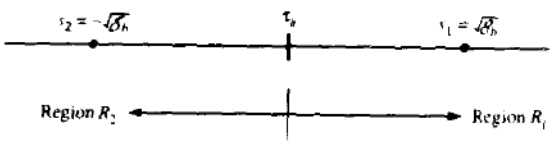


FIGURE 5-1-10 Signal space representation illustrating the operation of the optimum detector for binary (PAM) modulation.

R_2 consists of the set of points that are less than τ_h. If $r\sqrt{\mathscr{E}_b} > \tau_h$, the decision is made that s_1 was transmitted, and if $r\sqrt{\mathscr{E}_b} < \tau_h$, the decision is made that s_2 was transmitted. The threshold τ_h depends on N_0 and p. If $p = \frac{1}{2}$, $\tau_h = 0$. If $p > \frac{1}{2}$, the signal point s_1 is more probable and, hence, $\tau_h < 0$. In this case, the region R_1 is larger than R_2, so that s_1 is more likely to be selected than s_2. If $p < \frac{1}{2}$, the opposite is the case. Thus, the average probability of error is minimized.

It is interesting to note that in the case of unequal prior probabilities, it is necessary to know not only the values of the prior probabilities but also the value of the power spectral density N_0 in order to compute the threshold. When $p = \frac{1}{2}$, the threshold is zero, and knowledge of N_0 is not required by the detector.

We conclude this section with the proof that the decision rule based on the maximum-likelihood criterion minimizes the probability of error when the M signals are equally probable a priori. Let us denote by R_m the region in the N-dimensional space for which we decide that signal $s_m(t)$ was transmitted when the vector $\mathbf{r} = [r_1 \; r_2 \; \cdots \; r_N]$ is received. The probability of a decision error given that $s_m(t)$ was transmitted is

$$P(e \mid \mathbf{s}_m) = \int_{R_m^c} p(\mathbf{r} \mid \mathbf{s}_m)\, d\mathbf{r} \tag{5-1-55}$$

where R_m^c is the complement of R_m. The average probability of error is

$$\begin{aligned} P(e) &= \sum_{m=1}^{M} \frac{1}{M} P(e \mid \mathbf{s}_m) \\ &= \sum_{m=1}^{M} \frac{1}{M} \int_{R_m^c} p(\mathbf{r} \mid \mathbf{s}_m)\, d\mathbf{r} \\ &= \sum_{m=1}^{M} \frac{1}{M} \left[1 - \int_{R_m} p(\mathbf{r} \mid \mathbf{s}_m)\, d\mathbf{r} \right] \end{aligned} \tag{5-1-56}$$

Note that $P(e)$ is minimized by selecting the signal $\mathbf{s}_m$ if $p(\mathbf{r} \mid \mathbf{s}_m)$ is larger than $p(\mathbf{r} \mid \mathbf{s}_k)$ for all $m \neq k$.

When the M signals are not equally probable, the above proof can be generalized to show that the MAP criterion minimizes the average probability of error.

5-1-4 The Maximum-Likelihood Sequence Detector

When the signal has no memory, the symbol-by-symbol detector described in the preceding section is optimum in the sense of minimizing the probability of a symbol error. On the other hand, when the transmitted signal has memory, i.e., the signals transmitted in successive symbol intervals are interdependent, the optimum detector is a detector that bases its decisions on observation of a

sequence of received signals over successive signal intervals. Below, we describe two different types of detection algorithms. In this section, we describe a *maximum-likelihood sequence detection algorithm* that searches for the minimum euclidean distance path through the trellis that characterizes the memory in the transmitted signal. In the following section, we describe a maximum a posteriori probability algorithm that makes decisions on a symbol-by-symbol basis, but each symbol decision is based on an observation of a sequence of received signal vectors.

To develop the maximum likelihood sequence detection algorithm, let us consider, as an example, the NRZI signal described in Section 4-3-2. Its memory is characterized by the trellis shown in Fig. 4-3-14. The signal transmitted in each signal interval is binary PAM. Hence, there are two possible transmitted signals corresponding to the signal points $s_1 = -s_2 = \sqrt{\mathscr{E}_b}$, where $\mathscr{E}_b$ is the energy per bit. The output of the matched-filter or correlation demodulator for binary PAM in the kth signal interval may be expressed as

$$r_k = \pm\sqrt{\mathscr{E}_b} + n_k \tag{5-1-57}$$

where n_k is a zero-mean gaussian random variable with variance $\sigma_n^2 = N_0/2$. Consequently, the conditional pdfs for the two possible transmitted signals are

$$\begin{aligned} p(r_k \mid s_1) &= \frac{1}{\sqrt{2\pi}\,\sigma_n} \exp\left[-\frac{(r_k - \sqrt{\mathscr{E}_b})^2}{2\sigma_n^2}\right] \\ p(r_k \mid s_2) &= \frac{1}{\sqrt{2\pi}\,\sigma_n} \exp\left[-\frac{(r_k + \sqrt{\mathscr{E}_b})^2}{2\sigma_n^2}\right] \end{aligned} \tag{5-1-58}$$

Now, suppose we observe the sequence of matched-filter outputs $r_1, r_2, \ldots, r_K$. Since the channel noise is assumed to be white and gaussian, and $f(t - iT)$, $f(t - jT)$ for $i \neq j$ are orthogonal, it follows that $E(n_k n_j) = 0$, $k \neq j$. Hence, the noise sequence $n_1, n_2, \ldots, n_K$ is also white. Consequently, for any given transmitted sequence $\mathbf{s}^{(m)}$, the joint pdf of $r_1, r_2, \ldots, r_K$ may be expressed as a product of K marginal pdfs, i.e.,

$$\begin{aligned} p(r_1, r_2, \ldots, r_K \mid \mathbf{s}^{(m)}) &= \prod_{k=1}^{K} p(r_k \mid s_k^{(m)}) \\ &= \prod_{k=1}^{K} \frac{1}{\sqrt{2\pi}\,\sigma_n} \exp\left[-\frac{(r_k - s_k^{(m)})^2}{2\sigma_n^2}\right] \\ &= \left(\frac{1}{\sqrt{2\pi}\,\sigma_n}\right)^K \exp\left[-\sum_{k=1}^{K} \frac{(r_k - s_k^{(m)})^2}{2\sigma_n^2}\right] \end{aligned} \tag{5-1-59}$$

where either $s_k = \sqrt{\mathscr{E}_b}$ or $s_k = -\sqrt{\mathscr{E}_b}$. Then, given the received sequence $r_1, r_2, \ldots, r_K$ at the output of the matched filter or correlation demodulator, the detector determines the sequence $\mathbf{s}^{(m)} = \{s_1^{(m)}, s_2^{(m)}, \ldots, s_K^{(m)}\}$ that maximizes the conditional pdf $p(r_1, r_2, \ldots, r_K \mid \mathbf{s}^{(m)})$. Such a detector is called the *maximum-likelihood* (ML) *sequence detector.*

By taking the logarithm of (5-1-59) and neglecting the terms that are

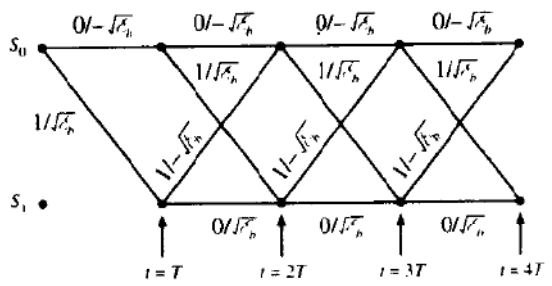


FIGURE 5-1-11 Trellis for NRZI signal.

independent of $(r_1, r_2, \ldots, r_K)$, we find that an equivalent ML sequence detector selects the sequence $\mathbf{s}^{(m)}$ that minimizes the *euclidean distance metric*

$$D(\mathbf{r}, \mathbf{s}^{(m)}) = \sum_{k=1}^{K} (r_k - s_k^{(m)})^2 \tag{5-1-60}$$

In searching through the trellis for the sequence that minimizes the euclidean distance $D(\mathbf{r}, \mathbf{s}^{(m)})$, it may appear that we must compute the distance $D(\mathbf{r}, \mathbf{s}^{(m)})$ for every possible sequence. For the NRZI example, which employs binary modulation, the total number of sequences is 2^K, where K is the number of outputs obtained from the demodulator. However, this is not the case. We may reduce the number of sequences in the trellis search by using the *Viterbi algorithm* to eliminate sequences as new data is received from the demodulator.

The Viterbi algorithm is a sequential trellis search algorithm for performing ML sequence detection. It is described in Chapter 8 as a decoding algorithm for convolutional codes. We describe it below in the context of the NRZI signal. We assume that the search process begins initially at state S_0. The corresponding trellis is shown in Fig. 5-1-11.

At time $t = T$, we receive $r_1 = s_1^{(m)} + n$ from the demodulator, and at $t = 2T$, we receive $r_2 = s_2^{(m)} + n_2$. Since the signal memory is one bit, which we denote by $L = 1$, we observe that the trellis reaches its regular (steady state) form after two transitions. Thus, upon receipt of r_2 at $t = 2T$ (and thereafter), we observe that there are two signal paths entering each of the nodes and two signal paths leaving each node. The two paths entering node S_0 at $t = 2T$ correspond to the information bits (0, 0) and (1, 1) or, equivalently, to the signal points $(-\sqrt{\mathscr{E}_b}, -\sqrt{\mathscr{E}_b})$ and $(\sqrt{\mathscr{E}_b}, -\sqrt{\mathscr{E}_b})$, respectively. The two paths entering node S_1 at $t = 2T$ correspond to the information bits (0, 1) and (1, 0) or, equivalently, to the signal points $(-\sqrt{\mathscr{E}_b}, \sqrt{\mathscr{E}_b})$ and $(\sqrt{\mathscr{E}_b}, \sqrt{\mathscr{E}_b})$, respectively.

For the two paths entering node S_0, we compute the two Euclidean distance metrics

$$\begin{aligned} D_0(0, 0) &= (r_1 + \sqrt{\mathscr{E}_b})^2 + (r_2 + \sqrt{\mathscr{E}_b})^2 \\ D_0(1, 1) &= (r_1 - \sqrt{\mathscr{E}_b})^2 + (r_2 + \sqrt{\mathscr{E}_b})^2 \end{aligned} \tag{5-1-61}$$

by using the outputs r_1 and r_2 from the demodulator. The Viterbi algorithm compares these two metrics and discards the path having the larger (greater-distance) metric.† The other path with the lower metric is saved and is called the *survivor* at $t = 2T$. The elimination of one of the two paths may be done without compromising the optimality of the trellis search, because any extension of the path with the larger distance beyond $t = 2T$ will always have a larger metric than the survivor that is extended along the same path beyond $t = 2T$.

Similarly, for the two paths entering node S_1 at $t = 2T$, we compute the two Euclidean distance metrics

$$\begin{aligned} D_1(0, 1) &= (r_1 + \sqrt{\mathscr{E}_b})^2 + (r_2 - \sqrt{\mathscr{E}_b})^2 \\ D_1(1, 0) &= (r_1 - \sqrt{\mathscr{E}_b})^2 + (r_2 - \sqrt{\mathscr{E}_b})^2 \end{aligned} \tag{5-1-62}$$

by using the outputs r_1 and r_2 from the demodulator. The two metrics are compared and the signal path with the larger metric is eliminated. Thus, at $t = 2T$, we are left with two survivor paths, one at node S_0 and the other at node S_1, and their corresponding metrics. The signal paths at nodes S_0 and S_1 are then extended along the two survivor paths.

Upon receipt of r_3 at $t = 3T$, we compute the metrics of the two paths entering state S_0. Suppose the survivors at $t = 2T$ are the paths (0, 0) at S_0 and (0, 1) at S_1. Then, the two metrics for the paths entering S_0 at $t = 3T$ are

$$\begin{aligned} D_0(0, 0, 0) &= D_0(0, 0) + (r_3 + \sqrt{\mathscr{E}_b})^2 \\ D_0(0, 1, 1) &= D_1(0, 1) + (r_3 + \sqrt{\mathscr{E}_b})^2 \end{aligned} \tag{5-1-63}$$

These two metrics are compared and the path with the larger (greater-distance) metric is eliminated. Similarly, the metrics for the two paths entering S_1 at $t = 3T$ are

$$\begin{aligned} D_1(0, 0, 1) &= D_0(0, 0) + (r_3 - \sqrt{\mathscr{E}_b})^2 \\ D_1(0, 1, 0) &= D_1(0, 1) + (r_3 - \sqrt{\mathscr{E}_b})^2 \end{aligned} \tag{5-1-64}$$

These two metrics are compared and the path with the larger (greater-distance) metric is eliminated.

This process is continued as each new signal sample is received from the demodulator. Thus, the Viterbi algorithm computes two metrics for the two signal paths entering a node at each stage of the trellis search and eliminates one of the two paths at each node. The two survivor paths are then extended forward to the next state. Therefore, the number of paths searched in the trellis is reduced by a factor of two at each stage.

It is relatively easy to generalize the trellis search performed by the Viterbi algorithm for M-ary modulation. For example, delay modulation employs

† Note that, for NRZI, the reception of r_2 from the demodulator neither increases nor decreases the relative difference between the two metrics, $D_0(0, 0)$ and $D_0(1, 1)$. At this point, one may ponder on the implication of this observation. In any case, we continue with the description of the ML sequence detector based on the Viterbi algorithm.

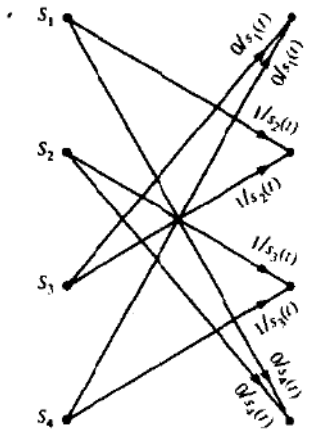


FIGURE 5-1-12 One stage of trellis diagram for delay modulation.

$M = 4$ signals and is characterized by the four-state trellis shown in Fig. 5-1-12. We observe that each state has two signal paths entering and two signal paths leaving each node. The memory of the signal is $L = 1$. Hence, the Viterbi algorithm will have four survivors at each stage and their corresponding metrics. Two metrics corresponding to the two entering paths are computed at each node, and one of the two signal paths entering the node is eliminated at each state of the trellis. Thus, the Viterbi algorithm minimizes the number of trellis paths searched in performing ML sequence detection.

From the description of the Viterbi algorithm given above, it is unclear as to how decisions are made on the individual detected information symbols given the surviving sequences. If we have advanced to some stage, say K, where $K \gg L$ in the trellis, and we compare the surviving sequences, we shall find that with probability approaching one all surviving sequences will be identical in bit (or symbol) positions $K - 5L$ and less. In a practical implementation of the Viterbi algorithm, decisions on each information bit (or symbol) are forced after a delay of $5L$ bits (or symbols), and hence, the surviving sequences are truncated to the $5L$ most recent bits (or symbols). Thus, a variable delay in bit or symbol detection is avoided. The loss in performance resulting from the suboptimum detection procedure is negligible if the delay is at least $5L$.

Example 5-1-4

Consider the decision rule for detecting the data sequence in an NRZI signal with a Viterbi algorithm having a delay of $5L$ bits. The trellis for the NRZI signal is shown in Fig. 5-1-11. In this case, $L = 1$, hence the delay in bit detection is set to five bits. Hence, at $t = 6T$, we shall have two surviving sequences, one for each of the two states and the corresponding metrics $\mu_6(b_1, b_2, b_3, b_4, b_5, b_6)$ and $\mu_6(b_1', b_2', b_3', b_4', b_5', b_6')$. At this stage, with probability nearly equal to one, the bit b_1 will be the same as b_1'; that is,

both surviving sequences will have a common first branch. If $b_1 \neq b_1'$, we may select the bit (b_1 or b_1') corresponding to the smaller of the two metrics. Then the first bit is dropped from the two surviving sequences. At $t = 7T$, the two metrics $\mu_7(b_2, b_3, b_4, b_5, b_6, b_7)$ and $\mu_7(b_2', b_3', b_4', b_5', b_6', b_7')$ will be used to determine the decision on bit b_2. This process continues at each stage of the search through the trellis for the minimum distance sequence. Thus the detection delay is fixed at five bits.†

5-1-5 A Symbol-by-Symbol Detector for Signals with Memory

In contrast to the maximum-likelihood sequence detector for detecting the transmitted information, we now describe a detector that makes symbol-by-symbol decisions based on the computation of the maximum a posteriori probability (MAP) for each detected symbol. Hence, this detector is optimum in the sense that it minimizes the probability of a symbol error. The detection algorithm that is presented below is due to Abend and Fritchman (1970), who developed it as a detection algorithm for channels with intersymbol interference, i.e., channels with memory.

We illustrate the algorithm in the context of detecting a PAM signal with M possible levels. Suppose that it is desired to detect the information symbol transmitted in the kth signal interval, and let $r_1, r_2, \ldots, r_{k+D}$ be the observed received sequence, where D is the delay parameter which is chosen to exceed the signal memory, i.e., $D \geq L$, where L is the inherent memory in the signal. On the basis of the received sequence, we compute the posterior probabilities

$$P(s^{(k)} = A_m \mid r_{k+D}, r_{k+D-1}, \ldots, r_1) \tag{5-1-65}$$

for the M possible symbol values and choose the symbol with the largest probability. Since

$$P(s^{(k)} = A_m \mid r_{k+D}, \ldots, r_1) = \frac{p(r_{k+D}, \ldots, r_1 \mid s^{(k)} = A_m) P(s^{(k)} = A_m)}{p(r_{k+D}, r_{k+D-1}, \ldots, r_1)} \tag{5-1-66}$$

and since the denominator is common for all M probabilities, the maximum a posteriori probability (MAP) criterion is equivalent to choosing the value of $s^{(k)}$ that maximizes the numerator of (5-1-66). Thus, the criterion for deciding on the transmitted symbol $s^{(k)}$ is

$$\hat{s}^{(k)} = \arg \left\{ \max_{s^{(k)}} p(r_{k+D}, \ldots, r_1 \mid s^{(k)} = A_m) P(s^{(k)} = A_m) \right\} \tag{5-1-67}$$

† One may have observed by now that the ML sequence detector and the symbol-by-symbol detector that ignores the memory in the NRZI signal reach the same decisions. Hence, there is no need for a decision delay. Nevertheless, the procedure described above applies in general.

When the symbols are equally probable, the probability $P(s^{(k)} = A_m)$ may be dropped from the computation.

The algorithm for computing the probabilities in (5-1-67) recursively begins with the first symbol $s^{(1)}$. We have

$$\begin{aligned}\tilde{s}^{(1)} &= \arg\left\{\max_{s^{(1)}} p(r_{k+D}, \ldots, r_1 \mid s^{(1)} = A_m)P(s^{(1)} = A_m)\right\} \\ &= \arg\left\{\max_{s^{(1)}} \sum_{s^{(1+D)}} \cdots \sum_{s^{(2)}} p(r_{1+D}, \ldots, r_1 \mid s^{(1+D)}, \ldots, s^{(1)})P(s^{(1+D)}, \ldots, s^{(1)})\right\} \\ &= \arg\left\{\max_{s^{(1)}} \sum_{s^{(1+D)}} \cdots \sum_{s^{(2)}} p_1(s^{(1+D)}, \ldots, s^{(2)}, s^{(1)})\right\} \qquad (5\text{-}1\text{-}68)\end{aligned}$$

where $\tilde{s}^{(1)}$ denotes the decision on $s^{(1)}$ and, for mathematical convenience, we have defined

$$p_1(s^{(1+D)}, \ldots, s^{(2)}, s^{(1)}) \equiv p(r_{1+D}, \ldots, r_1 \mid s^{(1+D)}, \ldots, s^{(1)})P(s^{(1+D)}, \ldots, s^{(1)}) \qquad (5\text{-}1\text{-}69)$$

The joint probability $P(s^{(1+D)}, \ldots, s^{(2)}, s^{(1)})$ may be omitted if the symbols are equally probable and statistically independent. As a consequence of the statistical independence of the additive noise sequence, we have

$$\begin{aligned}&p(r_{1+D}, \ldots, r_1 \mid s^{(1+D)}, \ldots, s^{(1)}) \\ &\quad = p(r_{1+D} \mid s^{(1+D)}, \ldots, s^{(1+D-L)})p(r_D \mid s^{(D)}, \ldots, s^{(D-L)}) \cdots \\ &\qquad\qquad p(r_2 \mid s^{(2)}, s^{(1)})p(r_1 \mid s^{(1)}) \qquad (5\text{-}1\text{-}70)\end{aligned}$$

where we assume that $s^{(k)} = 0$ for $k \leq 0$.

For detection of the symbol $s^{(2)}$, we have

$$\begin{aligned}\tilde{s}^{(2)} &= \arg\left\{\max_{s^{(2)}} p(r_{2+D}, \ldots, r_1 \mid s^{(2)} = A_m)P(s^{(2)} = A_m)\right\} \\ &= \arg\left\{\max_{s^{(2)}} \sum_{s^{(2+D)}} \cdots \sum_{s^{(3)}} p(r_{2+D}, \ldots, r_1 \mid s^{(2+D)}, \ldots, s^{(2)})P(s^{(2+D)}, \ldots, s^{(2)})\right\}\end{aligned} \qquad (5\text{-}1\text{-}71)$$

The joint conditional probability in the multiple summation can be expressed as

$$\begin{aligned}&p(r_{2+D}, \ldots, r_1 \mid s^{(2+D)}, \ldots, s^{(2)}) \\ &\quad = p(r_{2+D} \mid s^{(2+D)}, \ldots, s^{(2+D-L)})p(r_{1+D} \ldots, r_1 \mid s^{(1+D)}, \ldots, s^{(2)}) \qquad (5\text{-}1\text{-}72)\end{aligned}$$

Furthermore, the joint probability

$$p(r_{1+D}, \ldots, r_1 \mid s^{(1+D)}, \ldots, s^{(2)})P(s^{(1+D)}, \ldots, s^{(2)})$$

can be obtained from the probabilities computed previously in the detection of $s^{(1)}$. That is,

$$\begin{aligned} &p(r_{1+D}, \ldots, r_1 \mid s^{(1+D)}, \ldots, s^{(2)}) \\ &\quad = \sum_{s^{(1)}} p(r_{1+D}, \ldots, r_1 \mid s^{(1+D)}, \ldots, s^{(1)})P(s^{(1+D)}, \ldots, s^{(1)}) \\ &\quad = \sum_{s^{(1)}} p_1(s^{(1+D)}, \ldots, s^{(2)}, s^{(1)}) \end{aligned} \tag{5-1-73}$$

Thus, by combining (5-1-73) and (5-1-72) and then substituting into (5-1-71), we obtain

$$\bar{s}^{(2)} = \arg\left\{\max_{s^{(2)}} \sum_{s^{(2+D)}} \cdots \sum_{s^{(3)}} p_2(s^{(2+D)}, \ldots, s^{(3)}, s^{(2)})\right\} \tag{5-1-74}$$

where, by definition,

$$\begin{aligned} &p_2(s^{(2+D)}, \ldots, s^{(3)}, s^{(2)}) \\ &\quad = p(r_{2+D} \mid s^{(2+D)}, \ldots, s^{(2+D-L)})P(s^{(2+D)}) \sum_{s^{(1)}} p_1(s^{(1+D)}, \ldots, s^{(2)}, s^{(1)}) \end{aligned} \tag{5-1-75}$$

In general, the recursive algorithm for detecting the symbol $s^{(k)}$ is as follows: upon reception of $r_{k+D}, \ldots, r_2, r_1$, we compute

$$\begin{aligned} \bar{s}^{(k)} &= \arg\left\{\max_{s^{(k)}} p(r_{k+D}, \ldots, r_1 \mid s^{(k)})P(s^{(k)})\right\} \\ &= \arg\left\{\max_{s^{(k)}} \sum_{s^{(k+D)}} \cdots \sum_{s^{(k+1)}} p_k(s^{(k+D)}, \ldots, s^{(k+1)}, s^{(k)})\right\} \end{aligned} \tag{5-1-76}$$

where, by definition,

$$\begin{aligned} &p_k(s^{(k+D)}, \ldots, s^{(k-1)}, s^{(k)}) \\ &\quad = p(r_{k+D} \mid s^{(k+D)}, \ldots, s^{(k+D-L)})P(s^{(k+D)}) \sum_{s^{(k-1)}} p_{k-1}(s^{(k-1+D)}, \ldots, s^{(k-1)}) \end{aligned} \tag{5-1-77}$$

Thus, the recursive nature of the algorithm is established by the relations (5-1-76) and (5-1-77).

The major problem with the algorithm is its computational complexity. In particular, the averaging performed over the symbols $s^{(k+D)}, \ldots, s^{(k+1)}, s^{(k)}$ in (5-1-76) involves a large amount of computation per received signal, especially if the number M of amplitude levels $\{A_m\}$ is large. On the other hand, if M is small and the memory L is relatively short, this algorithm is easily implemented.

5-2 PERFORMANCE OF THE OPTIMUM RECEIVER FOR MEMORYLESS MODULATION

In this section, we evaluate the probability of error for the memoryless modulation signals described in Section 4-3-1. First, we consider binary PAM signals and then M-ary signals of various types.

5-2-1 Probability of Error for Binary Modulation

Let us consider binary PAM signals where the two signal waveforms are $s_1(t) = g(t)$ and $s_2(t) = -g(t)$, and $g(t)$ is an arbitrary pulse that is nonzero in the interval $0 \leq t \leq T_b$ and zero elsewhere.

Since $s_1(t) = -s_2(t)$, these signals are said to be *antipodal*. The energy in the pulse $g(t)$ is $\mathcal{E}_b$. As indicated in Section 4-3-1, PAM signals are one-dimensional, and, hence, their geometric representation is simply the one-dimensional vector $s_1 = \sqrt{\mathcal{E}_b}$, $s_2 = -\sqrt{\mathcal{E}_b}$. Figure 5-2-1 illustrates the two signal points.

Let us assume that the two signals are equally likely and that signal $s_1(t)$ was transmitted. Then, the received signal from the (matched filter or correlation) demodulator is

$$r = s_1 + n = \sqrt{\mathcal{E}_b} + n \tag{5-2-1}$$

where n represents the additive gaussian noise component, which has zero mean and variance $\sigma_n^2 = \frac{1}{2}N_0$. In this case, the decision rule based on the correlation metric given by (5-1-44) compares r with the threshold zero. If $r > 0$, the decision is made in favor of $s_1(t)$, and if $r < 0$, the decision is made that $s_2(t)$ was transmitted. Clearly, the two conditional pdfs of r are

$$p(r \mid s_1) = \frac{1}{\sqrt{\pi N_0}} e^{-(r - \sqrt{\mathcal{E}_b})^2/N_0} \tag{5-2-2}$$

$$p(r \mid s_2) = \frac{1}{\sqrt{\pi N_0}} e^{-(r + \sqrt{\mathcal{E}_b})^2/N_0} \tag{5-2-3}$$

FIGURE 5-2-1 Signal points for binary antipodal signals.

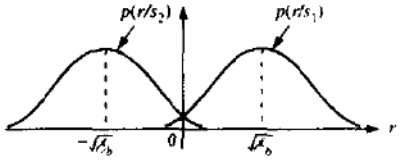


FIGURE 5-2-2 Conditional pdfs of two signals.

These two conditional pdfs are shown in Fig. 5-2-2.

Given that $s_1(t)$ was transmitted, the probability of error is simply the probability that $r < 0$, i.e.,

$$\begin{aligned} P(e \mid s_1) &= \int_{-\infty}^{0} p(r \mid s_1)\, dr \\ &= \frac{1}{\sqrt{\pi N_0}} \int_{-\infty}^{0} \exp\left[-\frac{(r - \sqrt{\mathscr{E}_b})^2}{N_0} \right] dr \\ &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{-\sqrt{2\mathscr{E}_b/N_0}} e^{-x^2/2}\, dx \\ &= \frac{1}{\sqrt{2\pi}} \int_{\sqrt{2\mathscr{E}_b/N_0}}^{\infty} e^{-x^2/2}\, dx \\ &= Q\left(\sqrt{\frac{2\mathscr{E}_b}{N_0}} \right) \end{aligned} \tag{5-2-4}$$

where $Q(x)$ is the Q-function defined in (2-1-97). Similarly, if we assume that $s_2(t)$ was transmitted, $r = -\sqrt{\mathscr{E}_b} + n$ and the probability that $r > 0$ is also $P(e \mid s_2) = Q(\sqrt{2\mathscr{E}_b/N_0})$. Since the signals $s_1(t)$ and $s_2(t)$ are equally likely to be transmitted, the average probability of error is

$$\begin{aligned} P_b &= \tfrac{1}{2}P(e \mid s_1) + \tfrac{1}{2}P(e \mid s_2) \\ &= Q\left(\sqrt{\frac{2\mathscr{E}_b}{N_0}} \right) \end{aligned} \tag{5-2-5}$$

We should observe two important characteristics of this performance measure. First, we note that the probability of error depends only on the ratio $\mathscr{E}_b/N_0$ and not on any other detailed characteristics of the signals and the noise. Secondly, we note that $2\mathscr{E}_b/N_0$ is also the output SNR_0 from the matched-filter (and correlation) demodulator. The ratio $\mathscr{E}_b/N_0$ is usually called the *signal-to-noise ratio per bit.*

We also observe that the probability of error may be expressed in terms of the distance between the two signals s_1 and s_2. From Fig. 5-2-1, we observe that the two signals are separated by the distance $d_{12} = 2\sqrt{\mathscr{E}_b}$. By substituting $\mathscr{E}_b = \frac{1}{4}d_{12}^2$ into (5-2-5), we obtain

$$P_b = Q\left(\sqrt{\frac{d_{12}^2}{2N_0}} \right) \tag{5-2-6}$$

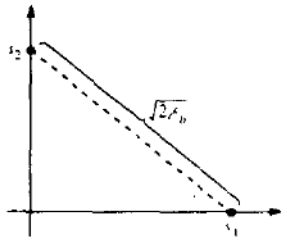


FIGURE 5-2-3 Signal points for binary orthogonal signals

This expression illustrates the dependence of the error probability on the distance between the two signal points.

Next, let us evaluate the error probability for binary orthogonal signals. Recall that the signal vectors $\mathbf{s}_1$ and $\mathbf{s}_1$ are two-dimensional, as shown in Fig. 5-2-3, and may be expressed, according to (4-3-30), as

$$\begin{aligned} \mathbf{s}_1 &= [\sqrt{\mathscr{E}_b} \quad 0] \\ \mathbf{s}_2 &= [0 \quad \sqrt{\mathscr{E}_b}] \end{aligned} \tag{5-2-7}$$

where $\mathscr{E}_b$ denotes the energy for each of the waveforms. Note that the distance between these signal points is $d_{12} = \sqrt{2\mathscr{E}_b}$.

To evaluate the probability of error, let us assume that $\mathbf{s}_1$ was transmitted. Then, the received vector at the output of the demodulator is

$$\mathbf{r} = [\sqrt{\mathscr{E}_b} + n_1 \quad n_2] \tag{5-2-8}$$

We can now substitute for r into the correlation metrics given by (5-1-44) to obtain $C(\mathbf{r}, \mathbf{s}_1)$ and $C(\mathbf{r}, \mathbf{s}_2)$. Then, the probability of error is the probability that $C(\mathbf{r}, \mathbf{s}_2) > C(\mathbf{r}, \mathbf{s}_1)$. Thus,

$$P(e \mid \mathbf{s}_1) = P[C(\mathbf{r}, \mathbf{s}_2) > C(\mathbf{r}_1, \mathbf{s}_1)] = P[n_2 - n_1 > \sqrt{\mathscr{E}_b}] \tag{5-2-9}$$

Since n_1 and n_2 are zero-mean statistically independent gaussian random variables each with variance $\frac{1}{2}N_0$, the random variable $x = n_2 - n_1$ is zero-mean gaussian with variance N_0. Hence,

$$\begin{aligned} P(n_2 - n_1 > \sqrt{\mathscr{E}_b}) &= \frac{1}{\sqrt{2\pi N_0}} \int_{\sqrt{2\mathscr{E}_b}}^{\infty} e^{-x^2/2N_0}\, dx \\ &= \frac{1}{\sqrt{2\pi}} \int_{\sqrt{2\mathscr{E}_b/N_0}}^{\infty} e^{-x^2/2}\, dx \\ &= Q\left(\sqrt{\frac{\mathscr{E}_b}{N_0}}\right) \end{aligned} \tag{5-2-10}$$

Due to symmetry, the same error probability is obtained when we assume that

$\mathbf{s}_2$ is transmitted. Consequently, the average error probability for binary orthogonal signals is

$$P_b = Q\left(\sqrt{\frac{\mathscr{E}_b}{N_0}}\right) = Q(\sqrt{\gamma_b}) \tag{5-2-11}$$

where, by definition, γ_b is the SNR per bit.

If we compare the probability of error for binary antipodal signals with that for binary orthogonal signals, we find that orthogonal signals require a factor of two increase in energy to achieve the same error probability as antipodal signals. Since $10 \log_{10} 2 = 3$ dB, we say that orthogonal signals are 3 dB poorer than antipodal signals. The difference of 3 dB is simply due to the distance between the two signal points, which is $d_{12}^2 = 2\mathscr{E}_b$ for orthogonal signals, whereas $d_{12}^2 = 4\mathscr{E}_b$ for antipodal signals.

The error probability versus $10 \log_{10} \mathscr{E}_b/N_0$ for these two types of signals is shown in Fig. 5-2-4. As observed from this figure, at any given error probability, the $\mathscr{E}_b/N_0$ required for orthogonal signals is 3 dB more than that for antipodal signals.

5-2-2 Probability of Error for M-ary Orthogonal Signals

For equal energy orthogonal signals, the optimum detector selects the signal resulting in the largest cross correlation between the received vector $\mathbf{r}$ and each of the M possible transmitted signal vectors $\{\mathbf{s}_m\}$, i.e.,

$$C(\mathbf{r}, \mathbf{s}_m) = \mathbf{r} \cdot \mathbf{s}_m = \sum_{k=1}^{M} r_k s_{mk}, \qquad m = 1, 2, \ldots, M \tag{5-2-12}$$

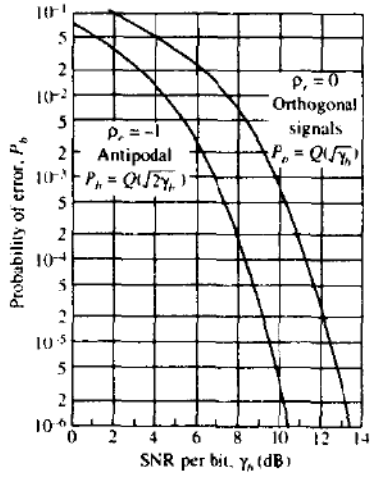


FIGURE 5-2-4 Probability of error for binary signals.

To evaluate the probability of error, let us suppose that the signal $\mathbf{s}_1$ is transmitted. Then the received signal vector is

$$\mathbf{r} = [\sqrt{\mathscr{E}_s} + n_1 \quad n_2 \quad n_3 \quad \dots \quad n_M] \tag{5-2-13}$$

where $n_1, n_2, \dots, n_M$ are zero-mean, mutually statistically independent gaussian random variables with equal variance $\sigma_n^2 = \frac{1}{2}N_0$. In this case, the outputs from the bank of M correlators are

$$\begin{aligned} C(\mathbf{r}, \mathbf{s}_1) &= \sqrt{\mathscr{E}_s}(\sqrt{\mathscr{E}_s} + n_1) \\ C(\mathbf{r}, \mathbf{s}_2) &= \sqrt{\mathscr{E}_s} n_2 \\ &\vdots \qquad \vdots \\ C(\mathbf{r}, \mathbf{s}_M) &= \sqrt{\mathscr{E}_s} n_M \end{aligned} \tag{5-2-14}$$

Note that the scale factor $\mathscr{E}_s$ may be elminated from the correlator outputs by dividing each output by $\sqrt{\mathscr{E}_s}$. Then, with this normalization, the pdf of the first correlator output ($r_1 = \sqrt{\mathscr{E}_s} + n_1$) is

$$p_{r_1}(x_1) = \frac{1}{\sqrt{\pi N_0}} \exp\left[-\frac{(x_1 - \sqrt{\mathscr{E}_s})^2}{N_0}\right] \tag{5-2-15}$$

and the pdfs of the other $M - 1$ correlator outputs are

$$p_{r_m}(x_m) = \frac{1}{\sqrt{\pi N_0}} e^{-x_m^2/N_0}, \qquad m = 2, 3, \dots, M \tag{5-2-16}$$

It is mathematically convenient to first derive the probability that the detector makes a correct decision. This is the probability that r_1 is larger than each of the other $M - 1$ correlator outputs $n_2, n_3, \dots n_M$. This probability may be expressed as

$$P_c = \int_{-\infty}^{\infty} P(n_2 < r_1, n_3 < r_1, \dots, n_M < r_1 \mid r_1) p(r_1)\, dr_1 \tag{5-2-17}$$

where $P(n_2 < r_1, n_3 < r_1, \dots, n_M < r_1 \mid r_1)$ denotes the joint probability that $n_2, n_3, \dots, n_M$ are all less than r_1, conditioned on any given r_1. Then this joint probability is averaged over all r_1. Since the $\{r_m\}$ are statistically independent, the joint probability factors into a product of $M - 1$ marginal probabilities of the form

$$\begin{aligned} P(n_m < r_1 \mid r_1) &= \int_{-\infty}^{r_1} p_{r_m}(x_m)\, dx_m, \qquad m = 2, 3, \dots, M \\ &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{r_1\sqrt{2/N_0}} e^{-x^2/2}\, dx \end{aligned} \tag{5-2-18}$$

These probabilities are identical for $m = 2, 3, \dots, M$, and, hence, the joint

probability under consideration is simply the result in (5-2-18) raised to the $(M-1)$th power. Thus, the probability of a correct decision is

$$P_c = \int_{-\infty}^{\infty} \left(\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{r_1\sqrt{2\mathscr{E}_s/N_0}} e^{-x^2/2}\, dx \right)^{M-1} p(r_1)\, dr_1 \tag{5-2-19}$$

and the probability of a (k-bit) symbol error is

$$P_M = 1 - P_c \tag{5-2-20}$$

where

$$P_M = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \left[1 - \left(\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{y} e^{-x^2/2}\, dx \right)^{M-1} \right] \exp\left[-\tfrac{1}{2}\left(y - \sqrt{\frac{2\mathscr{E}_s}{N_0}} \right)^2 \right] dy \tag{5-2-21}$$

The same expression for the probability of error is obtained when any one of the other $M-1$ signals is transmitted. Since all the M signals are equally likely, the expression for P_M given in (5-2-21) is the average probability of a symbol error. This expression can be evaluated numerically.

In comparing the performance of various digital modulation methods, it is desirable to have the probability of error expressed in terms of the SNR per bit, $\mathscr{E}_b/N_0$, instead of the SNR per symbol, $\mathscr{E}_s/N_0$. With $M = 2^k$, each symbol conveys k bits of information, and hence $\mathscr{E}_s = k\mathscr{E}_b$. Thus, (5-2-21) may be expressed in terms of $\mathscr{E}_b/N_0$ by substituting for $\mathscr{E}_s$.

Sometimes, it is also desirable to convert the probability of a symbol error into an equivalent probability of a binary digit error. For equiprobable orthogonal signals, all symbol errors are equiprobable and occur with probability

$$\frac{P_M}{M-1} = \frac{P_M}{2^k - 1} \tag{5-2-22}$$

Furthermore, there are $\binom{k}{n}$ ways in which n bits out of k may be in error. Hence, the average number of bit errors per k-bit symbol is

$$\sum_{n=1}^{k} \binom{k}{n} \frac{P_M}{2^k - 1} = k \frac{2^{k-1}}{2^k - 1} P_M \tag{5-2-23}$$

and the average bit error probability is just the result in (5-2-23) divided by k, the number of bits per symbol. Thus,

$$P_b = \frac{2^{k-1}}{2^k - 1} P_M \approx \frac{P_M}{2}, \qquad k \gg 1 \tag{5-2-24}$$

The graphs of the probability of a binary digit error as a function of the

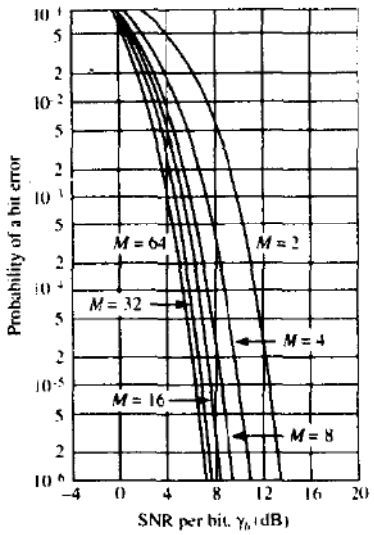


FIGURE 5-2-5 Probability of bit error for coherent detection of orthogonal signals.

SNR per bit, $\mathscr{E}_b/N_0$, are shown in Fig. 5-2-5 for $M = 2, 4, 8, 16, 32$ and 64. This figure illustrates that, by increasing the number M of waveforms, one can reduce the SNR per bit required to achieve a given probability of a bit error. For example, to achieve a $P_b = 10^{-5}$, the required SNR per bit is a little more than 12 dB for $M = 2$, but if M is increased to 64 signal waveforms ($k = 6$ bits/symbol), the required SNR per bit is approximately 6 dB. Thus, a savings of over 6 dB (a factor-of-four reduction) is realized in transmitter power (or energy) required to achieve a $P_b = 10^{-5}$ by increasing M from $M = 2$ to $M = 64$.

What is the minimum required $\mathscr{E}_b/N_0$ to achieve an arbitrarily small probability of error as $M \to \infty$? This question is answered below.

A Union Bound on the Probability of Error Let us investigate the effect of increasing M on the probability of error for orthogonal signals. To simplify the mathematical development, we first derive an upper bound on the probability of a symbol error that is much simpler than the exact form given in (5-2-21).

Recall that the probability of error for binary orthogonal signals is given by (5-2-11). Now, if we view the detector for M orthogonal signals as one that makes $M - 1$ binary decisions between the correlator output $C(\mathbf{r}, \mathbf{s}_1)$ that contains the signal and the other $M - 1$ correlator outputs $C(\mathbf{r}, \mathbf{s}_m)$, $m = 2, 3, \ldots, M$, the probability of error is upper-bounded by the *union bound* of the $M - 1$ events. That is, if E_i represents the event that $C(\mathbf{r}, \mathbf{s}_i) > C(\mathbf{r}, \mathbf{s}_1)$ for $i \neq 1$ then we have $P_M = P(\bigcup_{i=1}^{n} E_i) \leq \sum_{i=1}^{n} P(E_i)$. Hence,

$$P_M \leq (M-1)P_2 = (M-1)Q(\sqrt{\mathscr{E}_s/N_0}) < MQ(\sqrt{\mathscr{E}_s/N_0}) \qquad (5\text{-}2\text{-}25)$$

This bound can be simplified further by upper-bounding $Q(\sqrt{\mathscr{E}_s/N_0})$. We have

$$Q(\sqrt{\mathscr{E}_s/N_0}) < e^{-\mathscr{E}_s/2N_0} \tag{5-2-26}$$

Thus,

$$\begin{aligned} P_M &< Me^{-\mathscr{E}_s/2N_0} = 2^k e^{-k\mathscr{E}_b/2N_0} \\ P_M &< e^{-k(\mathscr{E}_b/N_0 - 2\ln 2)/2} \end{aligned} \tag{5-2-27}$$

As $k \to \infty$, or equivalently, as $M \to \infty$, the probability of error approaches zero exponentially, provided that $\mathscr{E}_b/N_0$ is greater than $2 \ln 2$, i.e.,

$$\frac{\mathscr{E}_b}{N_0} > 2 \ln 2 = 1.39 \quad (1.42 \text{ dB}) \tag{5-2-28}$$

The simple upper bound on the probability of error given by (5-2-27) implies that, as long as SNR > 1.42 dB, we can achieve an arbitrarily low P_M. However, this union bound is not a very tight upper bound at a sufficiently low SNR due to the fact that the upper bound for the Q function in (5-2-26) is loose. In fact, by more elaborate bounding techniques, it is shown in Chapter 7 that the upper bound in (5-2-27) is sufficiently tight for $\mathscr{E}_b/N_0 > 4 \ln 2$. For $\mathscr{E}_b/N_0 < 4 \ln 2$, a tighter upper bound on P_M is

$$P_M < 2e^{-k(\sqrt{\mathscr{E}_b/N_0} - \sqrt{\ln 2})^2} \tag{5-2-29}$$

Consequently, $P_M \to 0$ as $k \to \infty$, provided that

$$\frac{\mathscr{E}_b}{N_0} > \ln 2 = 0.693 \quad (-1.6 \text{ dB}) \tag{5-2-30}$$

Hence, -1.6 dB is the minimum required SNR per bit to achieve an arbitrarily small probability of error in the limit as $k \to \infty$ ($M \to \infty$). This minimum SNR per bit (-1.6 dB) is called the *Shannon limit* for an additive white Gaussian noise channel.

5-2-3 Probability of Error for M-ary Biorthogonal Signals

As indicated in Section 4-3, a set of $M = 2^k$ biorthogonal signals are constructed from $\frac{1}{2}M$ orthogonal signals by including the negatives of the orthogonal signals. Thus, we achieve a reduction in the complexity of the demodulator for the biorthogonal signals relative to that for orthogonal signals, since the former is implemented with $\frac{1}{2}M$ cross-correlators or matched filters, whereas the latter requires M matched filters or cross-correlators.

To evaluate the probability of error for the optimum detector, let us assume that the signal $s_1(t)$ corresponding to the vector $\mathbf{s}_1 = [\sqrt{\mathscr{E}_s}\ 0\ 0\ \ldots\ 0]$ was transmitted. Then, the received signal vector is

$$\mathbf{r} = [\sqrt{\mathscr{E}_s} + n_1 \quad n_2 \quad \ldots \quad n_{M/2}] \tag{5-2-31}$$

where the $\{n_m\}$ are zero-mean, mutually statistically independent and identically distributed gaussian random variables with variance $\sigma_n^2 = \frac{1}{2}N_0$. The

magnitude of the cross-correlators

$$C(\mathbf{r}, \mathbf{s}_m) = \mathbf{r} \cdot \mathbf{s}_m = \sum_{k=1}^{M/2} r_k s_{mk}, \qquad m = 1, 2, \ldots, \tfrac{1}{2}M \tag{5-2-32}$$

while the sign of this largest term is used to decide whether $s_m(t)$ or $-s_m(t)$ was transmitted. According to this decision rule, the probability of a correct decision is equal to the probability that $r_1 = \sqrt{\mathcal{E}_s} + n_1 > 0$ and r_1 exceeds $|r_m| = |n_m|$ for $m = 2, 3, \ldots, \frac{1}{2}M$. But

is similar to that for orthogonal signals (see Fig. 5-2-5). However, in this case, the probability of error for $M = 4$ is greater than that for $M = 2$. This is due to the fact that we have plotted the symbol error probability P_M in Fig. 5-2-6. If we plotted the equivalent bit error probability, we should find that the graphs for $M = 2$ and $M = 4$ coincide. As in the case of orthogonal signals, as $M \to \infty$ (or $k \to \infty$), the minimum required $\mathscr{E}_b/N_0$ to achieve arbitrarily small probability of error is -1.6 dB, the Shannon limit.

5-2-4 Probability of Error for Simplex Signals

Next we consider the probability of error for M simplex signals. Recall from Section 4-3 that simplex signals are a set of M equally correlated signals with mutual cross-correlation coefficient $\rho_{mn} = -1/(M-1)$. These signals have the same minimum separation of $\sqrt{2\mathscr{E}_s}$ between adjacent signal points in M-dimensional space as orthogonal signals. They achieve this mutual separation with a transmitted energy of $\mathscr{E}_s(M-1)/M$, which is less than that required for orthogonal signals by a factor of $(M-1)/M$. Consequently, the probability of error for simplex signals is identical to the probability of error for orthogonal signals, but this performance is achieved with a saving of

$$10 \log (1 - \rho) = 10 \log \frac{M}{M-1} \text{ dB} \tag{5-2-35}$$

in SNR. For $M = 2$, the saving is 3 db. However, as M is increased, the saving in SNR approaches 0 dB.

5-2-5 Probability of Error for M-ary Binary-Coded Signals

We have shown in Section 4-3 that binary-coded signal waveforms are represented by the signal vectors

$$\mathbf{s}_m = [s_{m1} \quad s_{m2} \quad \cdots \quad s_{mN}], \qquad m = 1, 2, \ldots, M$$

where $s_{mj} = \pm\sqrt{\mathscr{E}/N}$ for all m and j. N is the block length of the code, and is also the dimension of the M signal waveforms.

If $d_{\min}^{(e)}$ is the minimum euclidean distance of the M signal waveforms then the probability of a symbol error is upper-bounded as

$$\begin{aligned} P_m &< (M-1)P_b = (M-1)Q\left(\sqrt{\frac{(d_{\min}^{(e)})^2}{2N_0}}\right) \\ &< 2^k \exp\left[-\frac{(d_{\min}^{(e)})^2}{4N_0}\right] \end{aligned} \tag{5-2-36}$$

The value of the minimum euclidean distance will depend on the selection of the code words, i.e., the design of the code.

5-2-6 Probability of Error for M-ary PAM

Recall that M-ary PAM signals are represented geometrically as M one-dimensional signal points with value

$$s_m = \sqrt{\tfrac{1}{2}\mathscr{E}_g} A_m, \qquad m = 1, 2, \ldots, M \tag{5-2-37}$$

where $\mathscr{E}_g$ is the energy of the basic signal pulse $g(t)$. The amplitude values may be expressed as

$$A_m = (2m - 1 - M)d, \qquad m = 1, 2, \ldots, M \tag{5-3-38}$$

where the euclidean distance between adjacent signal points is $d\sqrt{2\mathscr{E}_g}$.

$$\mathscr{E}_{av} = \frac{1}{M}\sum_{m=1}^{M} \mathscr{E}_m = \frac{d^2\mathscr{E}_g}{2M}\sum_{m=1}^{M}(2m - 1 - M)^2$$

$$= \frac{d^2\mathscr{E}_g}{2M}\left[\tfrac{1}{3}M(M^2 - 1)\right] = \tfrac{1}{6}(M^2 - 1)\, d^2\mathscr{E}_g \tag{5-2-39}$$

Equivalently, we may characterize these signals in terms of their average power, which is

$$P_{av} = \frac{\mathscr{E}_{av}}{T} = \tfrac{1}{6}(M^2 - 1)\frac{d^2\mathscr{E}_g}{T} \tag{5-2-40}$$

The average probability of error for M-ary PAM can be determined from the decision rule that maximizes the correlation metrics given by (5-1-44). Equivalently, the detector compares the demodulator output r with a set of $M - 1$ thresholds, which are placed at the midpoints of successive amplitude levels, as shown in Fig. 5-2-7. Thus, a decision is made in favor of the amplitude level that is closest to r.

The placing of the thresholds as shown in Fig. 5-2-7 helps in evaluating the probability of error. We note that if the mth amplitude level is transmitted, the demodulator output is

$$r = s_m + n = \sqrt{\tfrac{1}{2}\mathscr{E}_g} A_m + n \tag{5-2-41}$$

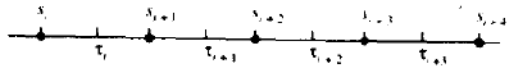


FIGURE 5-2-7 Placement of thresholds at midpoints of successive amplitude levels.

where the noise variable n has zero mean and variance $\sigma_n^2 = \frac{1}{2}N_0$. On the basis that all amplitude levels are equally likely a priori, the average probability of a symbol error is simply the probability that the noise variable n exceeds in magnitude one-half of the distance between levels. However, when either one of the two outside levels $\pm(M-1)$ is transmitted, an error can occur in one direction only. Thus, we have

$$\begin{aligned} P_M &= \frac{M-1}{M} P(|r - s_m| > d\sqrt{\tfrac{1}{2}\mathscr{E}_g}) \\ &= \frac{M-1}{M} \frac{2}{\sqrt{\pi N_0}} \int_{d\sqrt{\mathscr{E}_g/2}}^{\infty} e^{-x^2/N_0}\, dx \\ &= \frac{M-1}{M} \frac{2}{\sqrt{2\pi}} \int_{\sqrt{d^2\mathscr{E}_g/N_0}}^{\infty} e^{-x^2/2}\, dx \\ &= \frac{2(M-1)}{M} Q\left(\sqrt{\frac{d^2\mathscr{E}_g}{N_0}}\right) \end{aligned} \tag{5-2-42}$$

The error probability in (5-2-42) can also be expressed in terms of the average transmitted power. From (5-2-40), we note that

$$d^2\mathscr{E}_g = \frac{6}{M^2-1} P_{\text{av}} T \tag{5-2-43}$$

By substituting for $d^2\mathscr{E}_g$ in (5-2-42), we obtain the average probability of a symbol error for PAM in terms of the average power as

$$P_M = \frac{2(M-1)}{M} Q\left(\sqrt{\frac{6P_{\text{av}}T}{(M^2-1)N_0}}\right) \tag{5-2-44}$$

or, equivalently,

$$P_M = \frac{2(M-1)}{M} Q\left(\sqrt{\frac{6\mathscr{E}_{\text{av}}}{(M^2-1)N_0}}\right) \tag{5-2-45}$$

where $\mathscr{E}_{\text{av}} = P_{\text{av}}T$ is the average energy.

In plotting the probability of a symbol error for M-ary signals such as M-ary PAM, it is customary to use the SNR per bit as the basic parameter. Since $T = kT_b$ and $k = \log_2 M$, (5-2-45) may be expressed as

$$P_M = \frac{2(M-1)}{M} Q\left(\sqrt{\frac{(6\log_2 M)\mathscr{E}_{b\,\text{av}}}{(M^2-1)N_0}}\right) \tag{5-2-46}$$

where $\mathscr{E}_{b\text{av}} = P_{\text{av}}T_b$ is the average bit energy and $\mathscr{E}_{b\text{av}}/N_0$ is the average SNR

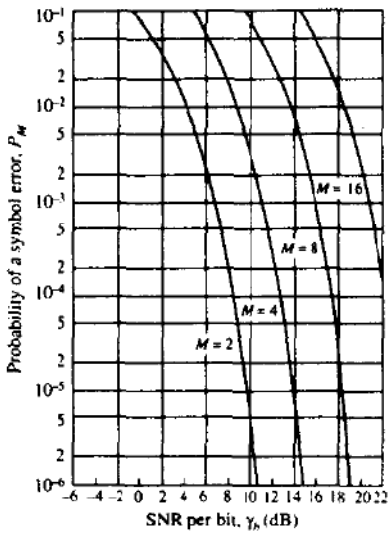


FIGURE 5-2-8 Probability of a symbol error for PAM.

per bit. Figure 5-2-8 illustrates the probability of a symbol error as a function of $10\log_{10} \mathscr{E}_{b\,av}/N_0$, with M as a parameter. Note that the case $M = 2$ corresponds to the error probability for binary antipodal signals. Also observe that the SNR per bit increases by over 4 dB for every factor-of-two increase in M. For large M, the additional SNR per bit required to increase M by a factor of two approaches 6 dB.

5-2-7 Probability of Error For M-ary PSK

Recall from Section 4-3 that digital phase-modulated signal waveforms may be expressed as

$$s_m(t) = g(t)\cos\left[2\pi f_c t + \frac{2\pi}{M}(m-1)\right], \quad 1 \leq m \leq M, \quad 0 \leq t \leq T \tag{5-2-47}$$

and have the vector representation

$$\mathbf{s}_m = \left[\sqrt{\mathscr{E}_s}\cos\frac{2\pi}{M}(m-1) \quad \sqrt{\mathscr{E}_s}\sin\frac{2\pi}{M}(m-1)\right] \tag{5-2-48}$$

where $\mathscr{E}_s = \frac{1}{2}\mathscr{E}_g$ is the energy in each of the waveforms and $g(t)$ is the pulse shape of the transmitted signal. Since the signal waveforms have equal energy, the optimum detector for the AWGN channel given by (5-1-44) computes the correlation metrics

$$C(\mathbf{r}, \mathbf{s}_m) = \mathbf{r}\cdot\mathbf{s}_m, \quad m = 1, 2, \ldots, M \tag{5-2-49}$$

In other words, the received signal vector $\mathbf{r} = [r_1 \quad r_2]$ is projected onto each of

the M possible signal vectors and a decision is made in favor of the signal with the largest projection.

The correlation detector described above is equivalent to a phase detector that computes the phase of the received signal from $\mathbf{r}$ and selects the signal vector $\mathbf{s}_m$ whose phase is closest to $\mathbf{r}$. Since the phase of $\mathbf{r}$ is

$$\Theta_r = \tan^{-1}\frac{r_2}{r_1} \tag{5-2-50}$$

we will determine the pdf of Θ_r, from which we shall compute the probability of error.

Let us consider the case in which the transmitted signal phase is $\Theta_r = 0$, corresponding to the signal $s_1(t)$. Hence, the transmitted signal vector is

$$s_0 = [\sqrt{\mathscr{E}_s} \quad 0] \tag{5-2-51}$$

and the received signal vector has components

$$\begin{aligned} r_1 &= \sqrt{\mathscr{E}_s} + n_1 \\ r_2 &= n_2 \end{aligned} \tag{5-2-52}$$

Because n_1 and n_2 are jointly gaussian random variables, it follows that r_1 and r_2 are jointly gaussian random variables with $E(r_1) = \sqrt{\mathscr{E}_s}$, $E(r_2) = 0$, and $\sigma_{r_1}^2 = \sigma_{r_2}^2 = \frac{1}{2}N_0 = \sigma_r^2$. Consequently,

$$p_r(r_1, r_2) = \frac{1}{2\pi\sigma_r^2}\exp\left[-\frac{(r_1 - \sqrt{\mathscr{E}_s})^2 + r_2^2}{2\sigma_r^2}\right] \tag{5-2-53}$$

The pdf of the phase Θ_r is obtained by a change in variables from (r_1, r_2) to

$$\begin{aligned} V &= \sqrt{r_1^2 + r_2^2} \\ \Theta_r &= \tan^{-1}(r_2/r_1) \end{aligned} \tag{5-2-54}$$

This yields the joint pdf

$$p_{V,\Theta_r}(V, \Theta_r) = \frac{V}{2\pi\sigma_r^2}\exp\left(-\frac{V^2 + \mathscr{E}_s - 2\sqrt{\mathscr{E}_s}\,V\cos\Theta_r}{2\sigma_r^2}\right)$$

Integration of $p_{V,\Theta_r}(V, \Theta_r)$ over the range of V yields $p_{\Theta_r}(\Theta_r)$. That is,

$$\begin{aligned} p_{\Theta_r}(\Theta_r) &= \int_0^\infty p_{V,\Theta_r}(V, \Theta_r)\,dV \\ &= \frac{1}{2\pi}e^{-2\gamma_s\sin^2\Theta_r}\int_0^\infty Ve^{-(V-\sqrt{4\gamma_s}\cos\Theta_r)^2/2}\,dV \end{aligned} \tag{5-2-55}$$

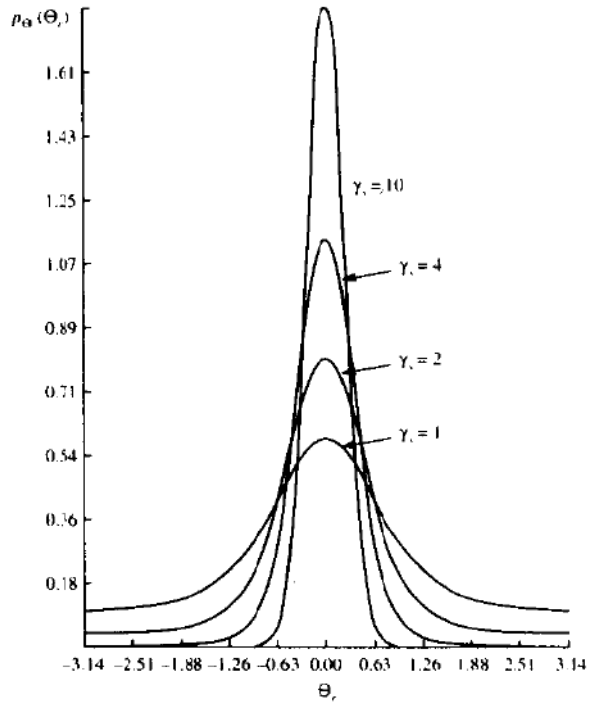


FIGURE 5-2-9 Probability density function $p_{\Theta_r}(\Theta_r)$ for $\gamma_s = 1, 2, 4$ and 10.

where for convenience, we have defined the symbol SNR as $\gamma_s = \mathscr{E}_s/N_0$. Figure 5-2-9 illustrates $f_{\Theta_r}(\Theta_r)$ for several values of the SNR parameter γ_s when the transmitted phase is zero. Note that $f_{\Theta_r}(\Theta_r)$ becomes narrower and more peaked about $\Theta_r = 0$ as the SNR γ_s increases.

When $s_1(t)$ is transmitted, a decision error is made if the noise causes the phase to fall outside the range $-\pi/M \leq \Theta_r \leq \pi/M$. Hence, the probability of a symbol error is

$$P_M = 1 - \int_{-\pi/M}^{\pi/M} p_{\Theta_r}(\Theta_r)\, d\Theta_r \tag{5-2-56}$$

In general, the integral of $p_{\Theta_r}(\Theta)$ does not reduce to a simple form and must be evaluated numerically, except for $M = 2$ and $M = 4$.

For binary phase modulation, the two signals $s_1(t)$ and $s_2(t)$ are antipodal, and, hence, the error probability is

$$P_2 = Q\left(\sqrt{\frac{2\mathscr{E}_b}{N_0}}\right) \tag{5-2-57}$$

When $M = 4$, we have in effect two binary phase-modulation signals in phase

quadrature. Since there is no crosstalk or interference between the signals on the two quadrature carriers, the bit error probability is identical to that in (5-2-57). On the other hand, the symbol error probability for $M = 4$ is determined by noting that

$$P_c = (1 - P_2)^2 = \left[1 - Q\left(\sqrt{\frac{2\mathscr{E}_b}{N_0}}\right)\right]^2 \tag{5-2-58}$$

where P_c is the probability of a correct decision for the 2-bit symbol. The result (5-2-58) follows from the statistical independence of the noise on the quadrature carriers. Therefore, the symbol error probability for $M = 4$ is

$$\begin{aligned} P_4 &= 1 - P_c \\ &= 2Q\left(\sqrt{\frac{2\mathscr{E}_b}{N_0}}\right)\left[1 - \tfrac{1}{2}Q\left(\sqrt{\frac{2\mathscr{E}_b}{N_0}}\right)\right] \end{aligned} \tag{5-2-59}$$

For $M > 4$, the symbol error probability P_M is obtained by numerically integrating (5-2-55). Figure 5-2-10 illustrates this error probability as a function of the SNR per bit for $M = 2$, 4, 8, 16, and 32. The graphs clearly illustrate the penalty in SNR per bit as M increases beyond $M = 4$. For example, at $P_M = 10^{-5}$, the difference between $M = 4$ and $M = 8$ is approximately 4 dB, and the difference between $M = 8$ and $M = 16$ is approximately 5 dB. For large values of M, doubling the number of phases requires an additional 6 dB/bit to achieve the same performance.

An approximation to the error probability for large values of M and for

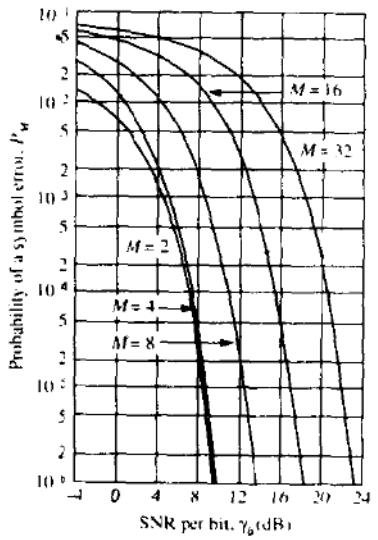


FIGURE 5-2-10 Probability of a symbol error for PSK signals.

large SNR may be obtained by first approximating $p_{\Theta_r}(\Theta)$. For $\mathscr{E}_s/N_0 \gg 1$ and $|\Theta_r| \leq \frac{1}{2}\pi$, $p_{\Theta_r}(\Theta_r)$ is well approximated as

$$p_{\Theta_r}(\Theta_r) \approx \sqrt{\frac{2\gamma_s}{\pi}} \cos \Theta_r e^{-2\gamma_s \sin^2 \Theta_r} \tag{5-2-60}$$

By substituting for $p_{\Theta_r}(\Theta_r)$ in (5-2-56) and performing the change in variable from Θ_r to $u = \sqrt{2\gamma_s} \sin \Theta_r$, we find that

$$\begin{aligned} P_M &\approx 1 - \int_{-\pi/M}^{\pi/M} \sqrt{\frac{2\gamma_s}{\pi}} \cos \Theta_r e^{-2\gamma_s \sin^2 \Theta_r} d\Theta_r \\ &\approx \frac{2}{\sqrt{\pi}} \int_{\sqrt{2\gamma_s} \sin(\pi/M)}^{\infty} e^{-u^2/2} du \\ &= 2Q\left(\sqrt{2\gamma_s} \sin \frac{\pi}{M}\right) = 2Q\left(\sqrt{2k\gamma_b} \sin \frac{\pi}{M}\right) \end{aligned} \tag{5-2-61}$$

where $k = \log_2 M$ and $\gamma_s = k\gamma_b$. Note that this approximation to the error probability is good for all values of M. For example, when $M = 2$ and $M = 4$, we have $P_2 = P_4 = 2Q(\sqrt{2\gamma_s})$, which compares favorably (a factor-of-two difference) with the exact probability given by (5-2-57).

The equivalent bit error probability for M-ary PSK is rather tedious to derive due to its dependence on the mapping of k-bit symbols into the corresponding signal phases. When a Gray code is used in the mapping, two k-bit symbols corresponding to adjacent signal phases differ in only a single bit. Since the most probable errors due to noise result in the erroneous selection of an adjacent phase to the true phase, most k-bit symbol errors contain only a single-bit error. Hence, the equivalent bit error probability for M-ary PSK is well approximated as

$$P_b \approx \frac{1}{k} P_M \tag{5-2-62}$$

Our treatment of the demodulation of PSK signals assumed that the demodulator had a perfect estimate of the carrier phase available. In practice, however, the carrier phase is extracted from the received signal by performing some nonlinear operation that introduces a phase ambiguity. For example, in binary PSK, the signal is often squared in order to remove the modulation, and the double-frequency component that is generated is filtered and divided by 2 in frequency in order to extract an estimate of the carrier frequency and phase ϕ. These operations result in a phase ambiguity of 180° in the carrier phase. Similarly, in four-phase PSK, the received signal is raised to the fourth power in order to remove the digital modulation, and the resulting fourth harmonic of the carrier frequency is filtered and divided by 4 in order to extract the carrier component. These operations yield a carrier frequency component containing the estimate of the carrier phase ϕ, but there are phase ambiguities of ±90° and 180° in the phase estimate. Consequently, we do not have an absolute estimate of the carrier phase for demodulation.

The phase ambiguity problem resulting from the estimation of the carrier phase ϕ can be overcome by encoding the information in phase differences between successive signal transmissions as opposed to absolute phase encoding. For example, in binary PSK, the information bit 1 may be transmitted by shifting the phase of the carrier by 180° relative to the previous carrier phase, while the information bit 0 is transmitted by a zero phase shift relative to the phase in the previous signaling interval. In four-phase PSK, the relative phase shifts between successive intervals are 0, 90°, 180°, and −90°, corresponding to the information bits 00, 01, 11, and 10, respectively. The generalization to $M > 4$ phases is straightforward. The PSK signals resulting from the encoding process are said to be *differentially encoded*. The encoding is performed by a relatively simple logic circuit preceding the modulator.

Demodulation of the differentially encoded PSK signal is performed as described above, by ignoring the phase ambiguities. Thus, the received signal is demodulated and detected to one of the M possible transmitted phases in each signaling interval. Following the detector is a relatively simple phase comparator that compares the phases of the demodulated signal over two consecutive intervals in order to extract the information.

Coherent demodulation of differently encoded PSK results in a higher probability of error than the error probability derived for absolute phase encoding. With differentially encoded PSK, an error in the demodulated phase of the signal in any given interval will usually result in decoding errors over two consecutive signaling intervals. This is especially the case for error probabilities below 0.1. Therefore, the probability of error in differentially encoded M-ary PSK is approximately twice the probability of error for M-ary PSK with absolute phase encoding. However, this factor-of-two increase in the error probability translates into a relatively small loss in SNR.

5-2-8 Differential PSK (DPSK) and its Performance

A differentially encoded phase-modulated signal also allows another type of demodulation that does not require the estimation of the carrier phase.† Instead, the received signal in any given signaling interval is compared to the phase of the received signal from the preceding signaling interval. To elaborate, suppose that we demodulate the differentially encoded signal by multiplying $r(t)$ by $\cos 2\pi f_c t$ and $\sin 2\pi f_c t$ and integrating the two products over the interval T. At the kth signaling interval, the demodulator output is

$$\mathbf{r}_k = [\sqrt{\mathscr{E}_s}\cos(\theta_k - \phi) + n_{k1} \quad \sqrt{\mathscr{E}_s}\sin(\theta_k - \phi) + n_{k2}]$$

or, equivalently,

$$r_k = \sqrt{\mathscr{E}_s}e^{j(\theta_k - \phi)} + n_k \tag{5-2-63}$$

† Because no phase estimation is required, DPSK is often considered to be a noncoherent communication technique. We take the view that DPSK represents a form of digital phase modulation in the extreme case where the phase estimate is derived only from the previous symbol interval.

where θ_k is the phase angle of the transmitted signal at the kth signaling interval, ϕ is the carrier phase, and $n_k = n_{k_1} + jn_{k_2}$ is the noise vector. Similarly, the received signal vector at the output of the demodulator in the preceding signaling interval is

$$r_{k-1} = \sqrt{\mathscr{E}_s} e^{j(\theta_{k-1} - \phi)} + n_{k-1} \tag{5-2-64}$$

The decision variable for the phase detector is the phase difference between these two complex numbers. Equivalently, we can project r_k onto r_{k-1} and use the phase of the resulting complex number; that is,

$$r_k r_{k-1}^* = \mathscr{E}_s e^{j(\theta_k - \theta_{k-1})} + \sqrt{\mathscr{E}_s} e^{j(\theta_k - \phi)} n_{k-1}^* + \sqrt{\mathscr{E}_s} e^{-j(\theta_{k-1} - \phi)} n_k + n_k n_{k-1}^* \tag{5-2-65}$$

which, in the absence of noise, yields the phase difference $\theta_k - \theta_{k-1}$. Thus, the mean value of $r_k r_{k-1}^*$ is independent of the carrier phase. Differentially encoded PSK signaling that is demodulated and detected as described above is called *differential PSK* (DPSK).

The demodulation and detection of DSPK using matched filters is illustrated in Figure 5-2-11. If the pulse $g(t)$ is rectangular, the matched filters may be replaced by integrate-and-dump filters.

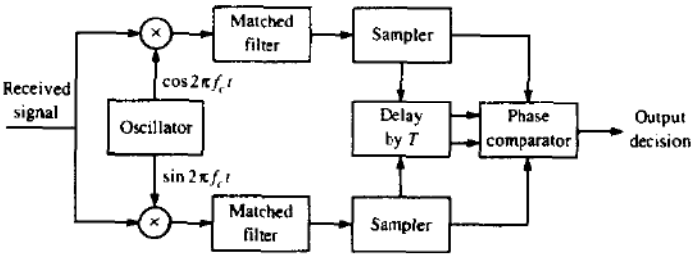
Let us now consider the evaluation of the error probability performance of a DPSK demodulator and detector. The derivation of the exact value of the probability of error for M-ary DPSK is extremely difficult, except for $M = 2$. The major difficulty is encountered in the determination of the pdf for the phase of the random variable $r_k r_{k-1}^*$, given by (5-2-65). However, an approximation to the performance of DPSK is easily obtained, as we now demonstrate.

Without loss of generality, suppose the phase difference $\theta_k - \theta_{k-1} = 0$. Furthermore, the exponential factors $e^{-j(\theta_{k-1} - \phi)}$ and $e^{j(\theta_k - \phi)}$ in (5-2-65) can be absorbed into the gaussian noise components n_{k-1} and n_k, without changing their statistical properties. Therefore, $r_k r_{k-1}^*$ in (5-2-65) can be expressed as

$$r_k r_{k-1}^* = \mathscr{E}_s + \sqrt{\mathscr{E}_s}(n_k + n_{k-1}^*) + n_k n_{k-1}^* \tag{5-2-66}$$

The complication in determining the pdf of the phase is the term $n_k n_{k-1}^*$. However, at SNRs of practial interest, the term $n_k n_{k-1}^*$ is small relative to the dominant noise term $\sqrt{\mathscr{E}_s}(n_k + n_{k-1}^*)$. If we neglect the term $n_k n_{k-1}^*$ and we

FIGURE 5-2-11 Block diagram of DPSK demodulator.



also normalize $r_k r_{k-1}^*$ by dividing through by $\sqrt{\mathscr{E}_s}$, the new set of decision metrics becomes

$$\begin{aligned} x &= \sqrt{\mathscr{E}_s} + \text{Re}\,(n_k + n_{k-1}^*) \\ y &= \text{Im}\,(n_k + n_{k-1}^*) \end{aligned} \tag{5-2-67}$$

The variables x and y are uncorrelated gaussian random variables with identical variances $\sigma_n^2 = N_0$. The phase is

$$\Theta_r = \tan^{-1}\frac{y}{x} \tag{5-2-68}$$

At this stage, we have a problem that is identical to the one we solved previously for phase-coherent demodulation. The only difference is that the noise variance is now twice as large as in the case of PSK. Thus we conclude that the performance of DPSK is 3 dB poorer than that for PSK. This result is relatively good for $M \geq 4$, but it is pessimistic for $M = 2$ in the sense that the loss in binary DPSK relative to binary PSK is less than 3 db at large SNR. This is demonstrated below.

In binary DPSK, the two possible transmitted phase differences are 0 and π rad. As a consequence, only the real part of $r_k r_{k-1}^*$ is needed for recovering the information. Using (5-2-67), we express the real part as

$$\text{Re}\,(r_k r_{k-1}^*) = \tfrac{1}{2}(r_k r_{k-1}^* + r_k^* r_{k-1})$$

Because the phase difference between the two successive signaling intervals is zero, an error is made if $\text{Re}\,(r_k r_{k-1}^*) < 0$. The probability that $r_k r_{k-1}^* + r_k^* r_{k-1} < 0$ is a special case of a derivation, given in Appendix B concerned with the probability that a general quadratic form in complex-valued gaussian random variables is less than zero. The general form for this probability is given by (B-21) of Appendix B, and it depends entirely on the first and second moments of the complex-valued gaussian random variables r_k and r_{k-1}. Upon evaluating the moments and the parameters that are functions of the moments, we obtain the probability of error for binary DPSK in the form

$$P_b = \tfrac{1}{2}e^{-\mathscr{E}_b/N_0} \tag{5-2-69}$$

where $\mathscr{E}_b/N_0$ is the SNR per bit.

The graph is shown in Fig. 5-2-12. Also shown in that illustration is the probability of error for binary, coherent PSK. We observe that at error probabilities of $P_b \leq 10^{-3}$ the difference in SNR between binary PSK and binary DPSK is less than 3 dB. In fact, at $P_b \leq 10^{-5}$, the difference in SNR is less than 1 dB.

The probability of a binary digit error for four-phase DPSK with Gray coding can be expressed in terms of well-known functions, but its derivation is quite involved. We simply state the result at this point and refer the interested reader to Appendix C for the details of derivation. It is expressed in the form

$$P_b = Q_1(a, b) - \tfrac{1}{2}I_0(ab)\exp\left[-\tfrac{1}{2}(a^2 + b^2)\right] \tag{5-2-70}$$

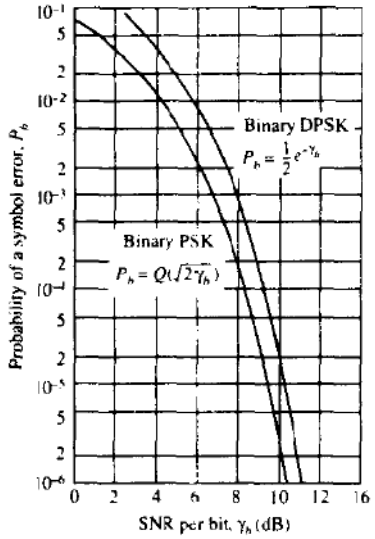


FIGURE 5-2-12 Probability of error for binary PSK and DPSK.

where $Q_1(a, b)$ is the Markum Q function defined by (2-1-122) and (2-1-123), $I_0(x)$ is the modified Bessel function of order zero, defined by (2-1-120), and the parameters a and b are defined as

$$a = \sqrt{2\gamma_b(1 - \sqrt{\tfrac{1}{2}})}$$
$$b = \sqrt{2\gamma_b(1 + \sqrt{\tfrac{1}{2}})} \tag{5-2-71}$$

Figure 5-2-13 illustrates the probability of a binary digit error for two- and

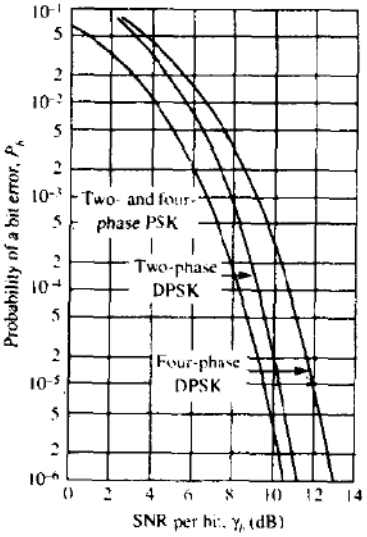


FIGURE 5-2-13 Probability of bit error for binary and four-phase PSK and DPSK.

four-phase DPSK and coherent PSK signaling obtained from evaluating the exact formulas derived in this section. Since binary DPSK is only slightly inferior to binary PSK at large SNR, and DPSK does not require an elaborate method for estimating the carrier phase, it is often used in digital communications systems. On the other hand, four-phase DPSK is approximately 2.3 dB poorer in performance than four-phase PSK at large SNR. Consequently the choice between these two four-phase systems is not as clear cut. One must weigh the 2.3 dB loss against the reduction in implementation complexity.

5-2-9 Probability of Error for QAM

Recall from Section 4-3 that QAM signal waveforms may be expressed as

$$s_m(t) = A_{mc}g(t)\cos 2\pi f_c t - A_{ms}g(t)\sin 2\pi f_c t, \quad 0 \leq t \leq T \tag{5-2-72}$$

where A_{mc} and A_{ms} are the information-bearing signal amplitudes of the quadrature carriers and $g(t)$ is the signal pulse. The vector representation of these waveforms is

$$\mathbf{s}_m = [A_{mc}\sqrt{\tfrac{1}{2}\mathscr{E}_g} \quad A_{ms}\sqrt{\tfrac{1}{2}\mathscr{E}_g}] \tag{5-2-73}$$

To determine the probability of error for QAM, we must specify the signal point constellation. We begin with QAM signal sets that have $M = 4$ points. Figure 5-2-14 illustrates two four-point signal sets. The first is a four-phase modulated signal and the second is a QAM signal with two amplitude levels, labeled A_1 and A_2, and four phases. Because the probability of error is dominated by the minimum distance between pairs of signal points, let us impose the condition that $d_{\min}^{(e)} = 2A$ for both signal constellations and let us evaluate the average transmitter power, based on the premise that all signal points are equally probable. For the four-phase signal, we have

$$P_{\text{av}} = \tfrac{1}{4}(4)2A^2 = 2A^2 \tag{5-2-74}$$

For the two-amplitude, four-phase QAM, we place the points on circles of radii A and $\sqrt{3}$A. Thus, $d_{\min}^{(e)} = 2A$, and

$$P_{\text{av}} = \tfrac{1}{4}[2(3A^2) + 2A^2] = 2A^2 \tag{5-2-75}$$

which is the same average power as the $M = 4$-phase signal constellation. Hence, for all practical purposes, the error rate performance of the two signal

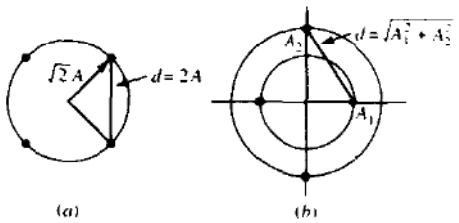


FIGURE 5-2-14 Two four-point signal constellations.

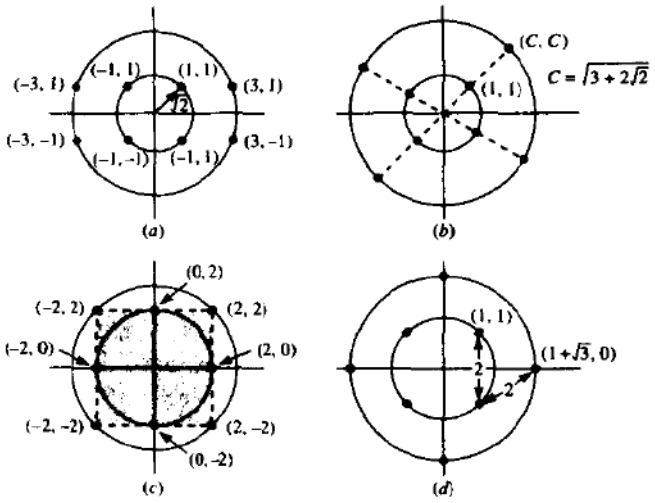


FIGURE 5-2-15 Four eight-point QAM signal constellations.

sets is the same. In other words, there is no advantage of the two-amplitude QAM signal set over $M = 4$-phase modulation.

Next, let us consider $M = 8$ QAM. In this case, there are many possible signal constellations. We shall consider the four signal constellations shown in Fig. 5-2-15, all of which consist of two amplitudes and have a minimum distance between signal points of $2A$. The coordinates (A_{mc}, A_{ms}) for each signal point, normalized by A, are given in the figure. Assuming that the signal points are equally probable, the average transmitted signal power is

$$P_{\text{av}} = \frac{1}{M}\sum_{m=1}^{M}(A_{mc}^2 + A_{ms}^2)$$

$$= \frac{A^2}{M}\sum_{m=1}^{M}(a_{mc}^2 + a_{mc}^2) \qquad (5\text{-}2\text{-}76)$$

where (a_{mc}, a_{ms}) are the coordinates of the signal points, normalized by A.

The two signal sets (a) and (c) in Fig. 5-2-15 contain signal points that fall on a rectangular grid and have $P_{\text{av}} = 6A^2$. The signal set (b) requires an average transmitted power $P_{\text{av}} = 6.83A^2$, and (d) requires $P_{\text{av}} = 4.73A^2$. Therefore, the fourth signal set requires approximately 1 dB less power than the first two and 1.6 dB less power than the third to achieve the same probability of error. This signal constellation is known to be the best eight-point QAM constellation because it requires the least power for a given minimum distance between signal points.

For $M \geq 16$, there are many more possibilities for selecting the QAM signal points in the two-dimensional space. For example, we may choose a circular multiamplitude constellation for $M = 16$, as shown in Fig. 4-3-4. In this case,

the signal points at a given amplitude level are phase-rotated by $\frac{1}{4}\pi$ relative to the signal points at adjacent amplitude levels. This 16-QAM constellation is a generalization of the optimum 8-QAM constellation. However, the circular 16-QAM constallation is not the best 16-point QAM signal constellation for the AWGN channel.

Rectangular QAM signal constellations have the distinct advantage of being easily generated as two PAM signals impressed on phase-quadrature carriers. In addition, they are easily demodulated. Although they are not the best M-ary QAM signal constellations for $M \geq 16$, the average transmitted power required to achieve a given minimum distance is only slightly greater than the average power required for the best M-ary QAM signal constellation. For these reasons, rectangular M-ary QAM signals are most frequently used in practice.

For rectangular signal constellations in which $M = 2^k$, where k is even, the QAM signal constellation is equivalent to two PAM signals on quadrature carriers, each having $\sqrt{M} = 2^{k/2}$ signal points. Since the signals in the phase-quadrature components can be perfectly separated at the demodulator, the probability of error for QAM is easily determined from the probability of error for PAM. Specifically, the probability of a correct decision for the M-ary QAM system is

$$P_c = (1 - P_{\sqrt{M}})^2 \tag{5-2-77}$$

where $P_{\sqrt{M}}$ is the probability of error of a $\sqrt{M}$ – ary PAM with one-half the average power in each quadrature signal of the equivalent QAM system. By appropriately modifying the probability of error for M-ary PAM, we obtain

$$P_{\sqrt{M}} = 2\left(1 - \frac{1}{\sqrt{M}}\right)Q\left(\sqrt{\frac{3}{M-1}\frac{\mathscr{E}_{av}}{N_0}}\right) \tag{5-2-78}$$

where $\mathscr{E}_{av}/N_0$ is the average SNR per symbol. Therefore, the probability of a symbol error for the M-ary QAM is

$$P_M = 1 - (1 - P_{\sqrt{M}})^2 \tag{5-2-79}$$

Note that this result is exact for $M = 2^k$ when k is even. On the other hand, when k is odd, there is no equivalent $\sqrt{M}$ – ary PAM system. This is no problem, however, because it is rather easy to determine the error rate for a rectangular signal set. If we employ the optimum detector that bases its decisions on the optimum distance metrics given by (5-1-43), it is relatively straightforward to show that the symbol error probability is tightly upper-bounded as

$$\begin{aligned} P_M &\leq 1 - \left[1 - 2Q\left(\sqrt{\frac{3\mathscr{E}_{av}}{(M-1)N_0}}\right)\right]^2 \\ &\leq 4Q\left(\sqrt{\frac{3k\mathscr{E}_{b\,av}}{(M-1)N_0}}\right) \end{aligned} \tag{5-2-80}$$

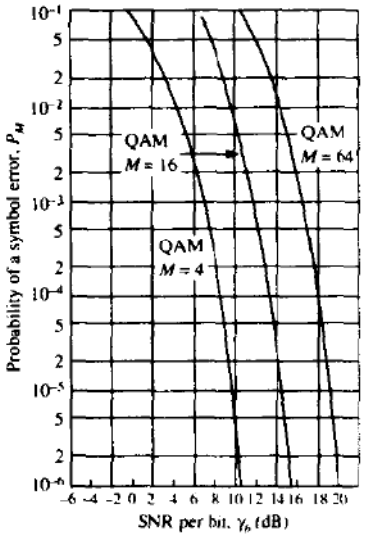


FIGURE 5-2-16 Probability of a symbol error for QAM.

for any $k \geq 1$, where $\mathscr{E}_{b\,av}/N_0$ is the average SNR per bit. The probability of a symbol error is plotted in Fig. 5-2-16 as a function of the average SNR per bit.

For non-rectangular QAM signal constellations, we may upper-bound the error probability by use of a union bound. An obvious upper bound is

$$P_M < (M-1)Q(\sqrt{[d_{\min}^{(\epsilon)}]^2/2N_0})$$

where $d_{\min}^{(\epsilon)}$ in the minimum euclidean distance between signal points. This bound may be loose when M is large. In such a case, we may approximate P_M by replacing $M-1$ by M_n, where M_n is the largest number of neighboring points that are at distance $d_{\min}^{(\epsilon)}$ from any constellation point.

It is interesting to compare the performance of QAM with that of PSK for any given signal size M, since both types of signals are two-dimensional. Recall that for M-ary PSK, the probability of a symbol error is approximated as

$$P_M \approx 2Q\left(\sqrt{2\gamma_s}\sin\frac{\pi}{M}\right) \tag{5-2-81}$$

where γ_s is the SNR per symbol. For M-ary QAM, we may use the expression (5-2-78). Since the error probability is dominated by the argument of the Q function, we may simply compare the arguments of Q for the two signal formats. Thus, the ratio of these two arguments is

$$\mathscr{R}_M = \frac{3/(M-1)}{2\sin^2(\pi/M)} \tag{5-2-82}$$

For example, when $M=4$, we have $\mathscr{R}_M = 1$. Hence, 4-PSK and 4-QAM yield comparable performance for the same SNR per symbol. On the other hand,

TABLE 5-2-1 SNR ADVANTAGE OF M-ARY QAM OVER M-ARY PSK

M	$10 \log_{10} \mathcal{R}_M$
8	1.65
16	4.20
32	7.02
64	9.95

when $M > 4$ we find that $\mathcal{R}_M > 1$, so that M-ary QAM yields better performance than M-ary PSK. Table 5-2-1 illustrates the SNR advantage of QAM over PSK for several values of M. For example, we observe that 32-QAM has a 7 dB SNR advantage over 32-PSK.

5-2-10 Comparison of Digital Modulation Methods

The digital modulation methods described in this chapter can be compared in a number of ways. For example, one can compare them on the basis of the SNR required to achieve a specified probability of error. However, such a comparison would not be very meaningful, unless it were made on the basis of some constraint, such as a fixed data rate of transmission or, equivalently, on the basis of a fixed bandwidth. With this goal in mind, let us consider the bandwidth requirements for several modulation methods.

For multiphase signals, the channel bandwidth required is simply the bandwidth of the equivalent lowpass signal pulse $g(t)$, which depends on its detailed characteristics. For our purposes, we assume that $g(t)$ is a pulse of duration T and that its bandwidth W is approximately equal to the reciprocal of T. Thus, $W = 1/T$ and, since $T = k/R = (\log_2 M)/R$, it follows that

$$W = \frac{R}{\log_2 M} \tag{5-2-83}$$

Therefore, as M is increased, the channel bandwidth required, when the bit rate R is fixed, decreases. The bandwidth efficiency is measured by the bit rate to bandwidth ratio, which is

$$\frac{R}{W} = \log_2 M \tag{5-2-84}$$

The bandwidth-efficient method for transmitting PAM is single-sideband. Then, the channel bandwidth required to transmit the signal is approximately equal to $1/2T$ and, since $T = k/R = (\log_2 M)/R$, it follows that

$$\frac{R}{W} = 2 \log_2 M \tag{5-2-85}$$

This is a factor of two better than PSK.

In the case of QAM, we have two orthogonal carriers, with each carrier having a PAM signal. Thus, we double the rate relative to PAM. However, the QAM signal must be transmitted via double sideband. Consequently, QAM and PAM have the same bandwidth efficiency when the bandwidth is referenced to the bandpass signal.

Orthogonal signals have totally different bandwidth requirements. If the $M = 2^k$ orthogonal signals are constructed by means of orthogonal carriers with minimum frequency separation of $1/2T$ for orthogonality, the bandwidth required for transmission of $k = \log_2 M$ information bits is

$$W = \frac{M}{2T} = \frac{M}{2(k/R)} = \frac{M}{2\log_2 M} R \tag{5-2-86}$$

In this case, the bandwidth increases as M increases. Similar relationships obtain for simplex and biorthogonal signals. In the case of biothogonal signals, the required bandwidth is one half of that for orthogonal signals.

A compact and meaningful comparison of these modulation methods is one based on the normalized data rate R/W (bits per second per hertz of bandwidth) versus the SNR per bit ($\mathscr{E}_b/N_0$) required to achieve a given error probability. Figure 5-2-17 illustrates the graph of R/W versus SNR per bit for PAM, QAM, PSK, and orthogonal signals, for the case in which the error probability is $P_M = 10^{-5}$. We observe that in the case of PAM, QAM, and PSK, increasing M results in a higher bit rate-to-bandwidth ratio R/W. However, the cost of achieving the higher data rate is an increase in the SNR per bit. Consequently, these modulation methods are appropriate for communication channels that are bandwidth limited, where we desire a bit rate-to-bandwidth ratio $R/W > 1$ and where there is sufficiently high SNR to support increases in M. Telephone channels and digital microwave radio channels are examples of such bandlimited channels.

In contrast, M-ary orthogonal signals yield a bit rate-to-bandwidth ratio of $R/W \leq 1$. As M increases, R/W decreases due to an increase in the required channel bandwidth. However, the SNR per bit required to achieve a given error probability (in this case, $P_M = 10^{-5}$) decreases as M increases. Consequently, M-ary orthogonal signals are appropriate for power-limited channels that have sufficiently large bandwidth to accommodate a large number of signals. In this case, as $M \to \infty$, the error probability can be made as small as desired, provided that $\mathscr{E}_b/N_0 > 0.693$ (-1.6 dB). This is the minimum SNR per bit required to achieve reliable transmission in the limit as the channel bandwidth $W \to \infty$ and the corresponding bit rate-to-bandwidth ratio $R/W \to 0$.

Also shown in Fig. 5-2-17 is the graph for the normalized capacity of the bandlimited AWGN channel, which is due to Shannon (1948). The ratio C/W, where C ($= R$) is the capacity in bits/s, represents the highest achievable bit rate-to-bandwidth ratio on this channel. Hence, it serves as the upper bound

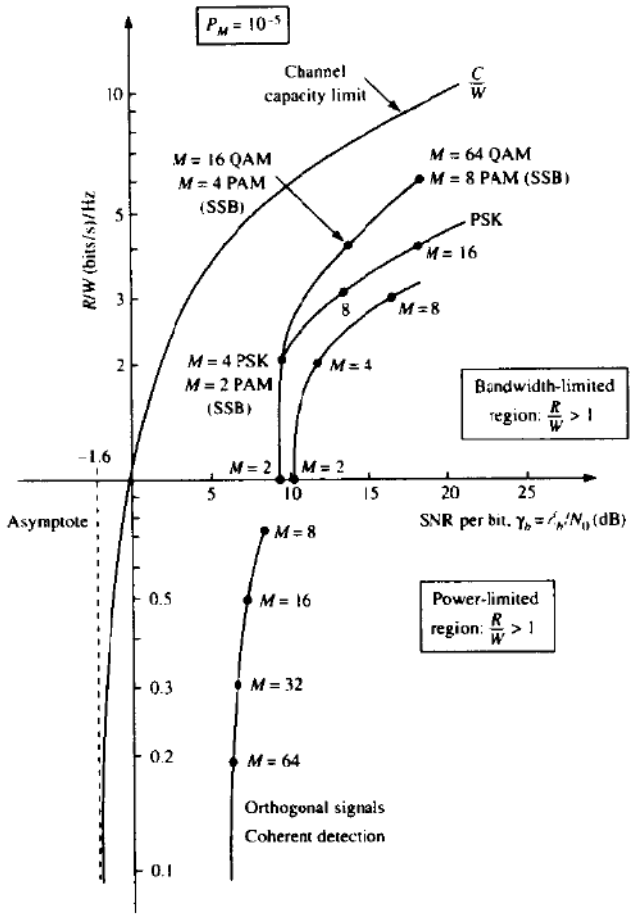


FIGURE 5-2-17 Comparison of several modulation methods at 10^{-5} symbol error probability.

on the bandwidth efficiency of any type of modulation. This bound is derived in Chapter 7 and discussed in greater detail there.

5-3 OPTIMUM RECEIVER FOR CPM SIGNALS

We recall from Section 4-3 that CPM is a modulation method with memory. The memory results from the continuity of the transmitted carrier phase from one signal interval to the next. The transmitted CPM signal may be expressed as

$$s(t) = \sqrt{\frac{2\mathcal{E}}{T}} \cos\left[2\pi f_c t + \phi(t; \mathbf{I})\right] \tag{5-3-1}$$

where $\phi(t; \mathbf{I})$ is the carrier phase. The filtered received signal for an additive gaussian noise channel is

$$r(t) = s(t) + n(t) \tag{5-3-2}$$

where

$$n(t) = n_c(t) \cos 2\pi f_c t - n_s(t) \sin 2\pi f_c t \tag{5-3-3}$$

5-3-1 Optimum Demodulation and Detection of CPM

The optimum receiver for this signal consists of a correlator followed by a maximum-likelihood sequence detector that searches the paths through the state trellis for the minimum euclidean distance path. The Viterbi algorithm is an efficient method for performing this search. Let us establish the general state trellis structure for CPM and then describe the metric computations.

Recall that the carrier phase for a CPM signal with a fixed modulation index h may be expressed as

$$\begin{aligned}\phi(t; \mathbf{I}) &= 2\pi h \sum_{k=-\infty}^{n} I_k q(t - kT) \\ &= \pi h \sum_{k=-\infty}^{n-L} I_k + 2\pi h \sum_{k=n-L+1}^{n} I_k q(t - kT) \\ &= \theta_n + \theta(t; \mathbf{I}), \qquad nT \leq t \leq (n+1)T\end{aligned} \tag{5-3-4}$$

where we have assumed that $q(t) = 0$ for $t < 0$, $q(t) = \frac{1}{2}$ for $t \geq LT$, and

$$q(t) = \int_0^t g(\tau)\, d\tau \tag{5-3-5}$$

The signal pulse $g(t) = 0$ for $t < 0$ and $t \geq LT$. For $L = 1$, we have a full response CPM, and for $L > 1$, where L is a positive integer, we have a partial response CPM signal.

Now, when h is rational, i.e., $h = m/p$ where m and p are relatively prime positive integers, the CPM scheme can be represented by a trellis. In this case, there are *p phase states*

$$\Theta_s = \left\{0, \frac{\pi m}{p}, \frac{2\pi m}{p}, \ldots, \frac{(p-1)\pi m}{p}\right\} \tag{5-3-6}$$

when m is even, and $2p$ phase states

$$\Theta_s = \left\{0, \frac{\pi m}{p}, \ldots, \frac{(2p-1)\pi m}{p}\right\} \tag{5-3-7}$$

when m is odd. If $L = 1$, these are the only states in the trellis. On the other hand, if $L > 1$, we have an additional number of states due to the partial

response character of the signal pulse $g(t)$. These additional states can be identified by expressing $\theta(t, \mathbf{I})$ given by (5-3-4) as

$$\theta(t; \mathbf{I}) = 2\pi h \sum_{k=n-L+1}^{n-1} I_k q(t - kT) + 2\pi h I_n q(t - nT) \tag{5-3-8}$$

The first term on the right-hand side of (5-3-8) depends on the information symbols $(I_{n-1}, I_{n-2}, \ldots, I_{n-L+1})$, which is called the *correlative state vector*, and represents the phase term corresponding to signal pulses that have not reached their final value. The second term in (5-3-8) represents the phase contribution due to the most recent symbol I_n. Hence, the state of the CPM signal (or the modulator) at time $t = nT$ may be expressed as the combined *phase state* and *correlative state*, denoted as

$$S_n = \{\theta_n, I_{n-1}, I_{n-2}, \ldots, I_{n-L+1}\} \tag{5-3-9}$$

for a partial response signal pulse of length LT, where $L > 1$. In this case, the number of states is

$$N_s = \begin{cases} pM^{L-1} & (\text{even } m) \\ 2pM^{L-1} & (\text{odd } m) \end{cases} \tag{5-3-10}$$

when $h = m/p$.

Now, suppose the state of the modulator at $t = nT$ is S_n. The effect of the new symbol in the time interval $nT \leq t \leq (n+1)T$ is to change the state from S_n to S_{n+1}. Hence, at $t = (n+1)T$, the state becomes

$$S_{n+1} = (\theta_{n+1}, I_n, I_{n+1}, \ldots, I_{n-L+2})$$

where

$$\theta_{n+1} = \theta_n + \pi h I_{n-L+1}$$

Example 5-3-1

Consider a binary CPM scheme with a modulation index $h = 3/4$ and a partial response pulse with $L = 2$. Let us determine the states S_n of the CPM scheme and sketch the phase tree and state trellis.

First, we note that there are $2p = 8$ phase states, namely,

$$\Theta_s = \{0, \pm\tfrac{1}{4}\pi, \pm\tfrac{1}{2}\pi, \pm\tfrac{3}{4}\pi, \pi\}$$

For each of these phase states, there are two states that result from the memory of the CPM scheme. Hence, the total number of states is $N_s = 16$, namely,

$$(0, 1),\ (0, -1),\ (\pi, 1),\ (\pi, -1),\ (\tfrac{1}{4}\pi, 1),\ (\tfrac{1}{4}\pi, -1),\ (\tfrac{1}{2}\pi, 1),\ (\tfrac{1}{2}\pi, -1),$$
$$(\tfrac{3}{4}\pi, 1),\ (\tfrac{3}{4}\pi, -1),\ (-\tfrac{1}{4}\pi, 1),\ (-\tfrac{1}{4}\pi, -1),\ (-\tfrac{1}{2}\pi, 1),\ (-\tfrac{1}{2}\pi, -1),$$
$$(-\tfrac{3}{4}\pi, 1),\ (-\tfrac{3}{4}\pi, -1)$$

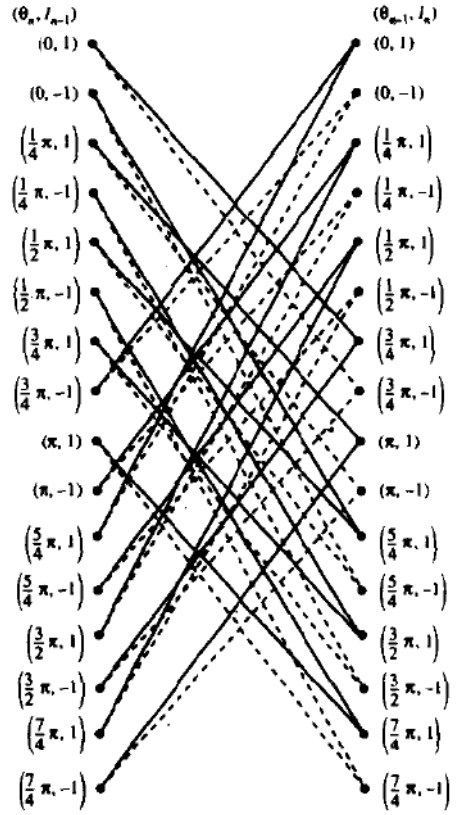


FIGURE 5-3-1 State trellis for partial response ($L = 2$) CPM with $h = \frac{3}{4}$.

If the system is in phase state $\theta_n = -\frac{1}{4}\pi$ and $I_{n-1} = -1$ then

$$\theta_{n+1} = \theta_n + \pi h I_{n-1}$$
$$= -\tfrac{1}{4}\pi - \tfrac{3}{4}\pi = -\pi$$

The state trellis is illustrated in Fig. 5-3-1. A path through the state trellis corresponding to the sequence $(1, -1, -1, -1, 1, 1)$ is illustrated in Fig. 5-3-2.

In order to sketch the phase tree, we must know the signal pulse shape $g(t)$. Figure 5-3-3 illustrates the phase tree when $g(t)$ is a rectangular pulse of duration $2T$, with initial state $(0, 1)$.

Having established the state trellis representation of CPM, let us now consider the metric computations performed in the Viterbi algorithm.

Metric Computations By referring back to the mathematical development

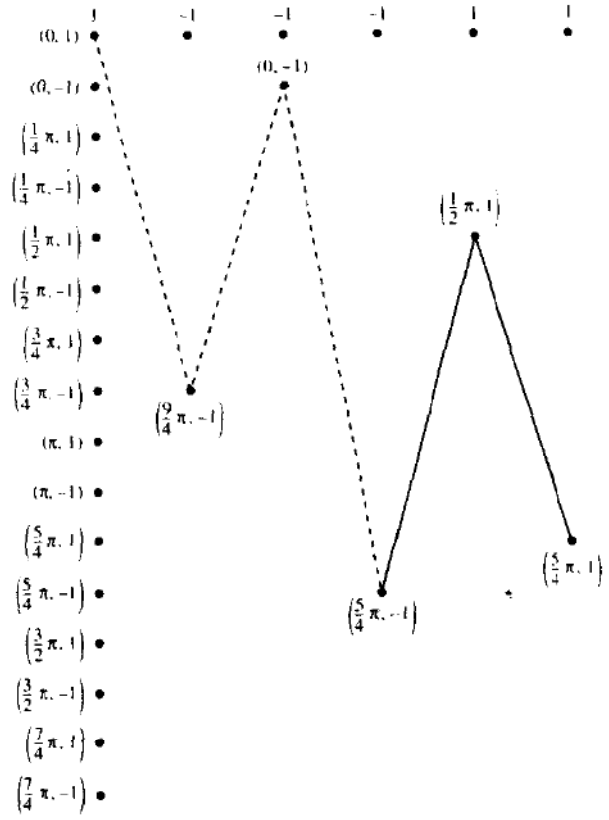


FIGURE 5-3-2 A single signal path through the trellis.

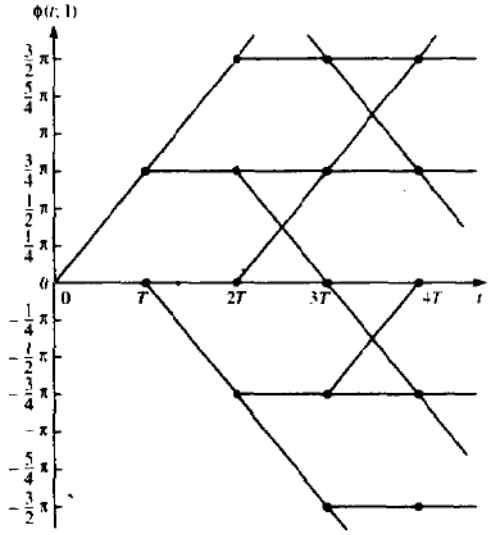


FIGURE 5-3-3 Phase tree for $L = 2$ partial response CPM with $h = \frac{3}{4}$.

for the derivation of the maximum likelihood demodulator given in Section 5-1-4, it is easy to show that the logarithm of the probability of the observed signal $r(t)$ conditioned on a particular sequence of transmitted symbols $\mathbf{I}$ is proportional to the cross-correlation metric

$$CM_n(\mathbf{I}) = \int_{-\infty}^{(n+1)T} r(t)\cos\left[\omega_c t + \phi(t;\mathbf{I})\right] dt$$

$$= CM_{n-1}(\mathbf{I}) + \int_{nT}^{(n+1)T} r(t)\cos\left[\omega_c t + \theta(t;\mathbf{I}) + \theta_n\right] dt \quad (5\text{-}3\text{-}11)$$

The term $CM_{n-1}(\mathbf{I})$ represents the metrics for the surviving sequences up to time nT, and the term

$$v_n(\mathbf{I};\theta_n) = \int_{nT}^{(n+1)T} r(t)\cos\left[\omega_c t + \theta(t;\mathbf{I}) + \theta_n\right] dt \quad (5\text{-}3\text{-}12)$$

represents the additonal increments to the metrics contributed by the signal in the time interval $nT \le t \le (n+1)T$. Note that there are M^L possible sequences $\mathbf{I} = (I_n, I_{n-1}, \ldots, I_{n-L+1})$ of symbols and p (or $2p$) possible phase states $\{\theta_n\}$. Therefore, there are pM^L (or $2pM^L$) different values of $v_n(\mathbf{I}, \theta_n)$, computed in each signal interval, and each value is used to increment the metrics corresponding to the pM^{L-1} surviving sequences from the previous signaling interval. A general block diagram that illustrates the computations of $v_n(\mathbf{I}; \theta_n)$ for the Viterbi decoder is shown in Fig. 5-3-4.

Note that the number of surviving sequences at each state of the Viterbi decoding process is pM^{L-1} (or $2pM^{L-1}$). For each surviving sequence, we have M new increments of $v_n(\mathbf{I}; \Theta_n)$ that are added to the existing metrics to yield pM^L (or $2pM^L$) sequences with pM^L (or $2pM^L$) metrics. However, this number is then reduced back to pM^{L-1} (or $2pM^{L-1}$) survivors with corresponding metrics by selecting the most probable sequence of the M sequences merging at each node of the trellis and discarding the other $M-1$ sequences.

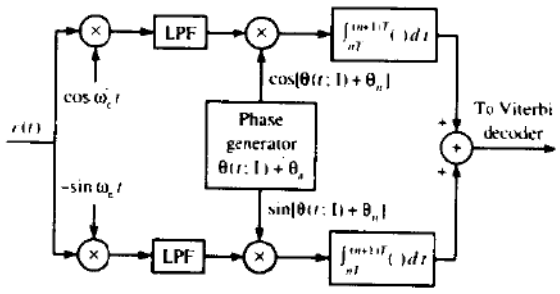


FIGURE 5-3-4 Computation of metric increments $v_n(\mathbf{I}; \theta_n)$.

5-3-2 Performance of CPM Signals

In evaluating the performance of CPM signals achieved with MLSE, we must determine the minimum euclidean distance of paths through the trellis that separate at the node at $t = 0$ and re-emerge at a later time at the same node. The distance between two paths through the trellis is related to the corresponding signals as we now demonstrate.

Suppose that we have two signals $s_i(t)$ and $s_j(t)$ corresponding to two phase trajectories $\phi(t; \mathbf{I}_i)$ and $\phi(t; \mathbf{I}_j)$. The sequences $\mathbf{I}_i$ and $\mathbf{I}_j$ must be different in their first symbol. Then, the euclidean distance between the two signals over an interval of length NT, where $1/T$ is the symbol rate, is defined as

$$\begin{aligned} d_{ij}^2 &= \int_0^{NT} [s_i(t) - s_j(t)]^2 \, dt \\ &= \int_0^{NT} s_i^2(t) \, dt + \int_0^{NT} s_j^2(t) \, dt - 2 \int_0^{NT} s_i(t) s_j(t) \, dt \\ &= 2N\mathscr{E} - 2 \frac{2\mathscr{E}}{T} \int_0^{NT} \cos [\omega_c t + \phi(t; \mathbf{I}_i)] \cos [\omega_c t + \phi(t; \mathbf{I}_j)] \, dt \\ &= 2N\mathscr{E} - \frac{2\mathscr{E}}{T} \int_0^{NT} \cos [\phi(t; \mathbf{I}_i) - \phi(t; \mathbf{I}_j)] \, dt \\ &= \frac{2\mathscr{E}}{T} \int_0^{NT} \{1 - \cos [\phi(t; \mathbf{I}_i) - \phi(t; \mathbf{I}_j)]\} \, dt \end{aligned} \tag{5-3-13}$$

Hence the euclidean distance is related to the phase difference between the paths in the state trellis according to (5-3-13).

It is desirable to express the distance d_{ij}^2 in terms of the bit energy. Since $\mathscr{E} = \mathscr{E}_b \log_2 M$, (5-3-13) may be expressed as

$$d_{ij}^2 = 2\mathscr{E}_b \delta_{ij}^2 \tag{5-3-14}$$

where δ_{ij}^2 is defined as

$$\delta_{ij}^2 = \frac{\log_2 M}{T} \int_0^{NT} \{1 - \cos [\phi(t; \mathbf{I}_i) - \phi(t; \mathbf{I}_j)]\} \, dt \tag{5-3-15}$$

Furthermore, we observe that $\phi(t; \mathbf{I}_i) - \phi(t; \mathbf{I}_j) = \phi(t; \mathbf{I}_i - \mathbf{I}_j)$, so that, with $\boldsymbol{\xi} = \mathbf{I}_i - \mathbf{I}_j$, (5-3-15) may be written as

$$\delta_{ij}^2 = \frac{\log_2 M}{T} \int_0^{NT} [1 - \cos \phi(t; \boldsymbol{\xi})] \, dt \tag{5-3-16}$$

where any element of $\boldsymbol{\xi}$ can take the values $0, \pm 2, \pm 4, \pm \ldots \pm 2(M-1)$, except that $\boldsymbol{\xi}_0 \neq 0$.

The error rate performance for CPM is dominated by the term corresponding to the minimum euclidean distance, and it may be expressed as

$$P_M = K_{\delta_{\min}} Q\left(\sqrt{\frac{\mathcal{E}_b}{N_0}\,\delta_{\min}^2}\right) \tag{5-3-17}$$

where

$$\begin{aligned}\delta_{\min}^2 &= \lim_{N\to\infty} \min_{i,j} \delta_{ij}^2 \\ &= \lim_{N\to\infty} \min_{i,j} \left\{\frac{\log_2 M}{T}\int_0^{NT} [1-\cos\phi(t;\mathbf{I}_i-\mathbf{I}_j)]\,dt\right\}\end{aligned} \tag{5-3-18}$$

We note that for conventional binary PSK with no memory, $N = 1$ and $\delta_{\min}^2 = \delta_{12}^2 = 2$. Hence, (5-3-17) agrees with our previous result.

Since $\delta_{\min}^2$ characterizes the performance of CPM with MLSE, we can investigate the effect on $\delta_{\min}^2$ resulting from varying the alphabet size M, the modulation index h, and the length of the transmitted pulse in partial response CPM.

First, we consider full response ($L = 1$) CPM. If we take $M = 2$ as a beginning, we note that the sequences

$$\begin{aligned}\mathbf{I}_i &= +1, -1, I_2, I_3 \\ \mathbf{I}_j &= -1, +1, I_2, I_3\end{aligned} \tag{5-3-19}$$

which differ for $k = 0, 1$ and agree for $k \geq 2$, result in two phase trajectories that merge after the second symbol. This corresponds to the difference sequence

$$\boldsymbol{\xi} = \{2, -2, 0, 0, \ldots\} \tag{5-3-20}$$

The euclidean distance for this sequence is easily calculated from (5-3-16), and provides an upper bound on $\delta_{\min}^2$. This upper bound for $M = 2$ is

$$d_H^2(h) = 2\left(1 - \frac{\sin 2\pi h}{2\pi h}\right), \qquad M = 2 \tag{5-3-21}$$

For example, where $h = \frac{1}{2}$, which corresponds to MSK, we have $d_H^2(\frac{1}{2}) = 2$, so that $\delta_{\min}^2(\frac{1}{2}) \leq 2$.

For $M > 2$ and full response CPM, it is also easily seen that phase trajectories merge at $t = 2T$. Hence, an upper bound on $\delta_{\min}^2$ can be obtained by considering the phase difference sequence $\boldsymbol{\xi} = \{\alpha, -\alpha, 0, 0, \ldots\}$ where $\alpha = \pm 2, \pm 4, \ldots, \pm 2(M-1)$. This sequence yields the upper bound

$$d_H^2(h) = \min_{1\leq k\leq M-1} \left\{(2\log_2 M)\left(1 - \frac{\sin 2k\pi h}{2k\pi h}\right)\right\} \tag{5-3-22}$$

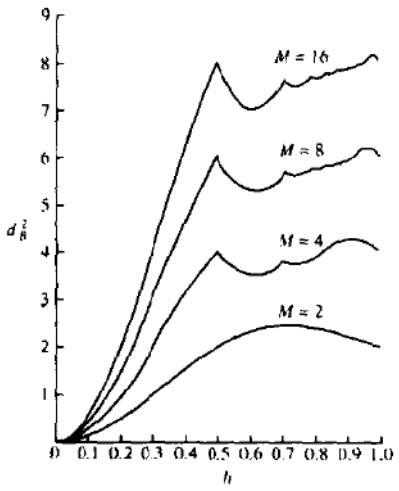


FIGURE 5-3-5 The upper bound d_B^2 as a function of the modulation index h for full response CPM with rectangular pulses. [*From Aulin and Sundberg (1984). © 1984, John Wiley Ltd. Reprinted with permission of the publisher.*]

The graphs of $d_B^2(h)$ versus h for $M = 2, 4, 8, 16$ are shown in Fig. 5-3-5. It is apparent from these graphs that large gains in performance can be achieved by increasing the alphabet size M. It must be remembered, however, that $\delta_{min}^2(h) \leq d_B^2(h)$. That is, the upper bound may not be achievable for all values of h.

The minimum euclidean distance $\delta_{\min}^2(h)$ has been determined, by evaluating (5-3-16), for a variety of CPM signals by Aulin and Sundberg (1981). For example, Fig. 5-3-6 illustrates the dependence of the euclidean distance for binary CPFSK as a function of the modulation index h, with the number N of

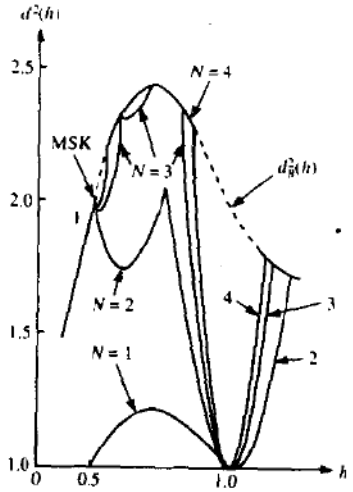


FIGURE 5-3-6 Squared minimum euclidean distance as a function of the modulation index for binary CPFSK. The upper bound is d_B^2. [*From Aulin and Sundberg (1981). © 1981 IEEE.*].

bit observation (decision) intervals ($N = 1, 2, 3, 4$) as a parameter. Also shown is the upper bound $d_B^2(h)$ given by (5-3-21). In particular, we note that when $h = \frac{1}{2}$, $\delta_{\min}^2(\frac{1}{2}) = 2$, which is the same squared distance as PSK (binary or quaternary) with $N = 1$. On the other hand, the required observation interval for MSK is $N = 2$ intervals, for which we have $\delta_{\min}^2(\frac{1}{2}) = 2$. Hence, the performance of MSK with MLSE is comparable to (binary or quaternary) PSK as we have previously observed.

We also note from Fig. 5-3-6 that the optimum modulation index for binary CPFSK is $h = 0.715$ when the observation interval is $N = 3$. This yields $\delta_{\min}^2(0.715) = 2.43$, or a gain of 0.85 dB relative to MSK.

Figure 5-3-7 illustrates the euclidean distance as a function of h for $M = 4$ CPFSK, with the length of the observation interval N as a parameter. Also shown (as a dashed line where it is not reached) is the upper bound d_B^2 evaluated from (5-3-22). Note that $\delta_{\min}^2$ achieves the upper bound for several values of h for some N. In particular, note that the maximum value of d_B^2, which occurs at $h \approx 0.9$, is approximately reached for $N = 8$ observed symbol intervals. The true maximum is achieved at $h = 0.914$ with $N = 9$. For this case, $\delta_{\min}^2(0.914) = 4.2$, which represents a 3.2 dB gain over MSK. Also note that the euclidean distance contains minima at $h = \frac{1}{3}, \frac{1}{2}, \frac{2}{3}, 1$, etc. These values of h are called *weak modulation indices* and should be avoided. Similar results are available for larger values of M, and may be found in the paper by Aulin and Sundberg (1981) and the text by Anderson *et al.* (1986).

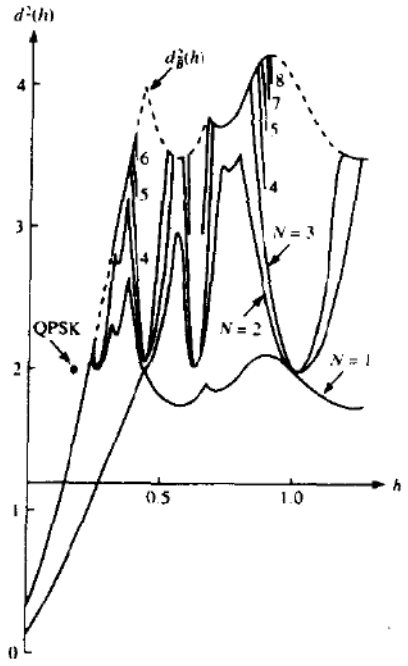


FIGURE 5-3-7 Squared minimum euclidean distance as a function of the modulation index for quaternary CPFSK. The upper bound is d_B^2. [*From Aulin and Sundberg (1981). © 1981 IEEE.*]

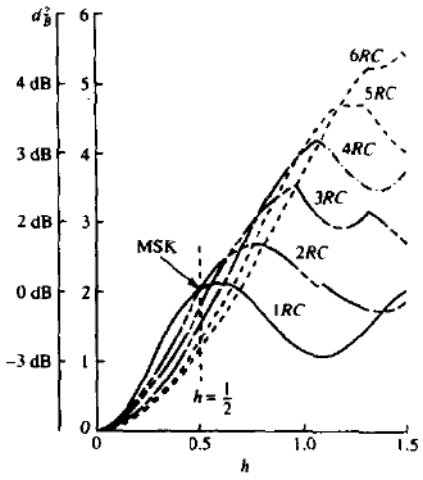


FIGURE 5-3-8 Upper bound d_B^2 on the minimum distance for partial response (raised cosine pulse) binary CPM. [*From Sundberg (1986). © 1986 IEEE.*]

Large performance gains can also be achieved with MLSE of CPM by using partial response signals. For example, the distance bound $d_B^2(h)$ for partial response, raised cosine pulses given by

$$g(t) = \begin{cases} \dfrac{1}{2LT}\left(1 - \cos\dfrac{2\pi t}{2LT}\right) & (0 \le t \le LT) \\ 0 & (\text{otherwise}) \end{cases} \tag{5-3-23}$$

is shown in Fig. 5-3-8 for $M = 2$. Here, note that, as L increases, d_B^2 also achieves higher values. Clearly, the performance of CPM improves as the correlative memory L increases, but h must also be increased in order to achieve the larger values of d_B^2. Since a larger modulation index implies a larger bandwidth (for fixed L), while a larger memory length L (for fixed h) implies a smaller bandwidth, it is better to compare the euclidean distance as a function of the normalized bandwidth $2WT_b$, where W is the 99% power bandwidth and T_b is the bit interval. Figure 5-3-9 illustrates this type of comparison with MSK used as a point of reference (0 dB). Note from this figure that there are several decibels to be gained by using partial response signals and higher signaling alphabets. The major price to be paid for this performance gain is the added exponentially increasing complexity in the implementation of the Viterbi decoder.

The performance results shown in Fig. 5-3-9 illustrate that 3–4 dB gain relative to MSK can be easily obtained with relatively no increase in bandwidth by the use of raised cosine partial response CPM and $M = 4$. Although these results are for raised cosine signal pulses, similar gains can be achieved with other partial response pulse shapes. We emphasize that this gain in SNR is achieved by introducing memory into the signal modulation and exploiting the memory in the demodulation of the signal. No redundancy through coding has

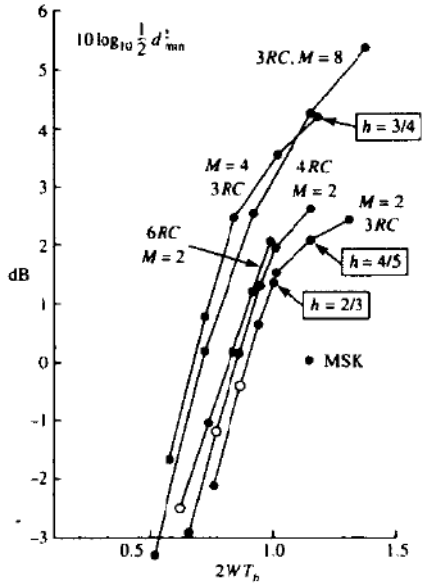


FIGURE 5-3-9 Power bandwidth tradeoff for partial response CPM signals with raised cosine pulses. W is the 99 percent in-band power bandwidth. [*From Sundberg (1986). © 1986 IEEE.*]

been introduced. In effect, the code has been built into the modulation and the trellis-type (Viterbi) decoding exploits the phase constraints in the CPM signal.

Additional gains in performance can be achieved by introducing additional redundancy through coding and increasing the alphabet size as a means of maintaining a fixed bandwidth. In particular, trellis-coded CPM using relatively simple convolution codes has been thoroughly investigated and many results are available in the technical literature. The Viterbi decoder for the convolutionally encoded CPM signal now exploits the memory inherent in the code and in the CPM signal. Performance gains of the order of 4–6 dB, relative to uncoded MSK with the same bandwidth, have been demonstrated by combining convolutional coding with CPM. Extensive numerical results for coded CPM are given by Lindell (1985).

Multi-h CPM By varying the modulation index from one signaling interval to another, it is possible to increase the minimum euclidean distance $\delta^2_{\min}$ between pairs of phase trajectories and, thus, improve the performance gain over constant-h CPM. Usually, multi-h CPM employs a fixed number H of modulation indices that are varied cyclically in successive signaling intervals. Thus, the phase of the signal varies piecewise linearly.

Significant gains in SNR are achievable by using only a small number of different values of h. For example, with full response ($L = 1$) CPM and $H = 2$, it is possible to obtain a gain of 3 dB relative to binary or quaternary PSK. By increasing H to $H = 4$, a gain of 4.5 dB relative to PSK can be obtained. The performance gain can also be increased with an increase in the signal alphabet.

Table 5-3-1 lists the performance gains achieved with $M = 2, 4,$ and 8 for several values of H. The upper bounds on the minimum euclidean distance are also shown in Fig. 5-3-10 for several values of M and H. Note that the major gain in performance is obtained when H is increased from $H = 1$ to $H = 2$. For $H > 2$, the additional gain is relatively small for small values of $\{h_i\}$. On the other hand, significant performance gains are achieved by increasing the alphabet size M.

The results shown above hold for full response CPM. One can also extend the use of multi-h CPM to partial response in an attempt to further improve performance. It is anticipated that such schemes will yield some additional performance gains, but numerical results on partial response, multi-h CPM are limited. The interested reader is referred to the paper by Aulin and Sundberg (1982b).

Multiamplitude CPM Multiamplitude CPM (MACPM) is basically a combined amplitude and phase digital modulation scheme that allows us to increase the signaling alphabet relative to CPM in another dimension and, thus, to achieve higher data rates on a band-limited channel. Simultaneously, the combination of multiple amplitude in conjunction with CPM results in a bandwidth-efficient modulation technique.

We have already observed the spectral characteristics of MACPM in Section 4-3. The performance characteristics of MACPM have been investigated by Mulligan (1988) for both uncoded and trellis-coded CPM. Of particular interest is the result that trellis-coded CPM with two amplitude levels achieves a gain of 3–4 dB relative to MSK without a significant increase in the signal bandwidth.

5-3-3 Symbol-by-Symbol Detection of CPM Signals

Besides the ML sequence detector, there are other types of detectors that can be used to recover the information sequence in a CPM signal. In this section, we consider symbol-by-symbol detectors. One type of symbol-by-symbol detector is the one described in Section 5-1-5, which exploits the memory of CPM by performing matched filtering or cross-correlation over several signaling intervals. Because of its computational complexity, however, this recursive algorithm has not been directly applied to the detection of CPM. Instead, two similar, albeit suboptimal, symbol-by-symbol detection methods have been described in the papers by deBuda (1972), Osborne and Luntz (1974), and Schonhoff (1976). One of these is functionally equivalent to the algorithm given in Section 5-1-5, and the second is a suboptimum approximation of the first. We shall describe these two methods in the context of demodulation of CPFSK signals, for which these detection algorithms have been applied directly.

To describe these methods, we assume that the signal is observed over the present signaling interval and D signaling intervals into the future in deciding on the information symbol transmitted in the present signaling interval. A

TABLE 5-3-1 MAXIMUM VALUES OF THE UPPER BOUND d_B^2 FOR MULTI-h LINEAR PHASE CPM[a]

M	H	Max d_B^2	dB gain compared with MSK	h_1	h_2	h_3	h_4	$\bar{h}$
2	1	2.43	0.85	0.715				0.715
2	2	4.0	3.0	0.5	0.5			0.5
2	3	4.88	3.87	0.620	0.686	0.714		0.673
2	4	5.69	4.54	0.73	0.55	0.73	0.55	0.64
4	1	4.23	3.25	0.914				0.914
4	2	6.54	5.15	0.772	0.772			0.772
4	3	7.65	5.83	0.795	0.795	0.795		0.795
8	1	6.14	4.87	0.964				0.964
8	2	7.50	5.74	0.883	0.883			0.883
8	3	8.40	6.23	0.879	0.879	0.879		0.879

[a] From Aulin and Sundberg (1982b).

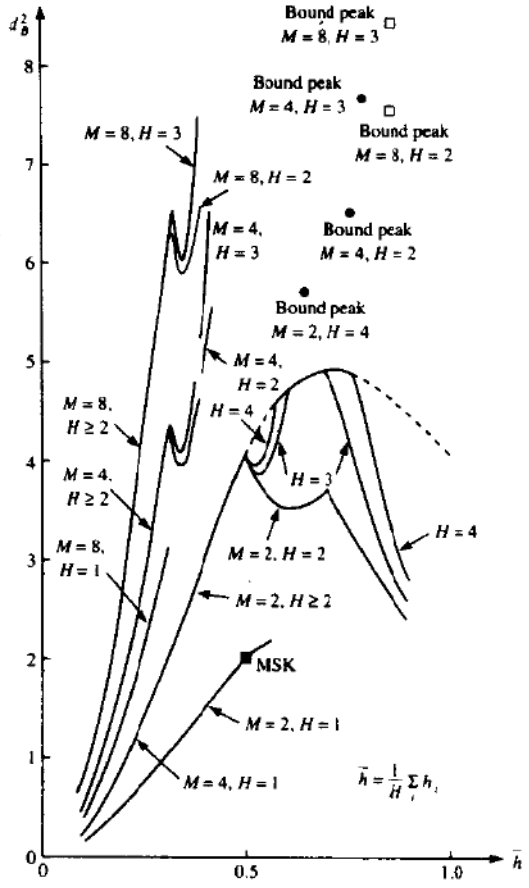


FIGURE 5-3-10 Upper bounds on minimum squared euclidean distance for various M and H values. [*From Aulin and Sundberg (1982b). © 1982 IEEE.*]

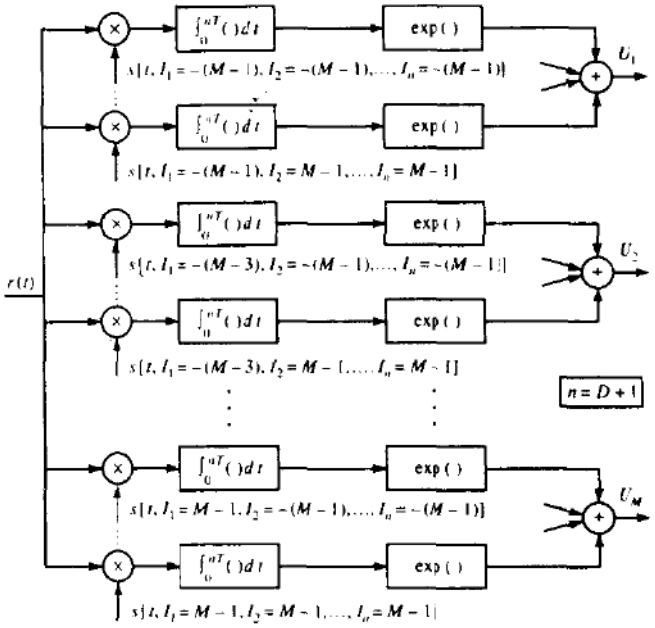


FIGURE 5-3-11 Block diagram of demodulator for detection of CPFSK.

block diagram of the demodulator, implemented as a bank of cross-correlators, is shown in Fig. 5-3-11. Recall that the transmitted CPFSK signal during the nth signaling interval is

$$s(t) = \text{Re}\,[v(t)e^{j2\pi f_c t}]$$

where

$$v(t) = \exp\left\{j\left[\frac{\pi h[t-(n-1)T]I_n}{T} + \pi h\sum_{k=0}^{n-1} I_k + \phi_0\right]\right\}$$

$h = 2f_dT$ is the modulation index, f_d is the peak frequency deviation, and ϕ_0 is the initial phase angle of the carrier.

In detecting the symbol I_1, the cross-correlations shown in Fig. 5-3-11 are performed with the reference signals $s(t, I_1, I_2, \ldots, I_{1+D})$ for all M^{D+1} possible values of the symbols $I_1, I_2, \ldots, I_{1+D}$ transmitted over the $D+1$ signaling intervals. But these correlations in effect generate the variables $r_1, r_2, \ldots, r_{1+D}$, which in turn are the arguments of the exponentials that occur in the pdf

$$p(r_1, r_2, \ldots, r_{1+D} \mid I_1, I_2, \ldots, I_{1+D})$$

Finally, the summations over the M^D possible values of the symbols $I_2, I_3, \ldots, I_{1+D}$ represent the averaging of

$$p(r_1, r_2, \ldots, r_{1+D} \mid I_1, I_2, \ldots, I_{1+D})P(I_1, I_2, \ldots, I_{1+D})$$

over the M^D possible values of these symbols. The M outputs of the demodulator constitute the decision variables from which the largest is selected to form the demodulated symbol. Consequently the metrics generated by the demodulator shown in Fig. 5-3-11 are equivalent to the decision variables given by (5-1-68) on which the decision on I_1 is based.

Signals received in subsequent signaling intervals are demodulated in the same manner. That is, the demodulator cross-correlates the signal received over $D+1$ signaling intervals with the M^{D+1} possible transmitted signals and forms the decision variables as illustrated in Fig. 5-3-11. Thus the decision made on the mth signaling interval is based on the cross-correlations performed over the signaling intervals $m, m+1, \ldots, m+D$. The initial phase in the correlation interval of duration $(D+1)T$ is assumed to be known. On the other hand, the algorithm described by (5-1-76) and (5-1-77) involves an additional averaging operation over the previously detected symbols. In this respect, the demodulator shown in Fig. 5-3-11 differs from the recursive algorithm described above. However, the difference is insignificant.

One suboptimum demodulation method that performs almost as well as the optimum method embodied in Fig. 5-3-11 bases its decision on the largest output from the bank of M^{D+1} cross-correlators. Thus the exponential functions and the summations are eliminated. But this method is equivalent to selecting the symbol I_m for which the probability density function $p(r_m, r_{m+1}, \ldots, r_{m+D} \mid I_m, I_{m+1}, \ldots, I_{m+D})$ is a maximum.

The performance of the detector shown in Fig. 5-3-11 has been upper-bounded and evaluated numerically. Figure 5-3-12 illustrates the performance of binary CPFSK with $n = D+1$ as a parameter. The modulation index $h = 0.715$ used in generating these results minimizes the probability of error as

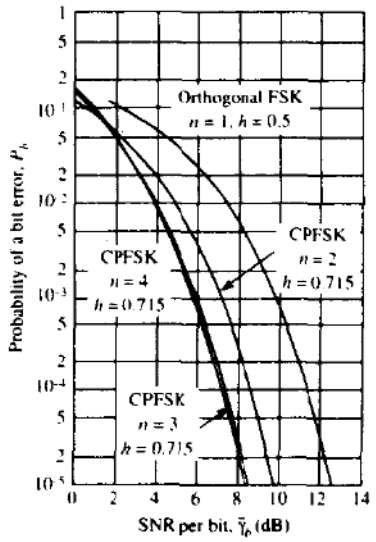


FIGURE 5-3-12 Performance of binary CPFSK with coherent detection.

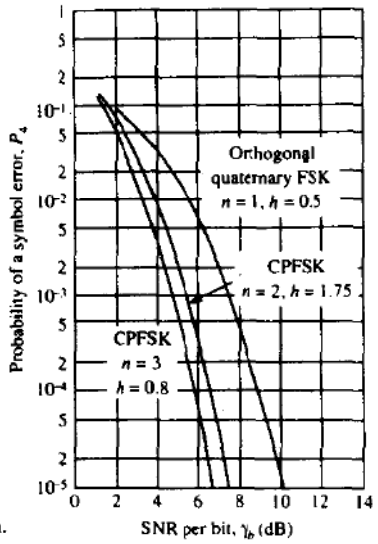


FIGURE 5-3-13 Performance of quaternary CPFSK with coherent detection.

shown by Schonhoff (1976). We note that an improvement of about 2.5 dB is obtained relative to orthogonal FSK ($n = 1$) by a demodulator that cross-correlates over two symbols. An additional gain of approximately 1.5 dB is obtained by extending the correlation time to three symbols. Further extension of the correlation time results in a relatively small additional gain.

Similar results are obtained with larger alphabet sizes. For example, Figs 5-3-13 and 5-3-14 illustrate the performance improvements for quaternary and

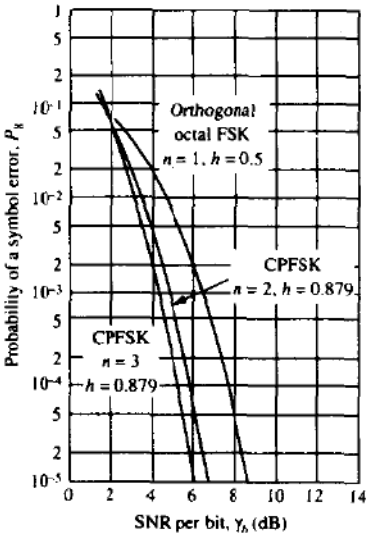


FIGURE 5-3-14 Performance of octal CPFSK with coherent detection.

octal CPFSK, respectively. The modulation indices given in these graphs are the ones that minimize the probability of a symbol error.

Instead of performing coherent detection, which requires knowledge of the carrier phase ϕ_0, we may assume that ϕ_0 is uniformly distributed over the interval 0 to 2π, and average over it in arriving at the decision variables. Thus coherent integration (cross-correlation) is performed over the $n = D + 1$ signaling intervals, but the outputs of the correlators are envelope-detected. This is called *noncoherent detection* of CPFSK. In this detection scheme, performance is optimized by selecting n to be odd and making the decision on the middle symbol in the sequence of n symbols. The numerical results on the probability of error for noncoherent detection of CPFSK are similar to the results illustrated above for coherent detection. That is, a gain of 2–3 dB in performance is achieved by increasing the correlation interval from $n = 1$ to $n = 3$ and to $n = 5$.

5-4 OPTIMUM RECEIVER FOR SIGNALS WITH RANDOM PHASE IN AWGN CHANNEL

In this section, we consider the design of the optimum receiver for carrier modulated signals when the carrier phase is unknown at the receiver and no attempt is made to estimate its value. Uncertainty in the carrier phase of the received signal may be due to one or more of the following reasons: First, the oscillators that are used at the transmitter and the receiver to generate the carrier signals are generally not phase synchronous. Second, the time delay in the propagation of the signal from the transmitter to the receiver is not generally known precisely. To elaborate on this point, a transmitted signal of the form

$$s(t) = \text{Re}\,[g(t)e^{j2\pi f_c t}]$$

that propagates through a channel with delay t_0 will be received as

$$\begin{aligned} s(t - t_0) &= \text{Re}\,[g(t - t_0)e^{j2\pi f_c(t - t_0)}] \\ &= \text{Re}\,[g(t - t_0)e^{-j2\pi f_c t_0}e^{j2\pi f_c t}] \end{aligned}$$

The carrier phase shift due to the propagation delay t_0 is

$$\phi = -2\pi f_c t_0$$

Note that large changes in the carrier phase ϕ can occur due to relatively small changes in the propagation delay. For example, if the carrier frequency $f_c = 1$ MHz, an uncertainty or a change in the propagation delay of 0.5 μs will cause a phase uncertainity of π rad. In some channels (e.g., radio channels) the time delay in the propagation of the signal from the transmitter to the receiver

may change rapidly and in an apparently random manner, so that the carrier phase of the received signal varies in an apparently random fashion.

In the absence of knowledge of the carrier phase, we may treat this signal parameter as a random variable and determine the form of the optimum receiver for recovering the transmitted information from the received signal. First, we treat the case of binary signals and, then, we consider M-ary signals.

5-4-1 Optimum Receiver for Binary Signals

We consider a binary communication system that uses the two carrier modulated signals $s_1(t)$ and $s_2(t)$ to transmit the information, where

$$s_m(t) = \text{Re}\,[s_{lm}(t)e^{j2\pi f_c t}], \quad m = 1, 2, \quad 0 \leq t \leq T \tag{5-4-1}$$

and $s_{lm}(t)$, $m = 1, 2$ are the equivalent lowpass signals. The two signals are assumed to have equal energy

$$\mathcal{E} = \int_0^T s_m^2(t)\,dt = \frac{1}{2}\int_0^T |s_{lm}(t)|^2\,dt \tag{5-4-2}$$

and are characterized by the complex-valued correlation coefficient

$$\rho_{12} = \rho = \frac{1}{\mathcal{E}}\int_0^T s_{l1}^*(t)s_{l2}(t)\,dt \tag{5-4-3}$$

The received signal is assumed to be a phase-shifted version of the transmitted signal and corrupted by the additive noise

$$\begin{aligned} n(t) &= \text{Re}\,\{[n_c(t) + jn_s(t)]e^{j2\pi f_c t}\} \\ &= \text{Re}\,[z(t)e^{j2\pi f_c t}] \end{aligned} \tag{5-4-4}$$

Hence, the received signal may be expressed as

$$r(t) = \text{Re}\,\{[s_{lm}(t)e^{j\phi} + z(t)]e^{j2\pi f_c t}\} \tag{5-4-5}$$

where

$$r_l(t) = s_{lm}(t)e^{j\phi} + z(t), \quad 0 \leq t \leq T \tag{5-4-6}$$

is the equivalent lowpass received signal. This received signal is now passed through a demodulator whose sampled output at $t = T$ is passed to the detector.

The Optimum Demodulator In Section 5-1-1, we demonstrated that if the received signal was correlated with a set of orthonormal functions $\{f_n(t)\}$ that

spanned the signal space, the outputs from the bank of correlators provide a set of sufficient statistics for the detector to make a decision that minimizes the probability of error. We also demonstrated that a bank of matched filters could be substituted for the bank of correlators.

A similar orthonormal decomposition can also be employed for a received signal with an unknown carrier phase. However, it is mathematically convenient to deal with the equivalent lowpass signal and to specify the signal correlators or matched filters in terms of the equivalent lowpass signal waveforms.

To be specific, the impulse response $h_l(t)$ of a filter that is matched to the complex-valued equivalent lowpass signal $s_l(t)$, $0 \leq t \leq T$, is given as (see Problem 5-6)

$$h_l(t) = s_l^*(T - t) \tag{5-4-7}$$

and the output of such a filter at $t = T$ is simply

$$\int_0^T |s_l(t)|^2 \, dt = 2\mathscr{E} \tag{5-4-8}$$

where $\mathscr{E}$ is the signal energy. A similar result is obtained if the signal $s_l(t)$ is correlated with $s_l^*(t)$ and the correlator is sampled at $t = T$. Therefore, the optimum demodulator for the equivalent lowpass received signal $s_l(t)$ given in (5-4-6) may be realized by two matched filters in parallel, one matched to $s_{l1}(t)$ and the other to $s_{l2}(t)$, and shown in Fig. 5-4-1. The output of the matched filters or correlators at the sampling instant are the two complex numbers

$$r_m = r_{mc} + jr_{ms}, \qquad m = 1, 2 \tag{5-4-9}$$

Suppose that the transmitted signal is $s_1(t)$. Then, it is easily shown (see Problem 5-35) that

$$\begin{aligned} r_1 &= 2\mathscr{E} \cos \phi + n_{1c} + j(2\mathscr{E} \sin \phi + n_{1s}) \\ r_2 &= 2\mathscr{E}|\rho| \cos(\phi + \alpha_0) + n_{2c} + j[2\mathscr{E}|\rho| \sin(\phi + \alpha_0) + n_{2s}] \end{aligned} \tag{5-4-10}$$

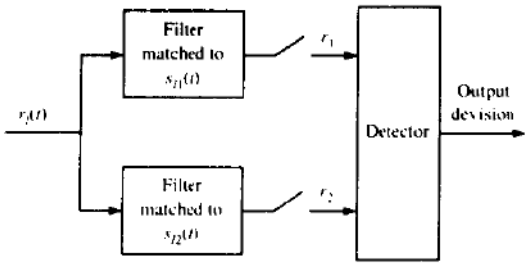


FIGURE 5-4-1 Optimum receiver for binary signals.

where ρ is the complex-valued correlation coefficient of the two signals $s_{l1}(t)$ and $s_{l2}(t)$, which may be expressed as $\rho = |\rho| \exp(\alpha_0)$. The random noise variables n_{1c}, n_{1s}, n_{2c}, and n_{2s} are jointly gaussian, with zero mean and equal variance.

The Optimum Detector The optimum detector observes the random variables $[r_{1c}\ r_{1s}\ r_{2c}\ r_{2s}] = \mathbf{r}$, where $r_1 = r_{1c} + jr_{1s}$ and $r_2 = r_{2c} + jr_{2s}$, and bases its decision on the posterior probabilities $P(\mathbf{s}_m \mid \mathbf{r})$, $m = 1, 2$. These probabilities may be expressed as

$$P(\mathbf{s}_m \mid \mathbf{r}) = \frac{p(\mathbf{r} \mid \mathbf{s}_m)P(\mathbf{s}_m)}{p(\mathbf{r})}, \qquad m = 1, 2 \tag{5-4-11}$$

and, hence, the optimum decision rule may be expressed as

$$P(\mathbf{s}_1 \mid \mathbf{r}) \underset{s_2}{\overset{s_1}{\gtrless}} P(\mathbf{s}_2 \mid \mathbf{r})$$

or, equivalently,

$$\frac{p(\mathbf{r} \mid \mathbf{s}_1)}{p(\mathbf{r} \mid \mathbf{s}_2)} \underset{s_2}{\overset{s_1}{\gtrless}} \frac{P(\mathbf{s}_2)}{P(\mathbf{s}_1)} \tag{5-4-12}$$

The ratio of pdfs on the left-hand side of (5-4-12) is the *likelihood ratio*, which we denote as

$$\Lambda(\mathbf{r}) = \frac{p(\mathbf{r} \mid \mathbf{s}_1)}{p(\mathbf{r} \mid \mathbf{s}_2)} \tag{5-4-13}$$

The right-hand side of (5-4-12) is the ratio of the two prior probabilities, which takes the value of unity when the two signals are equally probable.

The probability density functions $p(\mathbf{r} \mid \mathbf{s}_1)$ and $p(\mathbf{r} \mid \mathbf{s}_2)$ can be obtained by averaging the pdfs $p(\mathbf{r} \mid \mathbf{s}_m, \phi)$ over the pdf of the random carrier phase, i.e.,

$$p(\mathbf{r} \mid \mathbf{s}_m) = \int_0^{2\pi} p(\mathbf{r} \mid \mathbf{s}_m, \phi)p(\phi)\, d\phi \tag{5-4-14}$$

We shall perform the integration indicated in (5-4-14) for the special case in which the two signals are orthogonal, i.e., $\rho = 0$. In this case, the outputs of the demodulator are

$$\begin{aligned} r_1 &= r_{1c} + jr_{1s} \\ &= 2\mathscr{E} \cos\phi + n_{1c} + j(2\mathscr{E} \sin\phi + n_{1s}) \\ r_2 &= r_{2c} + jr_{2s} \\ &= n_{2c} + jn_{2s} \end{aligned} \tag{5-4-15}$$

where $(n_{1c}, n_{1s}, n_{2c}, n_{2s})$ are mutually uncorrelated and, hence, statistically independent, zero-mean gaussian random variables (see Problem 5-25). Hence, the joint pdf of $\mathbf{r} = [r_{1c}\ r_{1s}\ r_{2c}\ r_{2s}]$ may be expressed as a product of the marginal pdfs. Consequently,

$$p(r_{1c}, r_{1s} \mid \mathbf{s}_1, \phi) = \frac{1}{2\pi\sigma^2} \exp\left[-\frac{(r_{1c} - 2\mathscr{E}\cos\phi)^2 + (r_{1s} - 2\mathscr{E}\sin\phi)^2}{2\sigma^2}\right]$$
$$p(r_{2c}, r_{2s}) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{r_{2c}^2 + r_{2s}^2}{2\sigma^2}\right) \tag{5-4-16}$$

where $\sigma^2 = 2\mathscr{E}N_0$.

The uniform pdf for the carrier phase ϕ represents the most ignorance that can be exhibited by the detector. This is called the *least favorable pdf* for ϕ. With $p(\phi) = 1/2\pi$, $0 \leq \phi \leq 2\pi$, substituted into the integral in (5-4-14), we obtain

$$\frac{1}{2\pi}\int_0^{2\pi} p(r_{1c}, r_{1s} \mid \mathbf{s}_1, \phi)\, d\phi$$
$$= \frac{1}{2\pi}\exp\left(-\frac{r_{1c}^2 + r_{1c}^2 + 4\mathscr{E}^2}{2\sigma^2}\right)\frac{1}{2\pi}\int \exp\left[\frac{2\mathscr{E}(r_{1c}\cos\phi + r_{1s}\sin\phi)}{\sigma^2}\right] d\phi \tag{5-4-17}$$

But

$$\frac{1}{2\pi}\int_0^{2\pi} \exp\left[\frac{2\mathscr{E}(r_{1c}\cos\phi + r_{1s}\sin\phi)}{\sigma^2}\right] d\phi = I_0\left(\frac{2\mathscr{E}\sqrt{r_{1c}^2 + r_{1s}^2}}{\sigma^2}\right) \tag{5-4-18}$$

where $I_0(x)$ is the modified Bessel function of zeroth order, defined in (2-1-120).

By performing a similar integration as in (5-4-17) under the assumption that the signal $s_2(t)$ was transmitted, we obtain the result

$$p(r_{2c}, r_{2s} \mid \mathbf{s}_2) = \frac{1}{2\pi}\exp\left(-\frac{r_{2c}^2 + r_{2s}^2 + 4\mathscr{E}^2}{2\sigma^2}\right) I_0\left(\frac{2\mathscr{E}\sqrt{r_{2c}^2 + r_{2s}^2}}{\sigma^2}\right) \tag{5-4-19}$$

When, we substitute these results into the likelihood ratio given by (5-4-13), we obtain the result

$$\Lambda(\mathbf{r}) = \frac{I_0(2\mathscr{E}\sqrt{r_{1c}^2 + r_{1s}^2}/\sigma^2)}{I_0(2\mathscr{E}\sqrt{r_{2c}^2 + r_{2s}^2}/\sigma^2)} \underset{s_2}{\overset{s_1}{\gtrless}} \frac{P(\mathbf{s}_2)}{P(\mathbf{s}_1)} \tag{5-4-20}$$

Thus, the optimum detector computes the two envelopes $\sqrt{r_{1c}^2 + r_{1s}^2}$ and $\sqrt{r_{2c}^2 + r_{2s}^2}$ and the corresponding values of the Bessel function $I_0(2\mathscr{E}\sqrt{r_{1c}^2 + r_{1s}^2}/\sigma^2)$ and $I_0(2\mathscr{E}\sqrt{r_{2c}^2 + r_{2s}^2}/\sigma^2)$ to form the likelihood ratio. We observe that this computation requires knowledge of the noise variance σ^2.

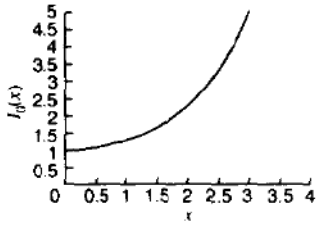


FIGURE 5-4-2 Graph of $I_0(x)$.

The likelihood ratio is then compared with the threshold $P(\mathbf{s}_2)/P(\mathbf{s}_1)$ to determine which signal was transmitted.

A significant simplification in the implementation of the optimum detector occurs when the two signals are equally probable. In such a case the threshold becomes unity, and, due to the monotonicity of the Bessel function shown in Fig. 5-4-2, the optimum detection rule simplifies to

$$\sqrt{r_{1c}^2 + r_{1s}^2} \underset{s_2}{\overset{s_1}{\gtrless}} \sqrt{r_{2c}^2 + r_{2s}^2} \tag{5-4-21}$$

Thus, the optimum detector bases its decision on the two envelopes $\sqrt{r_{1c}^2 + r_{1s}^2}$ and $\sqrt{r_{2c}^2 + r_{2s}^2}$, and, hence, it is called an *envelope detector.*

We observe that the computation of the envelopes of the received signal samples at the output of the demodulator renders the carrier phase irrelevant in the decision as to which signal was transmitted. Equivalently, the decision may be based on the computation of the squared envelopes $r_{1c}^2 + r_{1s}^2$ and $r_{2c}^2 + r_{2s}^2$, in which case the detector is called a *square-law detector.*

Binary FSK signals are an example of binary orthogonal signals. Recall that in binary FSK we employ two different frequencies, say f_1 and $f_2 = f_1 + \Delta f$, to transmit a binary information sequence. The choice of minimum frequency separation $\Delta f = f_2 - f_1$ is considered below. Thus, the signal waveforms may be expressed as

$$\begin{aligned} s_1(t) &= \sqrt{2\mathscr{E}_b/T_b}\cos 2\pi f_1 t, \quad 0 \le t \le T_b \\ s_2(t) &= \sqrt{2\mathscr{E}_b/T_b}\cos 2\pi f_2 t, \quad 0 \le t \le T_b \end{aligned} \tag{5-4-22}$$

and their equivalent lowpass counterparts are

$$\begin{aligned} s_{l1}(t) &= \sqrt{2\mathscr{E}_b/T_b}, \quad 0 \le t \le T_b \\ s_{l2}(t) &= \sqrt{2\mathscr{E}_b/T_b}e^{j2\pi\Delta f t}, \quad 0 \le t \le T_b \end{aligned} \tag{5-4-23}$$

The received signal may be expressed as

$$r(t) = \sqrt{\frac{2\mathscr{E}_b}{T_b}}\cos(2\pi f_m t + \phi_m) + n(t) \tag{5-4-24}$$

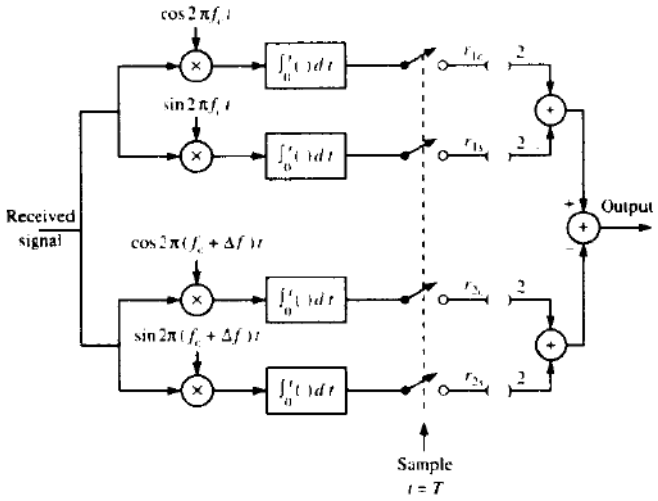


FIGURE 5-4-3 Demodulation and square-law detection of binary FSK signals.

where ϕ_m is the phase of the carrier frequency f_m. The demodulation of the real signal $r(t)$ may be accomplished, as shown in Fig. 5-4-3, by using four correlators with the basis functions

$$f_{1m}(t) = \sqrt{\frac{2}{T_b}} \cos [(2\pi f_1 + 2\pi m\, \Delta f)t], \qquad m = 0, 1$$
$$f_{2m}(t) = \sqrt{\frac{2}{T_b}} \sin [(2\pi f_1 + 2\pi m\, \Delta f)t], \qquad m = 0, 1 \tag{5-4-25}$$

The four outputs of the correlators are sampled at the end of each signal interval and passed to the detector. If the mth signal is transmitted, the four samples at the detector may be expressed as

$$r_{kc} = \sqrt{\mathcal{E}_b}\left[\frac{\sin [2\pi(k-m)\,\Delta f\, T]}{2\pi(k-m)\,\Delta f\, T} \cos \phi_m - \frac{\cos [2\pi(k-m)\,\Delta f\, T] - 1}{2\pi(k-m)\,\Delta f\, T} \sin \phi_m\right] + n_{kc}, \qquad k, m = 1, 2$$

(5-4-26)

$$r_{ks} = \sqrt{\mathcal{E}_b}\left[\frac{\cos 2\pi(k-m)\,\Delta f\, T - 1}{2\pi(k-m)\,\Delta f\, T} \cos \phi_m + \frac{\sin [2\pi(k-m)\,\Delta f\, T]}{2\pi(k-m)\,\Delta f\, T} \sin \phi_m\right] + n_{ks}, \qquad k, m = 1, 2$$

where n_{kc} and n_{ks} denote the gaussian noise components in the sampled outputs.

We observe that when $k = m$, the sampled values to the detector are

$$r_{mc} = \sqrt{\mathcal{E}_b}\cos\phi_m + n_{mc}$$
$$r_{ms} = \sqrt{\mathcal{E}_b}\sin\phi_m + n_{ms} \tag{5-4-27}$$

Furthermore, we observe that when $k \neq m$, the signal components in the samples r_{kc} and r_{ks} will vanish, independently of the values of the phase shifts ϕ_k, provided that the frequency separation between successive frequencies is $\Delta f = 1/T$. In such a case, the other two correlator outputs consist of noise only, i.e.,

$$r_{kc} = n_{kc}, \quad r_{ks} = n_{ks}, \quad k \neq m \tag{5-4-28}$$

With a frequency separation of $\Delta f = 1/T$, the relations (5-4-27) and (5-4-28) are consistent with the previous result (5-4-15) for the demodulator outputs. Therefore, we conclude that for envelope or square-law detection of FSK signals, the minimum frequency separation required for orthogonality of the signals is $\Delta f = 1/T$. This separation is twice as large as that required when the detection is phase-coherent.

5-4-2 Optimum Receiver for M-ary Orthogonal Signals

The generalization of the optimum demodulator and detector to the case of M-ary orthogonal signals is straightforward. If the equal energy and equally probable signal waveforms are represented as

$$s_m(t) = \text{Re}\,[s_{lm}(t)e^{j2\pi f_c t}], \quad m = 1, 2, \ldots, M, \quad 0 \leq t \leq T \tag{5-4-29}$$

where $s_{lm}(t)$ are the equivalent lowpass signals, the optimum correlation-type or matched-filter-type demodulator produces the M complex-valued random variables

$$r_m = r_{mc} + jr_{ms} = \int_0^T r_l(t)s_{lm}^*(t)\,dt, \quad m = 1, 2, \ldots, M \tag{5-4-30}$$

where $r_l(t)$ is the equivalent lowpass received signal. Then, the optimum detector, based on a random, uniformly distributed carrier phase, computes the M envelopes

$$|r_m| = \sqrt{r_{mc}^2 + r_{ms}^2}, \quad m = 1, 2, \ldots, M \tag{5-4-31}$$

or, equivalently, the squared envelopes $|r_m|^2$, and selects the signal with the largest envelope (or squared envelope).

In the special case of M-ary orthogonal FSK signals, the optimum receiver has the structure illustrated in Fig. 5-4-4. There are $2M$ correlators: two for each possible transmitted frequency. The minimum frequency separation between adjacent frequencies to maintain orthogonality is $\Delta f = 1/T$.

5-4-3 Probability of Error for Envelope Detection of M-ary Orthogonal Signals

Let us consider the transmission of M-ary orthogonal equal energy signals over an AWGN channel, which are envelope-detected at the receiver. We also

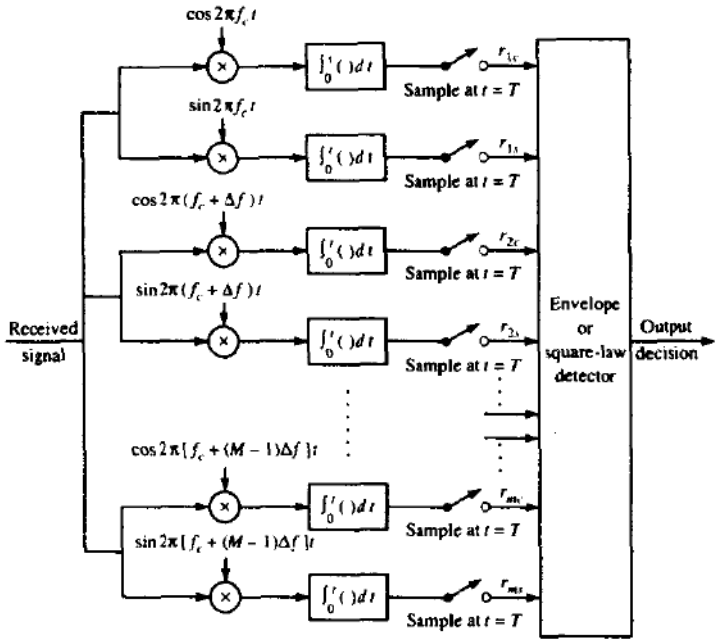


FIGURE 5-4-4 Demodulation of M-ary FSK signals for noncoherent detection.

assume that the M signals are equally probable a priori and that the signal $s_1(t)$ is transmitted in the signal interval $0 \leq t \leq T$.

The M decision metrics at the detector are the M envelopes

$$|\mathbf{r}_m| = \sqrt{r_{mc}^2 + r_{ms}^2}, \qquad m = 1, 2, \ldots, M \tag{5-4-32}$$

where

$$\begin{aligned} r_{1c} &= \sqrt{\mathscr{E}_s}\cos\phi_1 + n_{1c} \\ r_{1s} &= \sqrt{\mathscr{E}_s}\sin\phi_1 + n_{1s} \end{aligned} \tag{5-4-33}$$

and

$$r_{mc} = n_{mc}, \qquad r_{ms} = n_{ms}, \qquad m = 2, 3, \ldots, M \tag{5-4-34}$$

The additive noise components $\{n_{mc}\}$ and $\{n_{ms}\}$ are mutually statistically independent zero-mean gaussian variables with equal variance $\sigma^2 = \frac{1}{2}N_0$. Thus the pdfs of the random variables at the input to the detector are

$$p_{\mathbf{r}_1}(r_{1c}, r_{1s}) = \frac{1}{2\pi\sigma^2}\exp\left(-\frac{r_{1c}^2 + r_{1s}^2 + \mathscr{E}_s}{2\sigma^2}\right) I_0\left(\frac{\sqrt{\mathscr{E}_s(r_{1c}^2 + r_{1s}^2)}}{\sigma^2}\right) \tag{5-4-35}$$

$$p_{\mathbf{r}_m}(r_{mc}, r_{ms}) = \frac{1}{2\pi\sigma^2}\exp\left(-\frac{r_{mc}^2 + r_{ms}^2}{2\sigma^2}\right), \qquad m = 2, 3, \ldots, M \tag{5-4-36}$$

Let us make a change in variables in the joint pdfs given by (5-4-35) and (5-4-36). We define the normalized variables

$$R_m = \frac{\sqrt{r_{mc}^2 + r_{ms}^2}}{\sigma}$$
$$\Theta_m = \tan^{-1}\frac{r_{ms}}{r_{mc}} \tag{5-4-37}$$

Clearly, $r_{mc} = \sigma R_m \cos\Theta_m$ and $r_{ms} = \sigma R_m \sin\Theta_m$. The Jacobian of this transformation is

$$|\mathbf{J}| = \begin{vmatrix} \sigma\cos\Theta_m & \sigma\sin\Theta_m \\ -\sigma R_m \sin\Theta_m & \sigma R_m \cos\Theta_m \end{vmatrix} = \sigma^2 R_m \tag{5-4-38}$$

Consequently,

$$p(R_1, \Theta_1) = \frac{R_1}{2\pi}\exp\left[-\frac{1}{2}\left(R_1^2 + \frac{2\mathscr{E}_s}{N_0}\right)\right] I_0\left(\sqrt{\frac{2\mathscr{E}_s}{N_0}}R_1\right) \tag{5-4-39}$$

$$p(R_m, \Theta_m) = \frac{R_m}{2\pi}\exp(-\tfrac{1}{2}R_m^2), \qquad m = 2, 3, \ldots, M \tag{5-4-40}$$

Finally, by averaging $p(R_m, \Theta_m)$ over Θ_m, the factor of 2π is eliminated from (5-4-39) and (5-4-40). Thus, we find that R_1 has a Rice probability distribution and R_m, $m = 2, 3, \ldots, M$, are each Rayleigh-distributed.

The probability of a correct decision is simply the probability that $R_1 > R_2$, and $R_1 > R_3, \ldots$, and $R_1 > R_m$. Hence,

$$\begin{aligned} P_c &= P(R_2 < R_1, R_3 < R_1, \ldots, R_M < R_1) \\ &= \int_0^\infty P(R_2 < R_1, R_3 < R_1, \ldots, R_M < R_1 \mid R_1 = x) p_{R_1}(x)\, dx \end{aligned} \tag{5-4-41}$$

Because the random variables R_m, $m = 2, 3, \ldots, M$, are statistically iid, the joint probability in (5-4-41) conditioned on R_1 factors into a product of $M - 1$ identical terms. Thus,

$$P_c = \int_0^\infty [P(R_2 < R_1 \mid R_1 = x)]^{M-1} p_{R_1}(x)\, dx \tag{5-4-42}$$

where

$$\begin{aligned} P(R_2 < R_1 \mid R_1 = x) &= \int_0^x p_{R_2}(r_2)\, dr_2 \\ &= 1 - e^{-x^2/2} \end{aligned} \tag{5-4-43}$$

The $(M-1)$th power of (5-4-43) may be expressed as

$$(1 - e^{-x^2/2})^{M-1} = \sum_{n=0}^{M-1} (-1)^n \binom{M-1}{n} e^{-nx^2/2} \tag{5-4-44}$$

Substitution of this result into (5-4-42) and integration over x yields the probability of a correct decision as

$$P_C = \sum_{n=0}^{M-1} (-1)^n \binom{M-1}{n} \frac{1}{n+1} \exp\left[-\frac{n\mathscr{E}_s}{(n+1)N_0}\right] \tag{5-4-45}$$

where $\mathscr{E}_s/N_0$ is the SNR per symbol. Then, the probability of a symbol error, which is $P_M = 1 - P_c$, becomes

$$P_M = \sum_{n=1}^{M-1} (-1)^{n+1} \binom{M-1}{n} \frac{1}{n+1} \exp\left[-\frac{nk\mathscr{E}_b}{(n+1)N_0}\right] \tag{5-4-46}$$

where $\mathscr{E}_b/N_0$ is the SNR per bit.

For binary orthogonal signals ($M = 2$), (5-4-46) reduces to the simple form

$$P_2 = \tfrac{1}{2} e^{-\mathscr{E}_b/2N_0} \tag{5-4-47}$$

For $M > 2$, we may compute the probability of a bit error by making use of the relationship

$$P_b = \frac{2^{k-1}}{2^k - 1} P_M \tag{5-4-48}$$

which was established in Section 5-2. Figure 5-4-5 shows the bit-error probability as a function of the SNR per bit γ_b for $M = 2$, 4, 8, 16, and 32. Just as in the case of coherent detection of M-ary orthogonal signals (see Section 5-2-2), we observe that for any given bit-error probability, the SNR per bit decreases as M increases. It will be shown in Chapter 7 that, in the limit as $M \to \infty$ (or $k = \log_2 M \to \infty$), the probability of a bit error P_b can be made

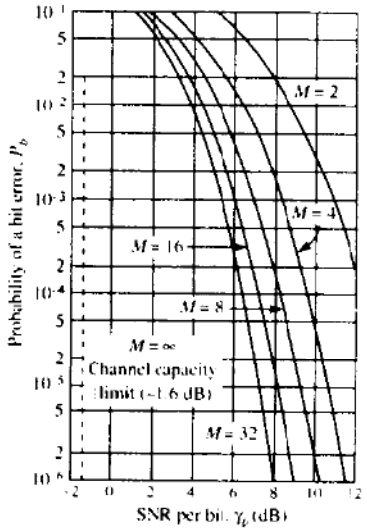


FIGURE 5-4-5 Probability of a bit error for noncoherent detection of orthogonal signals.

arbitrarily small provided that the SNR per bit is greater than the Shannon limit of -1.6 dB. The cost for increasing M is the bandwidth required to transmit the signals. For M-ary FSK, the frequency separation between adjacent frequencies is $\Delta f = 1/T$ for signal orthogonality. The bandwidth required for the M signals is $W = M\,\Delta f = M/T$. Also, the bit rate is $R = k/T$, where $k = \log_2 M$. Therefore, the bit-rate-to-bandwidth ratio is

$$\frac{R}{W} = \frac{\log_2 M}{M} \tag{5-4-49}$$

5-4-4 Probability of Error for Envelope Detection of Correlated Binary Signals

In this section, we consider the performance of the envelope detector for binary, equal-energy correlated signals. When the two signals are correlated, the input to the detector are the complex-valued random variables given by (5-4-10). We assume that the detector bases its decision on the envelopes $|r_1|$ and $|r_2|$, which are correlated (statistically dependent). The marginal pdfs of $R_1 = |r_1|$ and $R_2 = |r_2|$ are Ricean distributed, and may be expressed as

$$p(R_m) = \begin{cases} \dfrac{R_m}{2\mathscr{E}N_0} \exp\left(-\dfrac{R_m^2 + \beta_m^2}{4\mathscr{E}N_0}\right) I_0\left(\dfrac{\beta_m R_m}{2\mathscr{E}N_0}\right) & (R_m > 0) \\ 0 & (R_m < 0) \end{cases} \tag{5-4-50}$$

$m = 1, 2$, where $\beta_1 = 2\mathscr{E}$ and $\beta_2 = 2\mathscr{E}|\rho|$, based on the assumption that signal $s_1(t)$ was transmitted.

Since R_1 and R_2 are statistically dependent as a consequence of the nonorthogonality of the signals, the probability of error may be obtained by evaluating the double integral

$$P_b = P(R_2 > R_1) = \int_0^\infty \int_{x_1}^\infty p(x_1, x_2)\, dx_1\, dx_2 \tag{5-4-51}$$

where $p(x_1, x_2)$ is the joint pdf of the envelopes R_1 and R_2. This approach was first used by Helstrom (1955), who determined the joint pdf of R_1 and R_2 and evaluated the double integral in (5-4-51).

An alternative approach is based on the observation that the probability of error may also be expressed as

$$P_b = P(R_2 > R_1) = P(R_2^2 > R_1^2) = P(R_2^2 - R_1^2 > 0) \tag{5-4-52}$$

But $R_2^2 - R_1^2$ is a special case of a general quadratic form in complex-valued gaussian random variables, treated later in Appendix B. For the special case under consideration, the derivation yields the error probability in the form

$$P_b = Q_1(a, b) - \tfrac{1}{2} e^{-(a^2+b^2)/2} I_0(ab) \tag{5-4-53}$$

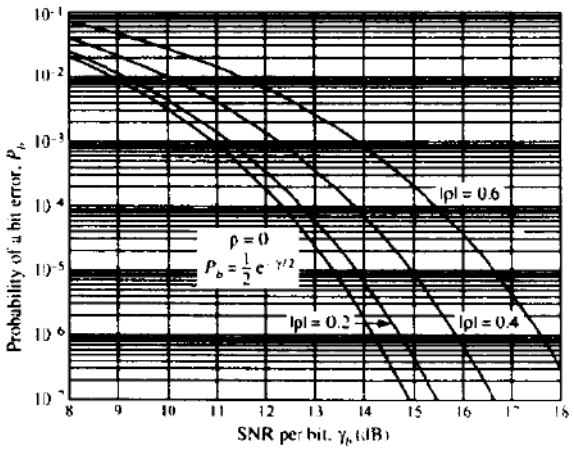


FIGURE 5-4-6 Probability of error for noncoherent detection.

where

$$a = \sqrt{\frac{\mathscr{E}_b}{2N_0}(1 - \sqrt{1 - |\rho|^2})}$$
$$b = \sqrt{\frac{\mathscr{E}_b}{2N_0}(1 + \sqrt{1 - |\rho|^2})} \tag{5-4-54}$$

$Q_1(a, b)$ is the Q function defined in (2-1-123) and $I_0(x)$ is the modified Bessel function of order zero.

The error probability P_b is illustrated in Fig. 5-4-6 for several values of $|\rho|$. P_b is minimized when $\rho = 0$; that is, when the signals are orthogonal. For this case, $a = 0$, $b = \sqrt{\mathscr{E}_b/N_0}$, and (5-4-53) reduces to

$$P_b = Q\left(0, \sqrt{\frac{\mathscr{E}_b}{N_0}}\right) - \tfrac{1}{2}e^{-\mathscr{E}_b/2N_0} \tag{5-4-55}$$

From the definition of $Q_1(a, b)$ in (2-1-123), it follows that

$$Q_1\left(0, \sqrt{\frac{\mathscr{E}_b}{N_0}}\right) = e^{-\mathscr{E}_b/2N_0}$$

Substitution of these relations into (5-4-55) yields the desired result given previously in (5-4-47). On the other hand, when $|\rho| = 1$, the error probability in (5-4-53) becomes $P_b = \frac{1}{2}$, as expected.

5-5 REGENERATIVE REPEATERS AND LINK BUDGET ANALYSIS

In the transmission of digital signals through an AWGN channel, we have observed that the performance of the communication system, measured in terms of the probability of error, depends solely on the received SNR, $\mathscr{E}_b/N_0$.

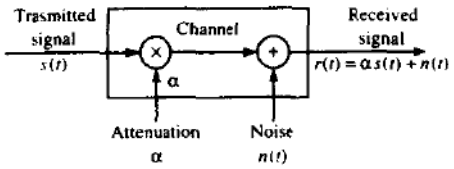


FIGURE 5-5-1 Mathematical model of channel with attenuation and additive noise.

where $\mathscr{E}_b$ is the transmitted energy per bit and $\frac{1}{2}N_0$ is the power spectral density of the additive noise. Hence, the additive noise ultimately limits the performance of the communication system.

In addition to the additive noise, another factor that affects the performance of a communication system is channel attenuation. All physical channels, including wire lines and radio channels, are lossy. Hence, the signal is attenuated as it travels through the channel. The simple mathematical model for the attenuation shown in Fig. 5-5-1 may be used for the channel. Consequently, if the transmitted signal is $s(t)$, the received signal, with $0 < \alpha \leq 1$ is

$$r(t) = \alpha s(t) + n(t) \tag{5-5-1}$$

Then, if the energy in the transmitted signal is $\mathscr{E}_b$, the energy in the received signal is $\alpha^2\mathscr{E}_b$. Consequently, the received signal has an SNR $\alpha^2\mathscr{E}_b/N_0$. Hence, the effect of signal attenuation is to reduce the energy in the received signal and thus to render the communication system more vulnerable to additive noise.

In analog communication systems, amplifiers called repeaters are used to periodically boost the signal strength in transmission through the channel. However, each amplifier also boosts the noise in the system. In contrast, digital communication systems allow us to detect and regenerate a clean (noise-free) signal in a transmission channel. Such devices, called *regenerative repeaters*, are frequently used in wireline and fiber optic communication channels.

5-5-1 Regenerative Repeaters

The front end of each regenerative repeater consists of a demodulator/detector that demodulates and detects the transmitted digital information sequence sent by the preceding repeater. Once detected, the sequence is passed to the transmitter side of the repeater, which maps the sequence into signal waveforms that are transmitted to the next repeater. This type of repeater is called a regenerative repeater.

Since a noise-free signal is regenerated at each repeater, the additive noise does not accumulate. However, when errors occur in the detector of a repeater, the errors are propagated forward to the following repeaters in the channel. To evaluate the effect of errors on the performance of the overall system, suppose that the modulation is binary PAM, so that the probability of

a bit error for one hop (signal transmission from one repeater to the next repeater in the chain) is

$$P_b = Q\left(\sqrt{\frac{2\mathscr{E}_b}{N_0}}\right)$$

Since errors occur with low probability, we may ignore the probability that any one bit will be detected incorrectly more than once in transmission through a channel with K repeaters. Consequently, the number of errors will increase linearly with the number of regenerative repeaters in the channel, and therefore, the overall probability of error may be approximated as

$$P_b \approx KQ\left(\sqrt{\frac{2\mathscr{E}_b}{N_0}}\right) \tag{5-5-2}$$

In contrast, the use of K analog repeaters in the channel reduces the received SNR by K, and hence, the bit error probability is

$$P_b \approx Q\left(\sqrt{\frac{2\mathscr{E}_b}{KN_0}}\right) \tag{5-5-3}$$

Clearly, for the same probability of error performance, the use of regenerative repeaters results in a significant saving in transmitter power compared with analog repeaters. Hence, in digital communication systems, regenerative repeaters are preferable. However, in wireline telephone channels that are used to transmit both analog and digital signals, analog repeaters are generally employed.

Example 5-5-1

A binary digital communication system transmits data over a wireline channel of length 1000 km. Repeaters are used every 10 km to offset the effect of channel attenuation. Let us determine the $\mathscr{E}_b/N_0$ that is required to achieve a probability of a bit error of 10^{-5} if (a) analog repeaters are employed, and (b) regenerative repeaters are employed.

The number of repeaters used in the system is $K = 100$. If regenerative repeaters are used, the $\mathscr{E}_b/N_0$ obtained from (5-5-2) is

$$10^{-5} = 100Q\left(\sqrt{\frac{2\mathscr{E}_b}{N_0}}\right)$$

$$10^{-7} = Q\left(\sqrt{\frac{2\mathscr{E}_b}{N_0}}\right)$$

which yields approximately 11.3 dB. If analog repeaters are used, the $\mathscr{E}_b/N_0$ obtained from (5-5-3) is

$$10^{-5} = Q\left(\sqrt{\frac{2\mathscr{E}_b}{100N_0}}\right)$$

which yields $\mathscr{E}_b/N_0 \approx 29.6$ dB. Hence, the difference in the required SNR is

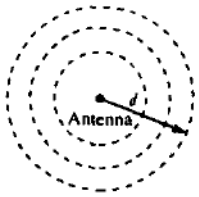


FIGURE 5-5-2 Isotropically radiating antenna.

about 18.3 dB, or approximately 70 times the transmitter power of the digital communication system.

5-5-2 Communication Link Budget Analysis

In the design of radio communications systems that transmit over line-of-sight microwave channels and satellite channels, the system designer must specify the size of the transmit and receive antennas, the transmitted power, and the SNR required to achieve a given level of performance at some desired data rate. The system design procedure is relatively straightforward and is outlined below.

Let us begin with a transmit antenna that radiates isotropically in free space at a power level of P_T watts as shown in Fig. 5-5-2. The power density at a distance d from the antenna is $P_T/4\pi d^2$ W/m^2. If the transmitting antenna has some directivity in a particular direction, the power density in that direction is increased by a factor called the antenna gain and denoted by G_T. In such a case, the power density at distance d is $P_TG_T/4\pi d^2$ W/m^2. The product P_TG_T is usually called the *effective radiated power* (ERP or EIRP), which is basically the radiated power relative to an isotropic antenna, for which $G_T = 1$.

A receiving antenna pointed in the direction of the radiated power gathers a portion of the power that is proportional to its cross-sectional area. Hence, the received power extracted by the antenna may be expressed as

$$P_R = \frac{P_TG_TA_R}{4\pi d^2} \tag{5-5-4}$$

where A_R is the *effective area of the antenna.* From electromagnetic field theory, we obtain the basic relationship between the gain G_R of an antenna and its effective area as

$$A_R = \frac{G_R\lambda^2}{4\pi} \text{ m}^2 \tag{5-5-5}$$

where $\lambda = c/f$ is the wavelength of the transmitted signal, c is the speed of light (3×10^8 m/s), and f is the frequency of the transmitted signal.

If we substitute (5-5-5) for A_R into (5-5-4), we obtain an expression for the received power in the form

$$P_R = \frac{P_TG_TG_R}{(4\pi d/\lambda)^2} \tag{5-5-6}$$

The factor

$$L_s = \left(\frac{\lambda}{4\pi d}\right)^2 \tag{5-5-7}$$

is called the *free-space path loss*. If other losses, such as atmospheric losses, are encountered in the transmission of the signal, they may be accounted for by introducing an additional loss factor, say L_a. Therefore, the received power may be written in general as

$$P_R = P_T G_T G_R L_s L_a \tag{5-5-8}$$

As indicated above, the important characteristics of an antenna are its gain and its effective area. These generally depend on the wavelength of the radiated power and the physical dimensions of the antenna. For example, a parabolic (dish) antenna of diameter D has an effective area

$$A_R = \tfrac{1}{4}\pi D^2 \eta \tag{5-5-9}$$

where $\frac{1}{4}\pi D^2$ is the physical area and η is the *illumination efficiency factor*, which falls in the range $0.5 \leq \eta \leq 0.6$. Hence, the antenna gain for a parabolic antenna of diameter D is

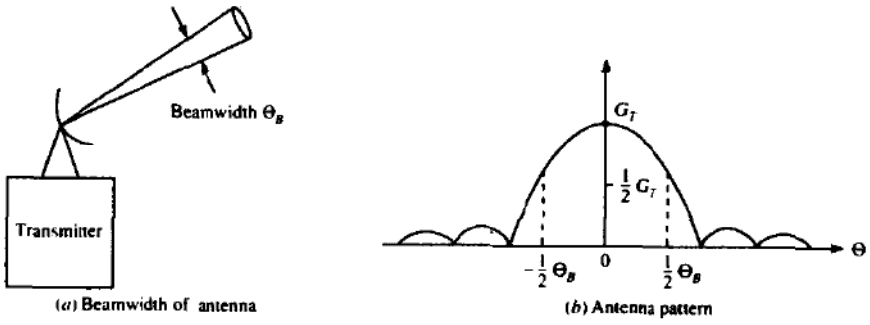
$$G_R = \eta\left(\frac{\pi D}{\lambda}\right)^2 \tag{5-5-10}$$

As a second example, a horn antenna of physical area A has an efficiency factor of 0.8, an effective area of $A_R = 0.8A$, and an antenna gain of

$$G_R = \frac{10A}{\lambda^2} \tag{5-5-11}$$

Another parameter that is related to the gain (directivity) of an antenna is its beamwidth, which we denote as Θ_B and which is illustrated graphically in Fig. 5-5-3. Usually, the beamwidth is measured as the -3 dB width of the

FIGURE 5-5-3 Antenna beamwidth and pattern.



(*a*) Beamwidth of antenna

(*b*) Antenna pattern

antenna pattern. For example, the −3 dB beamwidth of a parabolic antenna is approximately

$$\Theta_B = 70(\lambda/D)^\circ \tag{5-5-12}$$

so that G_T is inversely proportional to Θ_B^2. That is, a decrease of the beamwidth by a factor of two, which is obtained by doubling the diameter D, increases the antenna gain by a factor of four (6 dB).

Based on the general relationship for the received signal power given by (5-5-8), the system designer can compute P_R from a specification of the antenna gains and the distance between the transmitter and the receiver. Such computations are usually done on a power basis, so that

$$(P_R)_{\text{dB}} = (P_T)_{\text{dB}} + (G_T)_{\text{dB}} + (G_R)_{\text{dB}} + (L_s)_{\text{dB}} + (L_a)_{\text{dB}} \tag{5-5-13}$$

Example 5-5-2

Suppose that we have a satellite in geosynchronous orbit (36 000 km above the earth's surface) that radiates 100 W of power, i.e., 20 dB above 1 W (20 dBW). The transmit antenna has a gain of 17 dB, so that the ERP = 37 dBW. Also, suppose that the earth station employs a 3 m parabolic antenna and that the downlink is operating at a frequency of 4 GHz. The efficiency factor is $\eta = 0.5$. By substituting these numbers into (5-5-10), we obtain the value of the antenna gain as 39 dB. The free-space path loss is

$$L_s = 195.6 \text{ dB}$$

No other losses are assumed. Therefore, the received signal power is

$$\begin{aligned}(P_R)_{\text{dB}} &= 20 + 17 + 39 - 195.6 \\ &= -119.6 \text{ dBW}\end{aligned}$$

or, equivalently,

$$P_R = 1.1 \times 10^{-12} \text{ W}$$

To complete the link budget computation, we must also consider the effect of the additive noise at the receiver front end. Thermal noise that arises at the receiver front end has a relatively flat power density spectrum up to about 10^{12} Hz, and is given as

$$N_0 = k_B T_0 \text{ W/Hz} \tag{5-5-14}$$

where k_B is Boltzmann's constant (1.38×10^{-23} W s/K) and T_0 is the noise temperature in Kelvin. Therefore, the total noise power in the signal bandwidth W is $N_0 W$.

The performance of the digital communications system is specified by the $\mathscr{E}_b/N_0$ required to keep the error rate performance below some given value. Since

$$\frac{\mathscr{E}_b}{N_0} = \frac{T_b P_R}{N_0} = \frac{1}{R}\frac{P_R}{N_0} \tag{5-5-15}$$

it follows that

$$\frac{P_R}{N_0} = R\left(\frac{\mathscr{E}_b}{N_0}\right)_{\text{req}} \tag{5-5-16}$$

where $(\mathscr{E}_b/N_0)_{\text{req}}$ is the required SNR per bit. Hence, if we have P_R/N_0 and the required SNR per bit, we can determine the maximum data rate that is possible.

Example 5-5-3

For the link considered in Example 5-5-2, the received signal power is

$$P_R = 1.1 \times 10^{-12}\ \text{W} \quad (-119.6\ \text{dBW})$$

Now, suppose the receiver front end has a noise temperature of 300 K, which is typical for receiver in the 4 GHz range. Then

$$N_0 = 4.1 \times 10^{-21}\ \text{W/Hz}$$

or, equivalently, −203.9 dBW/Hz. Therefore,

$$\frac{P_R}{N_0} = -119.6 + 203.9 = 84.3\ \text{dB Hz}$$

If the required SNR per bit is 10 dB then, from (5-5-16), we have the available rate as

$$\begin{aligned} R_{\text{dB}} &= 84.3 - 10 \\ &= 74.3\ \text{dB} \quad \text{(with respect to 1 bit/s)} \end{aligned}$$

This corresponds to a rate of 26.9 megabits/s, which is equivalent to about 420 PCM channels, each operating at 64 000 bits/s.

It is a good idea to introduce some safety margin, which we shall call the *link margin* M_{dB}, in the above computations for the capacity of the communication link. Typically, this may be selected as $M_{\text{dB}} = 6$ dB. Then, the link budget computation for the link capacity may be expressed in the simple form

$$\begin{aligned} R_{\text{dB}} &= \left(\frac{P_R}{N_0}\right)_{\text{dB Hz}} - \left(\frac{\mathscr{E}_b}{N_0}\right)_{\text{req}} - M_{\text{dB}} \\ &= (P_T)_{\text{dBW}} + (G_T)_{\text{dB}} + (G_R)_{\text{dB}} \\ &\quad + (L_a)_{\text{dB}} + (L_s)_{\text{dB}} - \left(\frac{\mathscr{E}_b}{N_0}\right)_{\text{req}} - M_{\text{db}} \end{aligned} \tag{5-5-17}$$

BIBLIOGRAPHICAL NOTES AND REFERENCES

In the derivation of the optimum demodulator for a signal corrupted by AWGN, we applied mathematical techniques that were originally used in deriving optimum receiver structures for radar signals. For example, the

matched filter was first proposed by North (1943) for use in radar detection, and is sometimes called the North filter. An alternative method for deriving the optimum demodulator and detector is the Karhunen–Loeve expansion, which is described in the classical texts by Davenport and Root (1958), Helstrom (1968), and Van Trees (1968). Its use in radar detection theory is described in the paper by Kelly *et al.* (1960). These detection methods are based on the hypothesis testing methods developed by statisticians, e.g., Neyman and Pearson (1933) and Wald (1947).

The geometric approach to signal design and detection, which was presented in the context of digital modulation and which has its roots in Shannon's original work, is conceptually appealing and is now widely used since its introduction in the text by Wozencraft and Jacobs (1965).

Design and analysis of signal constellations for the AWGN channel have received considerable attention in the technical literature. Of particular significance is the performance analysis of two-dimensional (QAM) signal constellations that has been treated in the papers of Cahn (1960), Hancock and Lucky (1960), Campopiano and Glazer (1962), Lucky and Hancock (1962), Salz *et al.* (1971), Simon and Smith (1973), Thomas *et al.* (1974), and Foschini *et al.* (1974). Signal design based on multidimensional signal constellations has been described and analyzed in the paper by Gersho and Lawrence (1984).

The Viterbi algorithm was devised by Viterbi (1967) for the purpose of decoding convolutional codes. Its use as the optimal maximum-likelihood sequence detection algorithm for signals with memory was described by Forney (1972) and Omura (1971). Its use for carrier modulated signals was considered by Ungerboeck (1974) and MacKenchnie (1973). It was subsequently applied to the demodulation of CPM by Aulin and Sundberg (1981a, b) and others.

PROBLEMS

5-1 A matched filter has the frequency response

$$H(f) = \frac{1 - e^{-j2\pi fT}}{j2\pi f}$$

a Determine the impulse response $h(t)$ corresponding to $H(f)$.
b Determine the signal waveform to which the filter characteristic is matched.

5-2 Consider the signal

$$s(t) = \begin{cases} (A/T)t \cos 2\pi f_c t & (0 \leq t \leq T) \\ 0 & \text{(otherwise)} \end{cases}$$

a Determine the impulse response of the matched filter for the signal.
b Determine the output of the matched filter at $t = T$.
c Suppose the signal $s(t)$ is passed through a correlator that correlates the input $s(t)$ with $s(t)$. Determine the value of the correlator output at $t = T$. Compare your result with that in (b).

5-3 This problem deals with the characteristics of a DPSK signal.

a Suppose we wish to transmit the data sequence

$$1\ 1\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 1\ 1\ 0$$

by binary DPSK Let $s(t) = A\cos(2\pi f_c t + \theta)$ represent the transmitted signal in any signaling interval of duration T. Give the phase of the transmitted signal for the data sequence. Begin with $\theta = 0$ for the phase of the first bit to be transmitted.

b If the data sequence is uncorrelated, determine and sketch the power density spectrum of the signal transmitted by DPSK.

5-4 A binary digital communication system employs the signals

$$s_0(t) = 0, \quad 0 \leq t \leq T$$
$$s_1(t) = A, \quad 0 \leq t \leq T$$

for transmitting the information. This is called *on–off signaling*. The demodulator cross-correlates the received signal $r(t)$ with $s(t)$ and samples the output of the correlator at $t = T$.

a Determine the optimum detector for an AWGN channel and the optimum threshold, assuming that the signals are equally probable.

b Determine the probability of error as a function of the SNR. How does on–off signaling compare with antipodal signaling?

5-5 The correlation metrics given by (5-1-44) are

$$C(\mathbf{r}, \mathbf{s}_m) = 2\sum_{n=1}^{N} r_n s_{mn} - \sum_{n=1}^{N} s_{mn}^2, \quad m = 1, 2, \ldots, M$$

where

$$r_n = \int_0^T r(t) f_n(t)\, dt$$

$$s_{mn} = \int_0^T s_m(t) f_n(t)\, dt$$

Show that the correlation metrics are equivalent to the metrics

$$C(\mathbf{r}, \mathbf{s}_m) = 2\int_0^T r(t) s_m(t)\, dt - \int_0^T s_m^2(t)\, dt$$

5-6 Consider the equivalent lowpass (complex-valued) signal $s_l(t)$, $0 \leq t \leq T$, with energy

$$\mathcal{E} = \frac{1}{2}\int_0^T |s_l(t)|^2\, dt$$

Suppose that this signal is corrupted by AWGN, which is represented by its equivalent lowpass form $z(t)$. Hence, the observed signal is

$$r_l(t) = s_l(t) + z(t), \quad 0 \leq t \leq T$$

The received signal is passed through a filter that has an (equivalent lowpass) impulse response $h_l(t)$. Determine $h_l(t)$ so that the filter maximizes the SNR at its output (at $t = T$).

5-7 Let $z(t) = x(t) + jy(t)$ be a complex-valued, zero-mean white gaussian noise

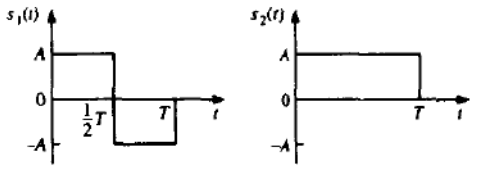


FIGURE P5-8

process with autocorrelation function $\phi_{zz}(\tau) = N_0\delta(\tau)$. Let $f_m(t)$, $m = 1, 2, \ldots, M$, be a set of M orthogonal equivalent lowpass waveforms defined on the interval $0 \leq t \leq T$. Define

$$N_{mr} = \text{Re}\left[\int_0^T z(t) f_m^*(t)\, dt\right], \quad m = 1, 2, \ldots, M$$

a Determine the variance of N_{mr}.
b Show that $E(N_{mr}N_{kr}) = 0$ for $k \neq m$.

5-8 The two equivalent lowpass signals shown in Fig. P5-8 are used to transmit a binary sequence over an additive white gaussian noise channel. The received signal can be expressed as

$$r_i(t) = s_i(t) + z(t), \quad 0 \leq t \leq T, \quad i = 1, 2$$

where $z(t)$ is a zero-mean gaussian noise process with autocorrelation function

$$\phi_{zz}(\tau) = \tfrac{1}{2}E[z^*(t)z(t+\tau)] = N_0\delta(\tau)$$

a Determine the transmitted energy in $s_1(t)$ and $s_2(t)$ and the cross-correlation coefficient ρ_{12}.
b Suppose the receiver is implemented by means of coherent detection using two matched filters, one matched to $s_1(t)$ and the other to $s_2(t)$. Sketch the equivalent lowpass impulse responses of the matched filters.
c Sketch the noise-free response of the two matched filters when the transmitted signal is $s_2(t)$.
d Suppose the receiver is implemented by means of two cross-correlators (multipliers followed by integrators) in parallel. Sketch the output of each integrator as a function of time for the interval $0 \leq t \leq T$ when the transmitted signal is $s_2(t)$.
e Compare the sketches in (c) and (d). Are they the same? Explain briefly.
f From your knowledge of the signal characteristics, give the probability of error for this binary communications system.

5-9 Suppose that we have a complex-valued gaussian random variable $z = x + jy$, where (x, y) are statistically independent variables with zero mean and variance $E(x^2) = E(y^2) = \sigma^2$. Let

$$r = z + m, \quad \text{where } m = m_r + jm_i$$

and define r as

$$r = a + jb$$

Clearly, $a = x + m_r$ and $b = y + m_i$. Determine the following probability density functions:
a $p(a, b)$;

b $p(u, \phi)$, where $u = \sqrt{a^2 + b^2}$ and $\phi = \tan^{-1} b/a$;
c $p(u)$.

Note: In (b) it is convenient to define $\theta = \tan^{-1}(m_i/m_r)$ so that

$$m_i = \sqrt{m_r^2 + m_i^2}\cos\theta, \qquad m_i = \sqrt{m_r^2 + m_i^2}\sin\theta.$$

Furthermore, you must use the relation

$$\frac{1}{2\pi}\int_0^{2\pi} e^{\alpha\cos(\phi-\theta)}\,d\phi = I_0(\alpha) = \sum_{n=0}^{\infty}\frac{\alpha^{2n}}{2^{2n}(n!)^2}$$

where $I_0(\alpha)$ is the modified Bessel function of order zero.

5-10 A ternary communication system transmits one of three signals, $s(t)$, 0, or $-s(t)$, every T seconds. The received signal is either $r_1(t) = s(t) + z(t)$, $r_2(t) = z(t)$, or $r_3(t) = -s(t) + z(t)$, where $z(t)$ is white gaussian noise with $E[z(t)] = 0$ and $\phi_{zz}(\tau) = \frac{1}{2}E[z(t)z^*(\tau)] = N_0\delta(t-\tau)$. The optimum receiver computes the correlation metric

$$U = \mathrm{Re}\left[\int_0^T r(t)s^*(t)\,dt\right]$$

and compares U with a threshold A and a threshold $-A$. If $U > A$, the decision is made that $s(t)$ was sent. If $U < -A$, the decision is made in favor of $-s(t)$. If $-A < U < A$, the decision is made in favor of 0.

a Determine the three conditional probabilities of error P_e given that $s(t)$ was sent, P_e given that $-s(t)$ was sent, and P_e given that 0 was sent.
b Determine the average probability of error P_e as a function of the threshold A, assuming that the three symbols are equally probable a priori.
c Determine the value of A that minimizes P_e.

5-11 The two equivalent lowpass signals shown in Fig. P5-11 are used to transmit a binary information sequence. The transmitted signals, which are equally probable, are corrupted by additive zero-mean white gaussian noise having an equivalent lowpass representation $z(t)$ with an autocorrelation function

$$\begin{aligned}\phi_{zz}(\tau) &= \tfrac{1}{2}E[z^*(t)z(t+\tau)]\\ &= N_0\delta(\tau)\end{aligned}$$

a What is the transmitted signal energy?
b What is the probability of a binary digit error if coherent detection is employed at the receiver?
c What is the probability of a binary digit error if noncoherent detection is employed at the receiver?

5-12 In Section 4-3-1 it was shown that the minimum frequency separation for orthogonality of binary FSK signals with coherent detection is $\Delta f = 1/2T$.

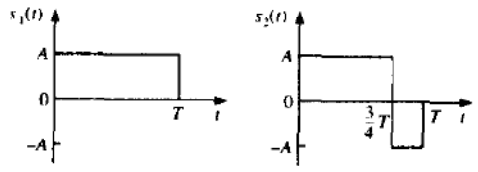


FIGURE P5-11

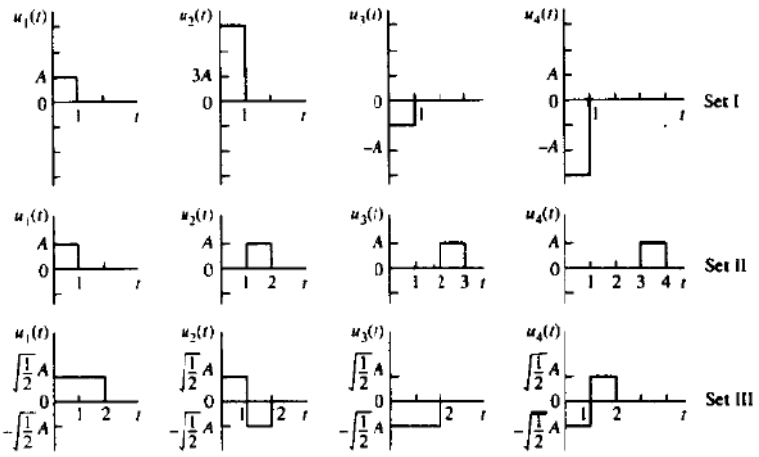


FIGURE P5-13

However, a lower error probability is possible with coherent detection of FSK if Δf is increased beyond $1/2T$. Show that the optimum value of Δf is $0.715/T$ and determine the probability of error for this value of Δf.

5-13 The equivalent lowpass waveforms for three signal sets are shown in Fig. P5-13. Each set may be used to transmit one of four equally probable messages over an additive white gaussian noise channel. The equivalent lowpass noise $z(t)$ has zero mean and autocorrelation function $\phi_{zz}(\tau) = N_0\delta(\tau)$.

- **a** Classify the signal waveforms in sets I, II, and III. In other words, state the category or class to which each signal set belongs.
- **b** What is the *average* transmitted energy for each signal set?
- **c** For signal set I, specify the average probability of error if the signals are detected coherently.
- **d** For signal set II, give a union bound on the probability of a symbol error if the detection is performed (i) coherently and (ii) noncoherently.
- **e** Is it possible to use noncoherent detection on signal set III? Explain.
- **f** Which signal set or signal sets would you select if you wished to achieve a ratio of bit rate to bandwidth (R/W) of at least 2. *Briefly* explain your answer.

5-14 Consider a quaternary ($M = 4$) communication system that transmits, every T seconds, one of four equally probable signals: $s_1(t)$, $-s_1(t)$, $s_2(t)$, $-s_2(t)$. The signals $s_1(t)$ and $s_2(t)$ are orthogonal with equal energy. The additive noise is white gaussian with zero mean and autocorrelation function $\phi_{zz}(\tau) = N_0\delta(\tau)$. The demodulator consists of two filters matched to $s_1(t)$ and $s_2(t)$, and their outputs at the sampling instant are U_1 and U_2. The detector bases its decision on the following rule:

$$U_1 > |U_2| \Rightarrow s_1(t), \qquad U_1 < -|U_2| \Rightarrow -s_1(t)$$
$$U_2 > |U_1| \Rightarrow s_2(t), \qquad U_2 < -|U_1| \Rightarrow -s_2(t)$$

Since the signal set is biorthogonal, the error probability is given by $(1 - P_c)$ where P_c is given by (5-2-34). Express this error probability in terms of a single integral

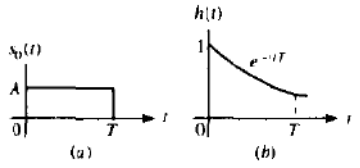


FIGURE P5-15

and, thus, show that the symbol error probability for a biorthogonal signal set with $M = 4$ is identical to that for four-phase PSK. *Hint*: A change in variables from U_1 and U_2 to $W_1 = U_1 + U_2$ and $W_2 = U_1 - U_2$ simplifies the problem.

5-15 The input $s(t)$ to a bandpass filter is

$$s(t) = \text{Re}\,[s_0(t)e^{j2\pi f_c t}]$$

where $s_0(t)$ is a rectangular pulse as shown in Fig. P5-15(*a*).

a Determine the output $y(t)$ of the bandpass filter for all $t \geq 0$ if the impulse response of the filter is

$$g(t) = \text{Re}\,[2h(t)e^{j2\pi f_c t}]$$

where $h(t)$ is an exponential as shown in Fig. 5-15(*b*).

b Sketch the *equivalent lowpass output* of the filter.

c When would you sample the output of the filter if you wished to have the maximum output at the sampling instant? What is the value of the maximum output?

d Suppose that in addition to the input signal $s(t)$, there is additive white gaussian noise

$$n(t) = \text{Re}\,[z(t)e^{j2\pi f_c t}]$$

where $\phi_{zz}(\tau) = N_0\delta(\tau)$. At the sampling instant determined in (c), the signal sample is corrupted by an additive gaussian noise term. Determine its mean and variance.

e What is the signal-to-noise ratio γ of the sampled output?

f Determine the signal-to-noise ratio when $h(t)$ is the matched filter to $s(t)$ and compare this result with the value of γ obtained in (e).

5-16 Consider the octal signal point constellations in Fig. P5-16.

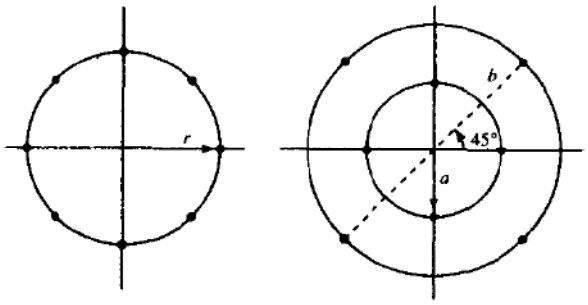


FIGURE P5-16

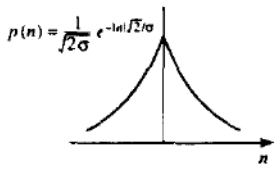


FIGURE P5-19

a The nearest-neighbor signal points in the 8-QAM signal constellation are separated in distance by A units. Determine the radii a and b of the inner and outer circles.
b The adjacent signal points in the 8-PSK are separated by a distance of A units. Determine the radius r of the circle.
c Determine the average transmitter powers for the two signal constellations and compare the two powers. What is the relative power advantage of one constellation over the other? (Assume that all signal points are equally probable.)

5-17 Consider the 8-point QAM signal constellation shown in Fig. P5-16.
a Is it possible to assign three data bits to each point of the signal constellation such that nearest (adjacent) points differ in only one bit position?
b Determine the symbol rate if the desired bit rate is 90 Mbits/s.

5-18 Suppose that binary PSK is used for transmitting information over an AWGN with a power spectral density of $\frac{1}{2}N_0 = 10^{-10}$ W/Hz. The transmitted signal energy is $\mathcal{E}_b = \frac{1}{2}A^2T$, where T is the bit interval and A is the signal amplitude. Determine the signal amplitude required to achieve an error probability of 10^{-6} when the data rate is (a) 10 kbits/s, (b) 100 kbits/s, and (c) 1 Mbit/s.

5-19 Consider a signal detector with an input

$$r = \pm A + n$$

where $+A$ and $-A$ occur with equal probability and the noise variable n is characterized by the (Laplacian) pdf shown in Fig. P5-19.
a Determine the probability of error as a function of the parameters A and σ
b Determine the SNR required to achieve an error probability of 10^{-5}. How does the SNR compare with the result for a Gaussian pdf?

5-20 Consider the two 8-point QAM signal constellations shown in Fig. P5-20. The minimum distance between adjacent points is $2A$. Determine the average transmitted power for each constellation, assuming that the signal points are equally probable. Which constellation is more power-efficient?

5-21 For the QAM signal constellation shown in Fig. P5-21, determine the optimum decision boundaries for the detector, assuming that the SNR is sufficiently high so that errors only occur between adjacent points.

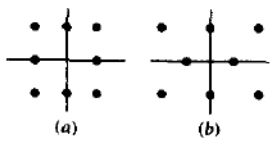


FIGURE P5-20

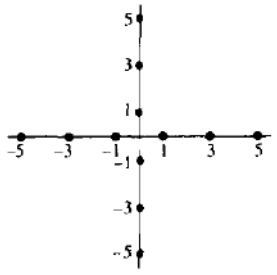


FIGURE P5-21

5-22 Specify a Gray code for the 16-QAM signal constellation shown in Fig. P5-21.

5-23 Two quadrature carriers $\cos 2\pi f_c t$ and $\sin 2\pi f_c t$ are used to transmit digital information through an AWGN channel at two different data rates, 10 kbits/s and 100 kbits/s. Determine the relative amplitudes of the signals for the two carriers so that the $\mathscr{E}_b/N_b$ for the two channels is identical.

5-24 Three messages m_1, m_2, and m_3 are to be transmitted over an AWGN channel with noise power spectral density $\frac{1}{2}N_0$. The messages are

$$s_1(t) = \begin{cases} 1 & (0 \leq t \leq T) \\ 0 & (\text{otherwise}) \end{cases}$$

$$s_2(t) = -s_3(t) = \begin{cases} 1 & (0 \leq t \leq \frac{1}{2}T) \\ -1 & (\frac{1}{2}T \leq 0 \leq T) \\ 0 & (\text{otherwise}) \end{cases}$$

a What is the dimensionality of the signal space?

b Find an appropriate basis for the signal space. [*Hint:* You can find the basis without using the Gram–Schmidt procedure.]

c Draw the signal constellation for this problem.

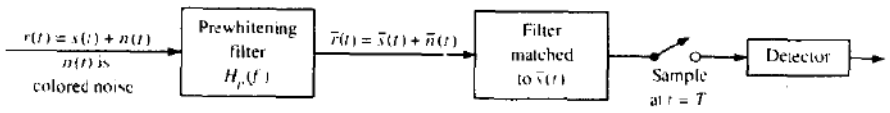
d Derive and sketch the optimal decision regions R_1, R_2, and R_3.

e Which of the three messages is more vulnerable to errors and why? In other words, which of $P(\text{error} \mid m_i \text{ transmitted})$, $i = 1, 2, 3$, is larger?

5-25 When the additive noise at the input to the demodulator is colored, the filter matched to the signal no longer maximizes the output SNR. In such a case we may consider the use of a prefilter that "whitens" the colored noise. The prefilter is followed by a filter matched to the prefiltered signal. Towards this end, consider the configuration shown in Fig. P5-25.

a Determine the frequency response characteristic of the prefilter that whitens the noise.

FIGURE P5-25



b Determine the frequency response characteristic of the filter matched to $\tilde{s}(t)$.
c Consider the prefilter and the matched filter as a single "generalized matched filter." What is the frequency response characteristic of this filter?
d Determine the SNR at the input to the detector.

5-26 Consider a digital communication system that transmits information via QAM over a voice-band telephone channel at a rate 2400 symbols/s. The additive noise is assumed to be white and gaussian.
a Determine the $\mathscr{E}_b/N_0$ required to achieve an error probability of 10^{-5} at 4800 bits/s.
b Repeat (a) for a rate of 9600 bits/s.
c Repeat (a) for a rate of 19 200 bits/s.
d What conclusions do you reach from these results?

5-27 Consider the four-phase and eight-phase signal constellations shown in Fig. P5-27. Determine the radii r_1 and r_2 of the circles such that the distance between two adjacent points in the two constellations is d. From this result, determine the additional transmitted energy required in the 8-PSK signal to achieve the same error probability as the four-phase signal at high SNR, where the probability of error is determined by errors in selecting adjacent points.

5-28 Digital information is to be transmitted by carrier modulation through an additive gaussian noise channel with a bandwidth of 100 kHz and $N_0 = 10^{-10}$ W/Hz. Determine the maximum rate that can be transmitted through the channel for four-phase PSK, binary FSK, and four-frequency orthogonal FSK, which is detected noncoherently.

5-29 In a MSK signal, the initial state for the phase is either 0 or π rad. Determine the terminal phase state for the following four input pairs of input data: (a) 00; (b) 01; (c) 10; (d) 11.

5-30 A continuous-phase FSK signal with $h = \frac{1}{2}$ is represented as

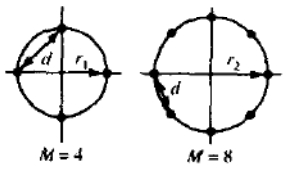
$$s(t) = \pm\sqrt{\frac{2\mathscr{E}_b}{T_b}}\cos\left(\frac{\pi t}{2T_b}\right)\cos 2\pi f_c t \pm \sqrt{\frac{2\mathscr{E}_b}{T_b}}\sin\left(\frac{\pi t}{2T_b}\right)\sin 2\pi f_c t, \quad 0 \le t \le 2T_b$$

where the ± signs depend on the information bits transmitted.
a Show that this signal has constant amplitude.
b Sketch a block diagram of the modulator for synthesizing the signal.
c Sketch a block diagram of the demodulator and detector for recovering the information.

5-31 Sketch the phase tree, the state trellis, and the state diagram for partial-response CPM with $h = \frac{1}{2}$ and

$$u(t) = \begin{cases} 1/4T & (0 \le t \le 2T) \\ 0 & (\text{otherwise}) \end{cases}$$

FIGURE P5-27



5-32 Determine the number of terminal phase states in the state trellis diagram for (a) a full response binary CPFSK with either $h = \frac{2}{3}$ or $\frac{3}{4}$ and (b) a partial-response $L = 3$ binary CPFSK with either $h = \frac{2}{3}$ or $\frac{3}{4}$.

5-33 Consider a biorthogonal signal set with $M = 8$ signal points. Determine a union bound for the probability of a symbol error as a function of $\mathscr{E}_b/N_0$. The signal points are equally likely a priori.

5-34 Consider an M-ary digital communication system where $M = 2^N$, and N is the dimension of the signal space. Suppose that the M signal vectors lie on the vertices of a hypercube that is centered at the origin. Determine the average probability of a symbol error as a function of $\mathscr{E}_s/N_0$ where $\mathscr{E}_s$ is the energy per symbol, $\frac{1}{2}N_0$ is the power spectral density of the AWGN, and all signal points are equally probable.

5-35 Consider the signal waveform

$$s(t) = \sum_{i=1}^{n} c_i p(t - kT_c)$$

where $p(t)$ is a rectangular pulse of unit amplitude and duration T_c. The $\{c_i\}$ may be viewed as a code vector $\mathbf{C} = [c_1 \ c_2 \ \dots \ c_n]$, where the elements $c_i = \pm 1$. Show that the filter matched to the waveform $s(t)$ may be realized as a cascade of a filter matched to $p(t)$ followed by a discrete-time filter matched to the vector $\mathbf{C}$. Determine the value of the output of the matched filter at the sampling instant $t = nT_c$.

5-36 A speech signal is sampled at a rate of 8 kHz, logarithmically compressed and encoded into a PCM format using 8 bits/sample. The PCM data is transmitted through an AWGN baseband channel via M-level PAM. Determine the bandwidth required for transmission when (a) $M = 4$, (b) $M = 8$, and (c) $M = 16$.

5-37 A Hadamard matrix is defined as a matrix whose elements are ± 1 and whose row vectors are pairwise orthogonal. In the case when n is a power of 2, an $n \times n$ Hadamard matrix is constructed by means of the recursion

$$\mathbf{H}_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, \quad \mathbf{H}_{2n} = \begin{bmatrix} \mathbf{H}_n & \mathbf{H}_n \\ \mathbf{H}_n & -\mathbf{H}_n \end{bmatrix}$$

a Let $\mathbf{C}_i$ denote the ith row of an $n \times n$ Hadamard matrix as defined above. Show that the waveforms constructed as

$$s_i(t) = \sum_{k=1}^{n} c_{ik} p(t - kT_c), \quad i = 1, 2, \dots, n$$

are orthogonal, where $p(t)$ is an arbitrary pulse confined to the time interval $0 \le t \le T_c$.

b Show that the matched filters (or cross-correlators) for the n waveforms $\{s_i(t)\}$ can be realized by a single filter (or correlator) matched to the pulse $p(t)$ followed by a set of n cross-correlators using the code words $\{\mathbf{C}_i\}$.

5-38 The discrete sequence

$$r_k = \sqrt{\mathscr{E}_b} c_k + n_k, \quad k = 1, 2, \dots, n$$

represents the output sequence of samples from a demodulator, where $c_k = \pm 1$ are elements of one of two possible code words, $\mathbf{C}_1 = [1 \ 1 \ \dots \ 1]$ and $\mathbf{C}_2 = [1 \ 1 \ \dots \ 1 \ -1 \ \dots \ -1]$. The code word $\mathbf{C}_2$ has w elements that are $+1$ and $n - w$

elements that are -1, where w is some positive integer. The noise sequence $\{n_k\}$ is white gaussian with variance σ^2.

a What is the optimum maximum likelihood detector for the two possible transmitted signals?

b Determine the probability of error as a function of the parameters $(\sigma^2, \mathscr{E}_b, w)$.

c What is the value of w that minimizes the error probability?

5-39 Derive the outputs r_1 and r_2 of the two correlators shown in Fig. 5-4-1. Assume that a signal $s_{l1}(t)$ is transmitted and that

$$r_l(t) = s_{l1}(t)e^{j\phi} + z(t)$$

where $z(t) = n_c(t) + jn_s(t)$ is the additive gaussian noise.

5-40 Determine the covariances and variances of the gaussian random noise variables n_{1c}, n_{2c}, n_{1s}, and n_{2s} in (5-4-15) and the joint pdf.

5-41 Derive the matched filter outputs given by (5-4-10).

5-42 In on–off keying of a carrier-modulated signal, the two possible signals are

$$s_0(t) = 0, \quad 0 \le t \le T_b$$

$$s_1(t) = \sqrt{\frac{2\mathscr{E}_b}{T_b}} \cos 2\pi f_c t, \quad 0 \le t \le T_b$$

The corresponding received signals are

$$r(t) = n(t), \quad 0 \le t \le T_b$$

$$r(t) = \sqrt{\frac{2\mathscr{E}_b}{T_b}} \cos (2\pi f_c t + \phi) + n(t), \quad 0 \le t \le T_b$$

where ϕ is the carrier phase and $n(t)$ is AWGN.

a Sketch a block diagram of the receiver (demodulator and detector) that employs noncoherent (envelope) detection.

b Determine the pdfs for the two possible decision variables at the detector corresponding to the two possible received signals.

c Derive the probability of error for the detector.

5-43 In two-phase DPSK, the received signal in one signaling interval is used as a phase reference for the received signal in the following signaling interval. The decision variable is

$$D = \text{Re}\,(V_m V_{m-1}^*) \underset{0}{\overset{1}{\gtrless}} 0$$

$$V_k = 2\alpha\mathscr{E}e^{j(\theta_k - \phi)} + N_k$$

represents the complex-valued output of the filter matched to the transmitted signal $u(t)$. N_k is a complex-valued gaussian variable having zero mean and statistically independent components.

a Writing $V_k = X_k + jY_k$, show that D is equivalent to

$$d = [\tfrac{1}{2}(X_m + X_{m-1})]^2 + [\tfrac{1}{2}(Y_m + Y_{m-1})]^2 - [\tfrac{1}{2}(X_m - X_{m-1})]^2 - [\tfrac{1}{2}(Y_m - Y_{m-1})]^2$$

b For mathematical convenience; suppose that $\theta_k = \theta_{k-1}$. Show that the random variables U_1, U_2, U_3, and U_4 are statistically independent gaussian variables, where $U_1 = \frac{1}{2}(X_m + X_{m-1})$, $U_2 = \frac{1}{2}(Y_m + Y_{m-1})$, $U_3 = \frac{1}{2}(X_m - X_{m-1})$, and $U_4 = \frac{1}{2}(Y_m - Y_{m-1})$.

c Define the random variables $W_1 = U_1^2 + U_2^2$ and $W_2 = U_3^2 + U_4^2$. Then

$$D = W_1 - W_2 \underset{H_0}{\overset{H_1}{\gtrless}} 0$$

Determine the probability density functions for W_1 and W_2.

d Determine the probability of error P_b, where

$$P_b = P(D < 0) = P(W_1 - W_2 < 0) = \int_0^\infty P(W_2 > w_1 \mid w_1) p(w_1)\, dw_1$$

5-44 Recall that MSK can be represented as a four-phase offset PSK modulation having the lowpass equivalent form

$$v(t) = \sum_k [I_k u(t - 2kT_b) + jJ_k u(t - 2kT_b - T_b)]$$

where

$$u(t) = \begin{cases} \sin(\pi t/2T_b) & (0 \leq t \leq 2T_b) \\ 0 & (\text{otherwise}) \end{cases}$$

and $\{I_k\}$ and $\{J_k\}$ are sequences of information symbols (± 1).

a Sketch the block diagram of an MSK demodulator for offset QPSK.

b Evaluate the performance of the four-phase demodulator for AWGN if no account is taken of the memory in the modulation.

c Compare the performance obtained in (b) with that for Viterbi decoding of the MSK signal.

d The MSK signal is also equivalent to binary FSK. Determine the performance of noncoherent detection of the MSK signal. Compare your result with (b) and (c).

5-45 Consider a transmission line channel that employs $n - 1$ regenerative repeaters plus the terminal receiver in the transmission of binary information. Assume that the probability or error at the detector of each receiver is p and that errors among repeaters are statistically independent.

a Show that the binary error probability at the terminal receiver is

$$P_n = \tfrac{1}{2}[1 - (1 - 2p)^n]$$

b If $p = 10^{-6}$ and $n = 100$, determine an approximate value of P_n.

5-46 A digital communication system consists of a transmission line with 100 digital (regenerative) repeaters. Binary antipodal signals are used for transmitting the information. If the overall end-to-end error probability is 10^{-6}, determine the probability of error for each repeater and the required $\mathscr{E}_b/N_0$ to achieve this performance in AWGN.

5-47 A radio transmitter has a power output of $P_T = 1$ W at a frequency of 1 GHz. The transmitting and receiving antennas are parabolic dishes with diameter $D = 3$ m.

a Determine the antenna gains.

b Determine the EIRP for the transmitter.

c The distance (free space) between the transmitting and receiving antennas is 20 km. Determine the signal power at the output of the receiving antenna in dBm.

5-48 A radio communication system transmits at a power level of 0.1 W at 1 GHz. The transmitting and receiving antennas are parabolic, each having a diameter of 1 m. The receiver is located 30 km from the transmitter.
- **a** Determine the gains of the transmitting and receiving antennas.
- **b** Determine the EIRP of the transmitted signal.
- **c** Determine the signal power from the receiving antenna.

5-49 A satellite in synchronous orbit is used to communicate with an earth station at a distance of 40 000 km. The satellite has an antenna with a gain of 15 dB and a transmitter power of 3 W. The earth station uses a 10 m parabolic antenna with an efficiency of 0.6. The frequency band is at $f = 10$ GHz. Determine the received power level at the output of the receiver antenna.

5-50 A spacecraft located 100 000 km from the earth is sending data at a rate of R bits/s. The frequency band is centered at 2 GHz and the transmitted power is 10 W. The earth station uses a parabolic antenna, 50 m in diameter, and the spacecraft has an antenna with a gain of 10 dB. The noise temperature of the receiver front end is $T_0 = 300$ K.
- **a** Determine the received power level.
- **b** If the desired $\mathscr{E}_b/N_0 = 10$ dB, determine the maximum bit rate that the spacecraft can transmit.

5-51 A satellite in geosynchronous orbit is used as a regenerative repeater in a digital communication system. Consider the satellite-to-earth link in which the satellite antenna has a gain of 6 dB and the earth station antenna has a gain of 50 dB. The downlink is operated at a center frequency of 4 GHz, and the signal bandwidth is 1 MHz. If the required $\mathscr{E}_b/N_0$ for reliable communication is 15 dB, determine the transmitted power for the satellite downlink. Assume that $N_0 = 4.1 \times 10^{-21}$ W/Hz.

6

CARRIER AND SYMBOL SYNCHRONIZATION

We have observed that in a digital communication system, the output of the demodulator must be sampled periodically, once per symbol interval, in order to recover the transmitted information. Since the propagation delay from the transmitter to the receiver is generally unknown at the receiver, symbol timing must be derived from the received signal in order to synchronously sample the output of the demodulator.

The propagation delay in the transmitted signal also results in a carrier offset, which must be estimated at the receiver if the detector is phase-coherent. In this chapter, we consider methods for deriving carrier and symbol synchronization at the receiver.

6-1 SIGNAL PARAMETER ESTIMATION

Let us begin by developing a mathematical model for the signal at the input to the receiver. We assume that the channel delays the signals transmitted through it and corrupts them by the addition of gaussian noise. Hence, the received signal may be expressed as

$$r(t) = s(t - \tau) + n(t)$$

where

$$s(t) = \text{Re}\,[s_l(t)e^{j2\pi f_c t}] \tag{6-1-1}$$

and where τ is the propagation delay and $s_l(t)$ is the equivalent lowpass signal.

The received signal may be expressed as

$$r(t) = \text{Re}\,\{[s_l(t - \tau)e^{j\phi} + z(t)]e^{j2\pi f_c t}\} \tag{6-1-2}$$

where the carrier phase ϕ, due to the propagation delay τ, is $\phi = -2\pi f_c \tau$.

Now, from this formulation, it may appear that there is only one signal parameter to be estimated, namely, the propagation delay, since one can determine ϕ from knowledge of f_c and τ. However, this is not the case. First of all, the oscillator that generates the carrier signal for demodulation at the receiver is generally not synchronous in phase with that at the transmitter. Furthermore, the two oscillators may be drifting slowly with time, perhaps in different directions. Consequently, the received carrier phase is not only dependent on the time delay τ. Furthermore, the precision to which one must synchronize in time for purpose of demodulating the received signal depends on the symbol interval T. Usually, the estimation error in estimating τ must be a relatively small fraction of T. For example, $\pm 1\%$ of T is adequate for practical applications. However, this level of precision is generally inadequate for estimating the carrier phase, even if ϕ depends only on τ. This is due to the fact that f_c is generally large, and, hence, a small estimation error in τ causes a large phase error.

In effect, we must estimate both parameters τ and ϕ in order to demodulate and coherently detect the received signal. Hence, we may express the received signal as

$$r(t) = s(t; \phi, \tau) + n(t) \tag{6-1-3}$$

where ϕ and τ represent the signal parameters to be estimated. To simplify the notation, we let ψ denote the parameter vector $\{\phi, \tau\}$, so that $s(t; \phi, \tau)$ is simply denoted by $s(t; \psi)$.

There are basically two criteria that are widely applied to signal parameter estimation: the *maximum-likelihood* (ML) criterion and the *maximum* a posteriori *probability* (MAP) criterion. In the MAP criterion, the signal parameter vector ψ is modeled as random, and characterized by an a priori probability density function $p(\psi)$. In the maximum-likelihood criterion, the signal parameter vector ψ is treated as deterministic but unknown.

By performing an orthonormal expansion of $r(t)$ using N orthonormal functions $\{f_n(t)\}$, we may represent $r(t)$ by the vector of coefficients $[r_1\ r_2\ \ldots\ r_N] \equiv \mathbf{r}$. The joint pdf of the random variables $[r_1\ r_2\ \ldots\ r_N]$ in the expansion can be expressed as $p(\mathbf{r} \mid \psi)$. Then, the ML estimate of ψ is the value that maximizes $p(\mathbf{r} \mid \psi)$. On the other hand, the MAP estimate is the value of ψ that maximizes the a posteriori probability density function

$$p(\psi \mid \mathbf{r}) = \frac{p(\mathbf{r} \mid \psi) p(\psi)}{p(\mathbf{r})} \tag{6-1-4}$$

We note that if there is no prior knowledge of the parameter vector ψ, we may assume that $p(\psi)$ is uniform (constant) over the range of values of the parameters. In such a case, the value of ψ that maximizes $p(\mathbf{r} \mid \psi)$ also maximizes $p(\psi \mid \mathbf{r})$. Therefore, the MAP and ML estimates are identical.

In our treatment of parameter estimation given below, we view the parameters ϕ and τ as unknown, but deterministic. Hence, we adopt the ML criterion for estimating them.

In the ML estimation of signal parameters, we require that the receiver extract the estimate by observing the received signal over a time interval $T_0 \geq T$, which is called the observation interval. Estimates obtained from a single observation interval are sometimes called one-shot estimates. In practice, however, the estimation is performed on a continuous basis by using tracking loops (either analog or digital) that continuously update the estimates. Nevertheless, one-shot estimates yield insight for tracking loop implementation. In addition, they prove useful in the analysis of the performance of ML estimation, and their performance can be related to that obtained with a tracking loop.

6-1-1 The Likelihood Function

Although it is possible to derive the parameter estimates based on the joint pdf of the random variables $[r_1 \ r_2 \ \ldots \ r_N]$ obtained from the expansion of $r(t)$, it is convenient to deal directly with the signal waveforms when estimating their parameters. Hence, we shall develop a continuous-time equivalent of the maximization of $p(\mathbf{r} \mid \boldsymbol{\psi})$.

Since the additive noise $n(t)$ is white and zero-mean gaussian, the joint pdf $p(\mathbf{r} \mid \boldsymbol{\psi})$ may be expressed as

$$p(\mathbf{r} \mid \boldsymbol{\psi}) = \left(\frac{1}{\sqrt{2\pi}\sigma}\right)^N \exp\left\{-\sum_{n=1}^{N} \frac{[r_n - s_n(\boldsymbol{\psi})]^2}{2\sigma^2}\right\} \tag{6-1-5}$$

where

$$\begin{aligned} r_n &= \int_{T_0} r(t) f_n(t)\, dt \\ s_n(\boldsymbol{\psi}) &= \int_{T_0} s(t; \boldsymbol{\psi}) f_n(t)\, dt \end{aligned} \tag{6-1-6}$$

where T_0 represents the integration interval in the expansion of $r(t)$ and $s(t; \boldsymbol{\psi})$.

We note that the argument in the exponent may be expressed in terms of the signal waveforms $r(t)$ and $s(t; \boldsymbol{\psi})$, by substituting from (6-1-6) into (6-1-5). That is,

$$\frac{1}{2\sigma^2} \sum_{n=1}^{N} [r_n - s_n(t; \boldsymbol{\psi})]^2 = \frac{1}{N_0} \int_{T_0} [r(t) - s(t; \boldsymbol{\psi})]^2\, dt \tag{6-1-7}$$

where the proof is left as an exercise for the reader (see Problem 6-1). Now, the maximization of $p(\mathbf{r} \mid \boldsymbol{\psi})$ with respect to the signal parameters $\boldsymbol{\psi}$ is equivalent to the maximization of the *likelihood function*.

$$\Lambda(\boldsymbol{\psi}) = \exp\left\{-\frac{1}{N_0} \int_{T_0} [r(t) - s(t; \boldsymbol{\psi})]^2\, dt\right\} \tag{6-1-8}$$

Below, we shall consider signal parameter estimation from the viewpoint of maximizing $\Lambda(\boldsymbol{\psi})$.

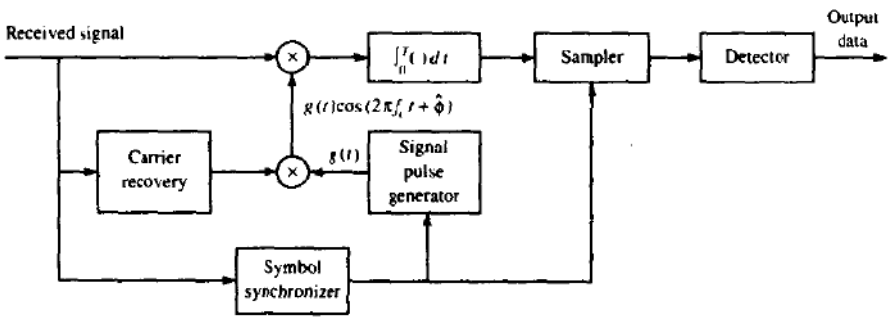


FIGURE 6-1-1 Block diagram of binary PSK receiver.

6-1-2 Carrier Recovery and Symbol Synchronization in Signal Demodulation

Symbol synchronization is required in every digital communication system which transmits information synchronously. Carrier recovery is required if the signal is detected coherently.

Figure 6-1-1 illustrates the block diagram of a binary PSK (or binary PAM) signal demodulator and detector. As shown, the carrier phase estimate $\hat{\phi}$ is used in generating the reference signal $g(t)\cos(2\pi f_c t + \hat{\phi})$ for the correlator. The symbol synchronizer controls the sampler and the output of the signal pulse generator. If the signal pulse is rectangular then the signal generator can be eliminated.

The block diagram of an M-ary PSK demodulator is shown in Fig. 6-1-2. In this case, two correlators (or matched filters) are required to correlate the received signal with the two quadrature carrier signals $g(t)\cos(2\pi f_c t + \hat{\phi})$ and $g(t)\sin(2\pi f_c t + \hat{\phi})$, where $\hat{\phi}$ is the carrier phase estimate. The detector is now a phase detector, which compares the received signal phases with the possible transmitted signal phases.

The block diagram of a PAM signal demodulator is shown in Fig. 6-1-3. In this case, a single correlator is required, and the detector is an amplitude detector, which compares the received signal amplitude with the possible transmitted signal amplitudes. Note that we have included an automatic gain control (AGC) at the front-end of the demodulator to eliminate channel gain variations, which would affect the amplitude detector. The AGC has a relatively long time constant, so that it does not respond to the signal amplitude variations that occur on a symbol-by-symbol basis. Instead, the AGC maintains a fixed average (signal plus noise) power at its output.

Finally, we illustrate the block diagram of a QAM demodulator in Fig. 6-1-4. As in the case of PAM, an AGC is required to maintain a constant average power signal at the input to the demodulator. We observe that the demodulator is similar to a PSK demodulator, in that both generate in-phase and quadrature signal samples (X, Y) for the detector. In the case of QAM,

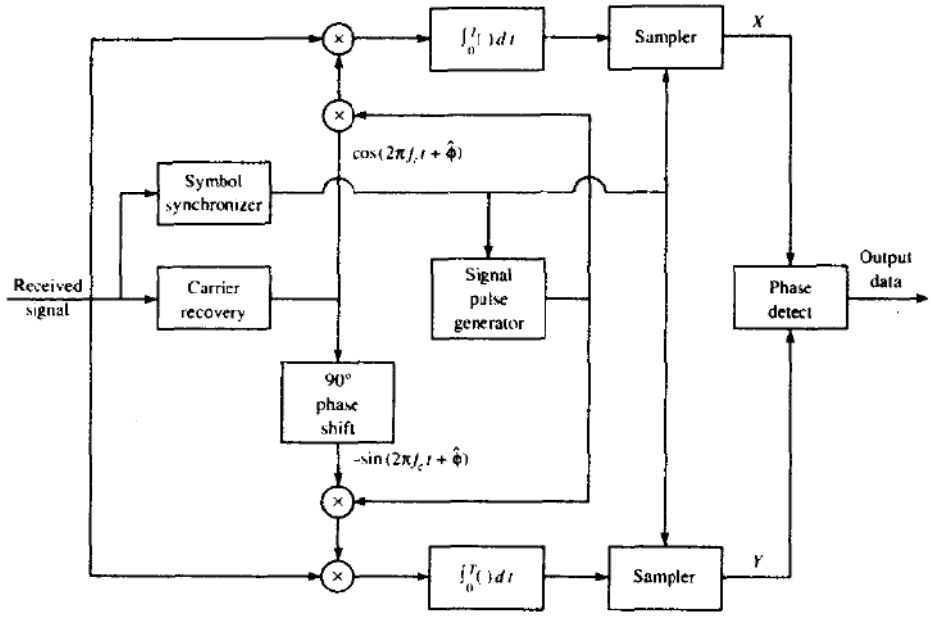


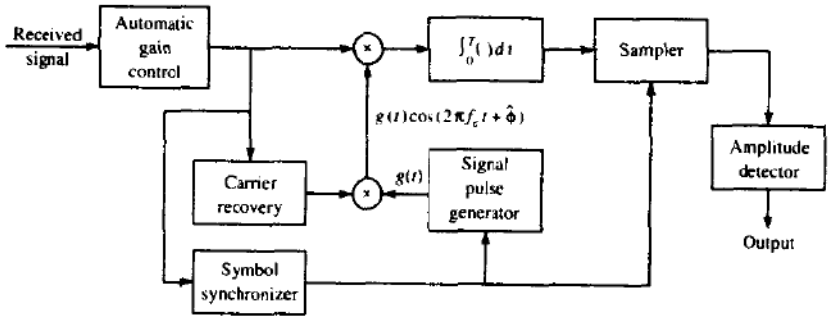
FIGURE 6-1-2 Block diagram of M-ary PSK receiver.

the detector computes the euclidean distance between the received noise-corrupted signal point and the M possible transmitted points, and selects the signal closest to the received point.

6-2 CARRIER PHASE ESTIMATION

There are two basic approaches for dealing with carrier synchronization at the receiver. One is to multiplex, usually in frequency, a special signal, called a pilot signal, that allows the receiver to extract and, thus, to synchronize its local oscillator to the carrier frequency and phase of the received signal. When

FIGURE 6-1-3 Block diagram of M-ary PAM receiver.



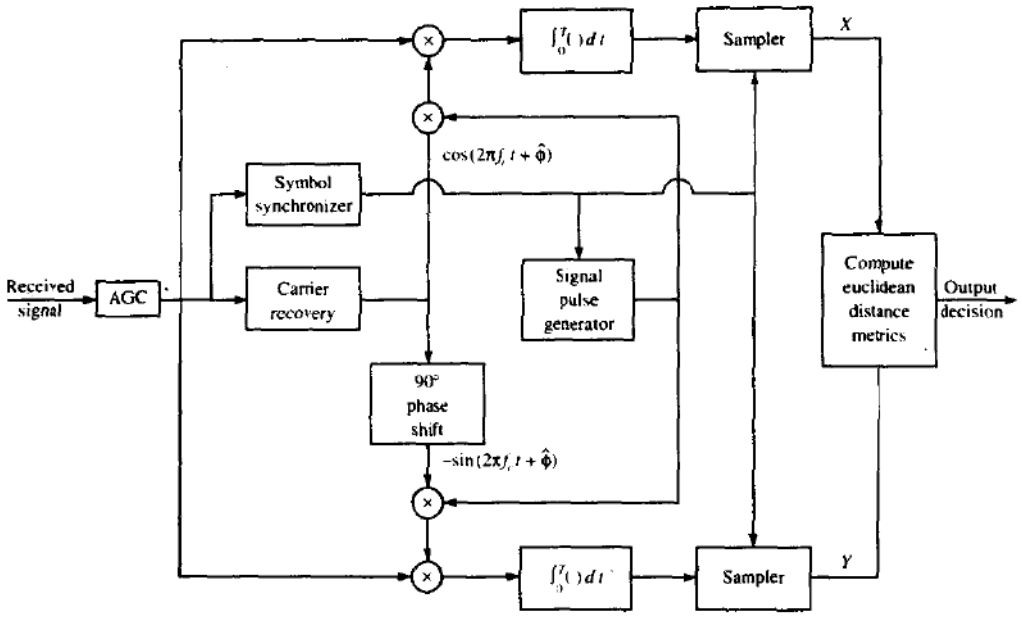


FIGURE 6-1-4 Block diagram of QAM receiver.

an unmodulated carrier component is transmitted along with the information-bearing signal, the receiver employs a phase-locked loop (PLL) to acquire and track the carrier component. The PLL is designed to have a narrow bandwidth so that it is not significantly affected by the presence of frequency components from the information-bearing signal.

The second approach, which appears to be more prevalent in practice, is to derive the carrier phase estimate directly from the modulated signal. This approach has the distinct advantage that the total transmitter power is allocated to the transmission of the information-bearing signal. In our treatment of carrier recovery, we confine our attention to the second approach; hence, we assume that the signal is transmitted via suppressed carrier.

In order to emphasize the importance of extracting an accurate phase estimate, let us consider the effect of a carrier phase error on the demodulation of a double-sideband, suppressed carrier (DSB/SC) signal. To be specific, suppose we have an amplitude-modulated signal of the form

$$s(t) = A(t)\cos(2\pi f_c t + \phi) \tag{6-2-1}$$

If we demodulate the signal by multiplying $s(t)$ with the carrier reference

$$c(t) = \cos(2\pi f_c t + \hat{\phi}) \tag{6-2-2}$$

we obtain

$$c(t)s(t) = \tfrac{1}{2}A(t)\cos(\phi - \hat{\phi}) + \tfrac{1}{2}A(t)\cos(4\pi f_c t + \phi + \hat{\phi})$$

The double-frequency component may be removed by passing the product signal $c(t)s(t)$ through a lowpass filter. This filtering yields the information-bearing signal

$$y(t) = \tfrac{1}{2}A(t)\cos(\phi - \hat{\phi}) \tag{6-2-3}$$

Note that the effect of the phase error $\phi - \hat{\phi}$ is to reduce the signal level in voltage by a factor $\cos(\phi - \hat{\phi})$ and in power by a factor $\cos^2(\phi - \hat{\phi})$. Hence, a phase error of 10° results in a signal power loss of 0.13 dB, and a phase error of 30° results in a signal power loss of 1.25 dB in an amplitude-modulated signal.

The effect of carrier phase errors in QAM and multiphase PSK is much more severe. The QAM and M-PSK signals may be represented as

$$s(t) = A(t)\cos(2\pi f_c t + \phi) - B(t)\sin(2\pi f_c t + \phi) \tag{6-2-4}$$

This signal is demodulated by the two quadrature carriers

$$\begin{aligned} c_c(t) &= \cos(2\pi f_c t + \hat{\phi}) \\ c_s(t) &= -\sin(2\pi f_c t + \hat{\phi}) \end{aligned} \tag{6-2-5}$$

Multiplication of $s(t)$ with $c_c(t)$ followed by lowpass filtering yields the in-phase component

$$y_I(t) = \tfrac{1}{2}A(t)\cos(\phi - \hat{\phi}) - \tfrac{1}{2}B(t)\sin(\phi - \hat{\phi}) \tag{6-2-6}$$

Similarly, multiplication of $s(t)$ by $c_s(t)$ followed by lowpass filtering yields the quadrature component

$$y_Q(t) = \tfrac{1}{2}B(t)\cos(\phi - \hat{\phi}) + \tfrac{1}{2}A(t)\sin(\phi - \hat{\phi}) \tag{6-2-7}$$

The expressions (6-2-6) and (6-2-7) clearly indicate that the phase error in the demodulation of QAM and M-PSK signals has a much more severe effect than in the demodulation of a PAM signal. Not only is there a reduction in the power of the desired signal component by a factor $\cos^2(\phi - \hat{\phi})$, but there is also crosstalk interference from the in-phase and quadrature components. Since the average power levels of $A(t)$ and $B(t)$ are similar, a small phase error causes a large degradation in performance. Hence, the phase accuracy requirements for QAM and multiphase coherent PSK are much higher than DSB/SC PAM.

6-2-1 Maximum-Likelihood Carrier Phase Estimation

First, we derive the maximum-likelihood carrier phase estimate. For simplicity, we assume that the delay τ is known and, in particular, we set $\tau = 0$. The function to be maximized is the likelihood function given in (6-1-8). With ϕ substituted for ψ, this function becomes

$$\begin{aligned} \Lambda(\phi) &= \exp\left\{-\frac{1}{N_0}\int_{T_0}[r(t) - s(t;\phi)]^2\,dt\right\} \\ &= \exp\left\{-\frac{1}{N_0}\int_{T_0} r^2(t)\,dt + \frac{2}{N_0}\int_{T_0} r(t)s(t;\phi)\,dt - \frac{1}{N_0}\int_{T_0} s^2(t;\phi)\,dt\right\}. \end{aligned} \tag{6-2-8}$$

Note that the first term of the exponential factor does not involve the signal parameter ϕ. The third term, which contains the integral of $s^2(t;\phi)$, is a constant equal to the signal energy over the observation interval T_0 for any value of ϕ. Only the second term, which involves the cross-correlation of the received signal $r(t)$ with the signal $s(t;\phi)$, depends on the choice of ϕ. Therefore, the likelihood function $\Lambda(\phi)$ may be expressed as

$$\Lambda(\phi) = C \exp\left[\frac{2}{N_0}\int_{T_0} r(t)s(t;\phi)\,dt\right] \tag{6-2-9}$$

where C is a constant independent of ϕ.

The ML estimate $\hat{\phi}_{\text{ML}}$ is the value of ϕ that maximizes $\Lambda(\phi)$ in (6-2-9). Equivalently, the value $\hat{\phi}_{\text{ML}}$ also maximizes the logarithm of $\Lambda(\phi)$, i.e., the log-likelihood function

$$\Lambda_L(\phi) = \frac{2}{N_0}\int_{T_0} r(t)s(t;\phi)\,dt \tag{6-2-10}$$

Note that in defining $\Lambda_L(\phi)$ we have ignored the constant term $\ln C$.

Example 6-2-1

As an example of the optimization to determine the carrier phase, let us consider the transmission of the unmodulated carrier $A\cos 2\pi f_c t$. The received signal is

$$r(t) = A\cos(2\pi f_c t + \phi) + n(t)$$

where ϕ is the unknown phase. We seek the value ϕ, say $\hat{\phi}_{\text{ML}}$, that maximizes

$$\Lambda_L(\phi) = \frac{2A}{N_0}\int_{T_0} r(t)\cos(2\pi f_c t + \phi)\,dt$$

A necessary condition for a maximum is that

$$\frac{d\Lambda_L(\phi)}{d\phi} = 0$$

This condition yields

$$\int_{T_0} r(t)\sin(2\pi f_c t + \hat{\phi}_{\text{ML}})\,dt = 0 \tag{6-2-11}$$

or, equivalently,

$$\hat{\phi}_{\text{ML}} = -\tan^{-1}\left[\int_{T_0} r(t)\sin 2\pi f_c t\,dt \Big/ \int_{T_0} r(t)\cos 2\pi f_c t\,dt\right] \tag{6-2-12}$$

We observe that the optimality condition given by (6-2-11) implies the use

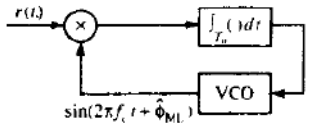


FIGURE 6-2-1 A PLL for obtaining the ML estimate of the phase of an unmodulated carrier.

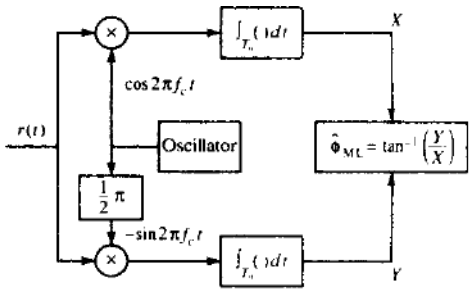


FIGURE 6-2-2 A (one-shot) ML estimate of the phase of an unmodulated carrier.

of a loop to extract the estimate as illustrated in Fig. 6-2-1. The loop filter is an integrator whose bandwidth is proportional to the reciprocal of the integration interval T_0. On the other hand, (6-2-12) implies an implementation that uses quadrature carriers to cross-correlate with $r(t)$. Then, $\hat{\phi}_{ML}$ is the inverse tangent of the ratio of these two correlator outputs, as shown in Fig. 6-2-2. Note that this estimation scheme yields $\hat{\phi}_{ML}$ explicitly.

This example clearly demonstrates that the PLL provides the ML estimate of the phase of an unmodulated carrier.

6-2-2 The Phase-Locked Loop

The PLL basically consists of a multiplier, a loop filter, and a voltage-controlled oscillator (VCO), as shown in Fig. 6-2-3. If we assume that the input to the PLL is the sinusoid $\cos(2\pi f_c t + \phi)$ and the output of the VCO is $\sin(2\pi f_c t + \hat{\phi})$, where $\hat{\phi}$ represents the estimate of ϕ, the product of these two signals is

$$\begin{aligned} e(t) &= \cos(2\pi f_c t + \phi)\sin(2\pi f_c t + \hat{\phi}) \\ &= \tfrac{1}{2}\sin(\hat{\phi} - \phi) + \tfrac{1}{2}\sin(4\pi f_c t + \phi + \hat{\phi}) \end{aligned} \tag{6-2-13}$$

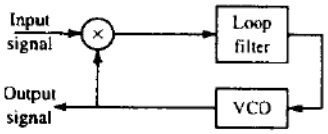


FIGURE 6-2-3 Basic elements of a phase-located loop (PLL).

The loop filter is a lowpass filter that responds only to the low-frequency component $\frac{1}{2}\sin(\hat{\phi}-\phi)$ and removes the component at $2f_c$. This filter is usually selected to have the relatively simple transfer function

$$G(s)=\frac{1+\tau_2 s}{1+\tau_1 s} \tag{6-2-14}$$

where τ_1 and τ_2 are design parameters ($\tau_1 \gg \tau_2$) that control the bandwidth of the loop. A higher-order filter that contains additional poles may be used if necessary to obtain a better loop response.

The output of the loop filter provides the control voltage $v(t)$ for the VCO. The VCO is basically a sinusoidal signal generator with an instantaneous phase given by

$$2\pi f_c t+\hat{\phi}(t)=2\pi f_c t+K\int_{-\infty}^{t} v(\tau)\,d\tau \tag{6-2-15}$$

where K is a gain constant in rad/V. Hence,

$$\hat{\phi}(t)=K\int_{-\infty}^{t} v(\tau)\,d\tau \tag{6-2-16}$$

By neglecting the double-frequency term resulting from the multiplication of the input signal with the output of the VCO, we may reduce the PLL into the equivalent closed-loop system model shown in Fig. 6-2-4. The sine function of the phase difference $\phi-\hat{\phi}$ makes this system nonlinear, and, as a consequence, the analysis of its performance in the presence of noise is somewhat involved but, nevertheless, it is mathematically tractable for some simple loop filters.

In normal operation when the loop is tracking the phase of the incoming carrier, the phase error $\phi-\hat{\phi}$ is small and, hence,

$$\sin(\hat{\phi}-\phi)\approx\hat{\phi}-\phi \tag{6-2-17}$$

With this approximation, the PLL becomes linear and is characterized by the closed-loop transfer function

$$H(s)=\frac{KG(s)/s}{1+KG(s)/s} \tag{6-2-18}$$

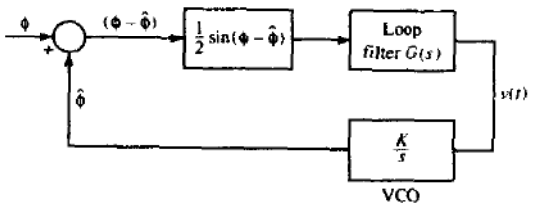


FIGURE 6-2-4 Model of phase-locked loop.

where the factor of $\frac{1}{2}$ has been absorbed into the gain parameter K. By substituting from (6-2-14) for $G(s)$ into (6-2-18), we obtain

$$H(s) = \frac{1 + \tau_2 s}{1 + (\tau_2 + 1/K)s + (\tau_1/K)s^2} \tag{6-2-19}$$

Hence, the closed-loop system for the linearized PLL is second-order when $G(s)$ is given by (6-2-14). The parameter τ_2 controls the position of the zero, while K and τ_1 are used to control the position of the closed-loop system poles. It is customary to express the denominator of $H(S)$ in the standard form

$$D(s) = s^2 + 2\zeta\omega_n s + \omega_n^2 \tag{6-2-20}$$

where ζ is called the *loop damping factor* and ω_n is the natural frequency of the loop. In terms of the loop parameters, $\omega_n = \sqrt{K/\tau_1}$, and $\zeta = (\tau_2 + 1/K)/2\omega_n$, the closed-loop transfer function becomes

$$H(s) = \frac{(2\zeta\omega_n - \omega_n^2/K)s + \omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{6-2-21}$$

The (one-sided) noise-equivalent bandwidth (see Problem 2-24) of the loop is

$$\begin{aligned} B_{eq} &= \frac{\tau_2^2(1/\tau_2^2 + K/\tau_1)}{4(\tau_2 + 1/K)} \\ &= \frac{1 + (\tau_2\omega_n)^2}{8\zeta\omega_n} \end{aligned} \tag{6-2-22}$$

The magnitude response $20 \log |H(\omega)|$ as a function of the normalized frequency ω/ω_n is illustrated in Fig. 6-2-5, with the damping factor ζ as a parameter and $\tau_1 \gg 1$. Note that $\zeta = 1$ results in a critically damped loop response, $\zeta < 1$ produces an underdamped response, and $\zeta > 1$ yields an overdamped response.

In practice, the selection of the bandwidth of the PLL involves a trade-off between speed of response and noise in the phase estimate, which is the topic considered below. On the one hand, it is desirable to select the bandwidth of the loop to be sufficiently wide to track any time variations in the phase of the received carrier. On the other, a wideband PLL allows more noise to pass into the loop, which corrupts the phase estimate. Below, we assess the effects of noise in the quality of the phase estimate.

6-2-3 Effect of Additive Noise on the Phase Estimate

In order to evaluate the effects of noise on the estimate of the carrier phase, let us assume that the noise at the input to the PLL is narrowband. For this analysis, we assume that the PLL is tracking a sinusoidal signal of the form

$$s(t) = A_c \cos [2\pi f_c t + \phi(t)] \tag{6-2-23}$$

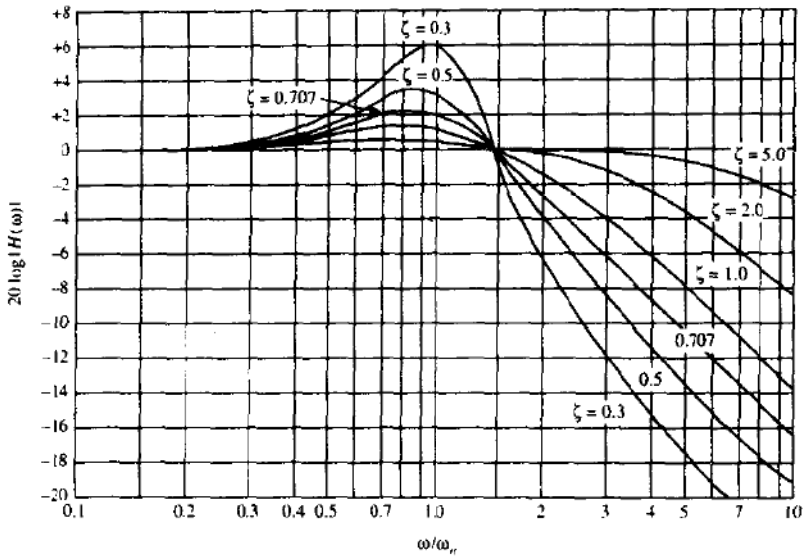


FIGURE 6-2-5 Frequency response of a second-order loop. [*From* Phaselock Techniques, *2nd edition, by F. M. Gardner, © 1979 by John Wiley and Sons, Inc. Reprinted with permission of the publisher.*]

that is corrupted by the additive narrowband noise

$$n(t) = x(t)\cos 2\pi f_c t - y(t)\sin 2\pi f_c t \tag{6-2-24}$$

The in-phase and quadrature components of the noise are assumed to be statistically independent, stationary gaussian noise processes with (two-sided) power spectral density $\frac{1}{2}N_0$ W/Hz. By using simple trigonometric identities, the noise term in (6-2-24) can be expressed as

$$n(t) = n_c(t)\cos[2\pi f_c t + \phi(t)] - n_s(t)\sin[2\pi f_c t + \phi(t)] \tag{6-2-25}$$

where

$$\begin{aligned} n_c(t) &= x(t)\cos\phi(t) + y(t)\sin\phi(t) \\ n_s(t) &= -x(t)\sin\phi(t) + y(t)\cos\phi(t) \end{aligned} \tag{6-2-26}$$

We note that

$$n_c(t) + jn_s(t) = [x(t) + jy(t)]e^{-j\phi(t)}$$

so that the quadrature components $n_c(t)$ and $n_s(t)$ have exactly the same statistical characteristics as $x(t)$ and $y(t)$.

If $s(t) + n(t)$ is multiplied by the output of the VCO and the double-frequency terms are neglected, the input to the loop filter is the noise-corrupted signal

$$\begin{aligned} e(t) &= A_c\sin\Delta\phi + n_c(t)\sin\Delta\phi - n_s(t)\cos\Delta\phi \\ &= A_c\sin\Delta\phi + n_1(t) \end{aligned} \tag{6-2-27}$$

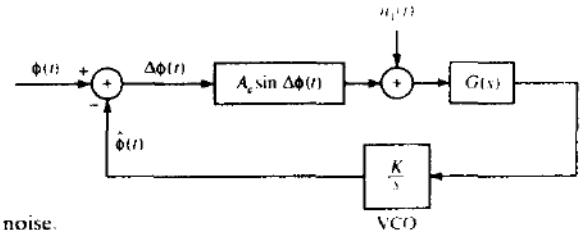


FIGURE 6-2-6 Equivalent PLL model with additive noise.

where, by definition, $\Delta\phi = \phi - \hat{\phi}$ is the phase error. Thus, we have the equivalent model for the PLL with additive noise as shown in Fig. 6-2-6.

When the power $P_c = \frac{1}{2}A_c^2$ of the incoming signal is much larger than the noise power, we may linearize the PLL and, thus, easily determine the effect of the additive noise on the quality of the estimate $\hat{\phi}$. Under these conditions, the model for the linearized PLL with additive noise is illustrated in Fig. 6-2-7. Note that the gain parameter A_c may be normalized to unity, provided that the noise terms are scaled by $1/A_c$, i.e., the noise terms become

$$n_2(t) = \frac{n_c(t)}{A_c} \sin \Delta\phi - \frac{n_s(t)}{A_c} \cos \Delta\phi \tag{6-2-28}$$

Since the noise $n_2(t)$ is additive at the input to the loop, the variance of the phase error $\Delta\phi$, which is also the variance of the VCO output phase, is

$$\sigma_{\hat{\phi}}^2 = \frac{N_0 B_{\text{eq}}}{A_c^2} \tag{6-2-29}$$

where B_{eq} is the (one-sided) equivalent noise bandwidth of the loop, given in (6-2-22). Note that $\sigma_{\hat{\phi}}^2$ is simply the ratio of total noise power within the bandwidth of the PLL divided by the signal power A^2. Hence,

$$\sigma_{\hat{\phi}}^2 = 1/\gamma_L \tag{6-2-30}$$

where γ_L is defined as the signal-to-noise ratio

$$\text{SNR} \equiv \gamma_L = \frac{A_c^2}{N_0 B_{\text{eq}}} \tag{6-2-31}$$

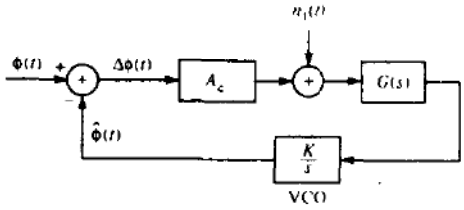


FIGURE 6-2-7 Linearized PLL model with additive noise.

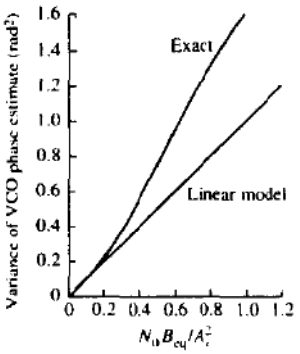


FIGURE 6-2-8 Comparison of VCO phase variance for exact and approximate (linear model) first-order PLL. [*From* Principles of Coherent Communication, *by A. J. Viterbi; © 1966 by McGraw-Hill Book Company. Reprinted with permission of the publisher.*]

The expression for the variance σ_ϕ^2 of the VCO phase error applies to the case where the SNR is sufficiently high that the linear model for the PLL applies. An exact analysis based on the nonlinear PLL is mathematically tractable when $G(s) = 1$, which results in a first-order loop. In this case, the probability density function for the phase error may be derived (see Viterbi, 1966) and has the form

$$p(\Delta\phi) = \frac{\exp(\gamma_L \cos \Delta\phi)}{2\pi I_0(\gamma_L)} \tag{6-2-32}$$

where γ_L is the SNR given by (6-2-31) with B_{eq} being the appropriate noise bandwidth of the first-order loop, and $I_0(\cdot)$ is the modified Bessel function of order zero.

From the expression for $p(\Delta\phi)$, we may obtain the exact value of the variance for the phase error on a first-order PLL. This is plotted in Fig. 6-2-8 as a function of $1/\gamma_L$. Also shown for comparison is the result obtained with the linearized PLL model. Note that the variance for the linear model is close to the exact variance for $\gamma_L > 3$. Hence, the linear model is adequate for practical purposes.

Approximate analyses of the statistical characteristics of the phase error for the nonlinear PLL have also been performed. Of particular importance is the transient behavior of the PLL during initial acquisition. Another important problem is the behavior of PLL at low SNR. It is known, for example, that when the SNR at the input to the PLL drops below a certain value, there is a rapid deterioration in the performance of the PLL. The loop begins to lose lock and an impulsive-type of noise, characterized as clicks, is generated which degrades the performance of the loop. Results on these topics can be found in the texts by Viterbi (1966), Lindsey (1972), Lindsey and Simon (1973), and Gardner (1979), and in the survey papers by Gupta (1975) and Lindsey and Chie (1981).

Up to this point, we have considered carrier phase estimation when the carrier signal is unmodulated. Below, we consider carrier phase recovery when the signal carries information.

6-2-4 Decision-Directed Loops

A problem arises in maximizing either (6-2-9) or (6-2-10) when the signal $s(t; \phi)$ carries the information sequence $\{I_n\}$. In this case we can adopt one of two approaches: either we assume that $\{I_n\}$ is known or we treat $\{I_n\}$ as a random sequence and average over its statistics.

In decision-directed parameter estimation, we assume that the information sequence $\{I_n\}$ over the observation interval has been estimated and, in the absence of demodulation errors, $\bar{I}_n = I_n$, where $\bar{I}_n$ denotes the detected value of the information I_n. In this case $s(t; \phi)$ is completely known except for the carrier phase. Decision-directed phase estimation was first described by Proakis *et al.* (1964).

To be specific, let us consider the decision-directed phase estimate for the class of linear modulation techniques for which the received *equivalent lowpass signal* may be expressed as

$$\begin{aligned} r(t) &= e^{-j\phi} \sum_n I_n g(t - nT) + z(t) \\ &= s_l(t) e^{-j\phi} + z(t) \end{aligned} \tag{6-2-33}$$

where $s_l(t)$ is a known signal if the sequence $\{I_n\}$ is assumed known. The likelihood function and corresponding log-likelihood function for the equivalent lowpass signal are

$$\Lambda(\phi) = C \exp\left\{\text{Re}\left[\frac{1}{N_0}\int_{T_0} r(t) s_l^*(t) e^{j\phi}\, dt\right]\right\} \tag{6-2-34}$$

$$\Lambda_L(\phi) = \text{Re}\left\{\left[\frac{1}{N_0}\int_{T_0} r(t) s_l^*(t)\, dt\right] e^{j\phi}\right\} \tag{6-2-35}$$

If we substitute for $s_l(t)$ in (6-2-35) and assume that the observation interval $T_0 = KT$, where K is a positive integer, we obtain

$$\begin{aligned} \Lambda_L(\phi) &= \text{Re}\left\{e^{j\phi} \frac{1}{N_0} \sum_{n=0}^{K-1} I_n^* \int_{nT}^{(n+1)T} r(t) g^*(t - nT)\, dt\right\} \\ &= \text{Re}\left\{e^{j\phi} \frac{1}{N_0} \sum_{n=0}^{K-1} I_n^* y_n\right\} \end{aligned} \tag{6-2-36}$$

where, by definition

$$y_n = \int_{nT}^{(n+1)T} r(t) g^*(t - nT)\, dt \tag{6-2-37}$$

Note that y_n is the output of the matched filter in the nth signal interval. The ML estimate of ϕ is easily found from (6-2-36) by differentiating the log-likelihood

$$\Lambda_L(\phi) = \text{Re}\left(\frac{1}{N_0} \sum_{n=0}^{K-1} I_n^* y_n\right) \cos\phi - \text{Im}\left(\frac{1}{N_0} \sum_{n=0}^{K-1} I_n^* y_n\right) \sin\phi$$

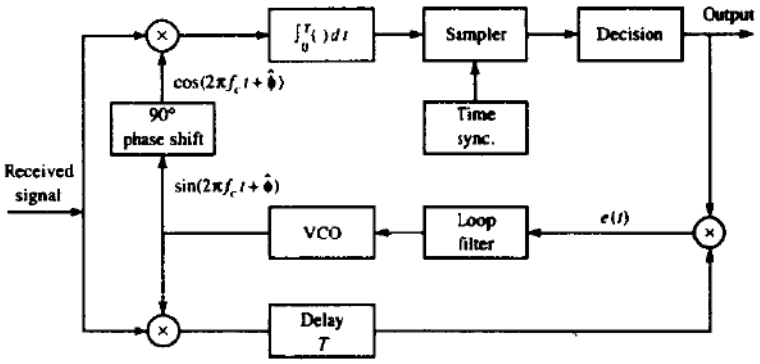


FIGURE 6-2-9 Carrier recovery with a decision-feedback PLL.

with respect to ϕ and setting the derivative equal to zero. Thus, we obtain

$$\hat{\phi}_{\mathrm{ML}} = -\tan^{-1}\left[\mathrm{Im}\left(\sum_{n=0}^{K-1} I_n^* y_n\right) \Big/ \mathrm{Re}\left(\sum_{n=0}^{K-1} I_n^* y_n\right)\right] \tag{6-2-38}$$

We call $\hat{\phi}_{\mathrm{ML}}$ in (6-2-38) the *decision-directed* (or *decision-feedback*) *carrier phase estimate.* It is easily shown (Problem 6-10) that the mean value of $\hat{\phi}_{\mathrm{ML}}$ is ϕ, so that the estimate is unbiased. Furthermore, the pdf of $\hat{\phi}_{\mathrm{ML}}$ can be obtained (Problem 6-11) by using the procedure described in Section 5-2-7.

A decision-feedback PLL (DFPLL) that is appropriate for a double-sideband PAM signal of the form $A(t)\cos(2\pi f_c t + \phi)$ is shown in Fig. 6-2-9. The received signal is multiplied by the quadrature carriers $c_c(t)$ and $c_s(t)$, as given by (6-2-5), which are derived from the VCO. The product signal

$$\begin{aligned} r(t)\cos(2\pi f_c t + \hat{\phi}) &= \tfrac{1}{2}[A(t) + n_c(t)]\cos\Delta\phi \\ &\quad - \tfrac{1}{2}n_s(t)\sin\Delta\phi + \text{double-frequency terms} \end{aligned} \tag{6-2-39}$$

is used to recover the information carried by $A(t)$. The detector makes a decision on the symbol that is received every T seconds. Thus, in the absence of decision errors, it reconstructs $A(t)$ free of any noise. This reconstructed signal is used to multiply the product of the second quadrature multiplier, which has been delayed by T seconds to allow the demodulator to reach a decision. Thus, the input to the loop filter in the absence of decision errors is the error signal

$$\begin{aligned} e(t) &= \tfrac{1}{2}A(t)\{[A(t) + n_c(t)]\sin\Delta\phi - n_s(t)\cos\Delta\phi\} \\ &\quad + \text{double-frequency terms} \\ &= \tfrac{1}{2}A^2(t)\sin\Delta\phi + \tfrac{1}{2}A(t)[n_c(t)\sin\Delta\phi - n_s(t)\cos\Delta\phi] \\ &\quad + \text{double-frequency terms} \end{aligned} \tag{6-2-40}$$

The loop filter is lowpass and, hence, it rejects the double-frequency term in $e(t)$. The desired component is $A^2(t)\sin\Delta\phi$, which contains the phase error for driving the loop.

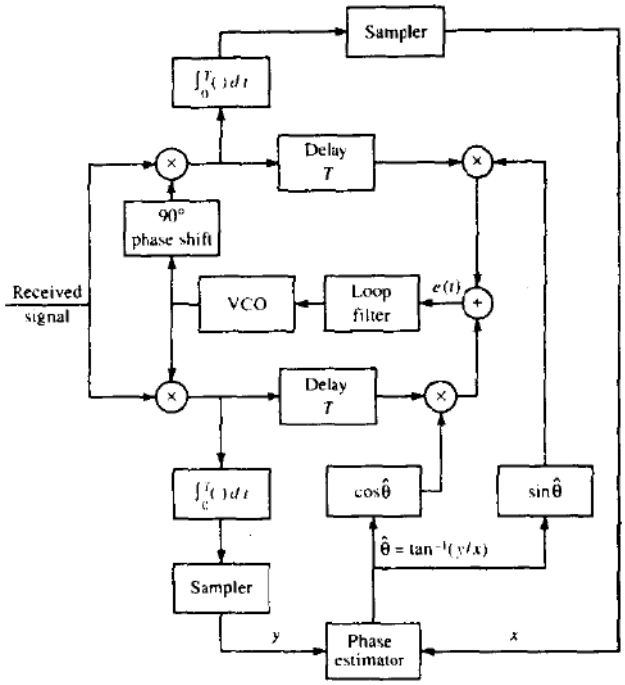


FIGURE 6-2-10 Carrier recovery for M-ary PSK using a decision-feedback PLL.

In the case of M-ary PSK, the DFPLL has the configuration shown in Fig. 6-2-10. The received signal is demodulated to yield the phase estimate

$$\hat{\theta}_m = \frac{2\pi}{M}(m-1)$$

which, in the absence of a decision error, is the transmitted signal phase. The two outputs of the quadrature multipliers are delayed by the symbol duration T and multiplied by $\cos\theta_m$ and $\sin\theta_m$ to yield

$$\begin{aligned} &r(t)\cos(2\pi f_c t + \hat{\phi})\sin\theta_m \\ &\quad = \tfrac{1}{2}[A\cos\theta_m + n_c(t)]\sin\theta_m\cos(\phi - \hat{\phi}) \\ &\qquad - \tfrac{1}{2}[A\sin\theta_m + n_s(t)]\sin\theta_m\sin(\phi - \hat{\phi}) \\ &\qquad + \text{double-frequency terms} \\ &r(t)\sin(2\pi f_c t + \hat{\phi})\cos\theta_m \\ &\quad = -\tfrac{1}{2}[A\cos\theta_m + n_c(t)]\cos\theta_m\sin(\phi - \hat{\phi}) \\ &\qquad - \tfrac{1}{2}[A\sin\theta_m + n_s(t)]\cos\theta_m\cos(\phi - \hat{\phi}) \\ &\qquad + \text{double-frequence terms} \end{aligned} \tag{6-2-41}$$

The two signals are added to generate the error signal

$$e(t) = -\tfrac{1}{2}A \sin(\phi - \hat{\phi}) + \tfrac{1}{2}n_c(t) \sin(\phi - \hat{\phi} - \theta_m) + \tfrac{1}{2}n_s(t) \cos(\phi - \hat{\phi} - \theta_m) + \text{double-frequency terms} \quad (6\text{-}2\text{-}42)$$

This error signal is the input to the loop filter that provides the control signal for the VCO.

We observe that the two quadrature noise components in (6-2-42) appear as additive terms. There is no term involving a product of two noise components as in an Mth-power law device, described in the next section. Consequently, there is no additional power loss associated with the decision-feedback PLL.

This M-phase tracking loop has a phase ambiguity of $360°/M$, necessitating the need to differentially encode the information sequence prior to transmission and differentially decode the received sequence after demodulation to recover the information.

The ML estimate in (6-2-38) is also appropriate for QAM. The ML estimate for offset QPSK is also easily obtained (Problem 6-12) by maximizing the log-likelihood function in (6-2-35), with $s_l(t)$ given as

$$s_l(t) = \sum_n I_n g(t - nT) + j \sum_n J_n g(t - nT - \tfrac{1}{2}T) \quad (6\text{-}2\text{-}43)$$

where $I_n = \pm 1$ and $J_n = \pm 1$.

Finally, we should also mention that carrier phase recovery for CPM signals can be accomplished in a decision-directed manner by use of a PLL. From the optimum demodulator for CPM signals, which is described in Section 5-3, we can generate an error signal that is filtered in a loop filter whose output drives a PLL.

6-2-5 Non-Decision-Directed Loops

Instead of using a decision-directed scheme to obtain the phase estimate, we may treat the data as random variables and simply average $\Lambda(\phi)$ over these random variables prior to maximization. In order to carry out this integration, we may use either the actual probability distribution function of the data, if it is known or, perhaps, we may assume some probability distribution that might be a reasonable approximation to the true distribution. The following example illustrates the first approach.

Example 6-2-2

Suppose the real signal $s(t)$ carries binary modulation. Then, in a signal interval, we have

$$s(t) = A \cos 2\pi f_c t, \quad 0 \leq t \leq T$$

where $A = \pm 1$ with equal probability. Clearly, the pdf of A is given as

$$p(A) = \tfrac{1}{2}\delta(A - 1) + \tfrac{1}{2}\delta(A + 1)$$

Now, the likelihood function $\Lambda(\phi)$ given by (6-2-9) is conditional on a given value of A and must be averaged over the two values. Thus,

$$\begin{aligned}\bar{\Lambda}(\phi) &= \int_{-\infty}^{\infty} \Lambda(\phi) p(A)\, dA \\ &= \tfrac{1}{2} \exp\left[\frac{2}{N_0}\int_0^T r(t)\cos(2\pi f_c t + \phi)\, dt\right] \\ &\quad + \tfrac{1}{2}\exp\left[-\frac{2}{N_0}\int_0^T r(t)\cos(2\pi f_c t + \phi)\, dt\right] \\ &= \cosh\left[\frac{2}{N_0}\int_0^T r(t)\cos(2\pi f_c t + \phi)\, dt\right]\end{aligned}$$

and the corresponding log-likelihood function is

$$\bar{\Lambda}_L(\phi) = \ln\cosh\left[\frac{2}{N_0}\int_0^T r(t)\cos(2\pi f_c t + \phi)\, dt\right] \tag{6-2-44}$$

If we differentiate $\bar{\Lambda}_L(\phi)$ and set the derivative equal to zero, we obtain the ML estimate for the non-decision-directed estimate. Unfortunately, the functional relationship in (6-2-44) is highly nonlinear and, hence, an exact solution is difficult to obtain. On the other hand, approximations are possible. In particular,

$$\ln\cosh x = \begin{cases} \frac{1}{2}x^2 & (|x| \ll 1) \\ |x| & (|x| \gg 1) \end{cases} \tag{6-2-45}$$

With these approximations, the solution for ϕ becomes tractable.

In this example, we averaged over the two possible values of the information symbol. When the information symbols are M-valued, where M is large, the averaging operation yields highly nonlinear functions of the parameter to be estimated. In such a case, we may simplify the problem by assuming that the information symbols are continuous random variables. For example, we may assume that the symbols are zero-mean gaussian. The following example illustrates this approximation and the resulting form for the average likelihood function.

Example 6-2-3

Let us consider the same signal as in Example 6-2-2, but now we assume that the amplitude A is zero-mean gaussian with unit variance. Thus,

$$p(A) = \frac{1}{\sqrt{2\pi}} e^{-A^2/2}$$

If we average $\Lambda(\phi)$ over the assumed pdf of A, we obtain the average likelihood $\bar{\Lambda}(\phi)$ in the form

$$\bar{\Lambda}(\phi) = C \exp\left\{\left[\frac{2}{N_0}\int_0^T r(t)\cos(2\pi f_c t + \phi)\,dt\right]^2\right\} \tag{6-2-46}$$

and the corresponding log-likelihood as

$$\bar{\Lambda}_L(\phi) = \left[\frac{2}{N_0}\int_0^T r(t)\cos(2\pi f_c t + \phi)\,dt\right]^2 \tag{6-2-47}$$

We can obtain the ML estimate of ϕ by differentiating $\bar{\Lambda}_L(\phi)$ and setting the derivative to zero.

It is interesting to note that the log-likelihood function is quadratic under the gaussian assumption and that it is approximately quadratic, as indicated in (6-2-45) for small values of the cross-correlation of $r(t)$ with $s(t;\phi)$. In other words, if the cross-correlation over a single interval is small, the gaussian assumption for the distribution of the information symbols yields a good approximation to the log-likelihood function.

In view of these results, we may use the gaussian approximation on all the symbols in the observation interval $T_0 = KT$. Specifically, we assume that the K information symbols are statistically independent and identically distributed. By averaging the likelihood function $\Lambda(\phi)$ over the gaussian pdf for each of the K symbols in the interval $T_0 = KT$, we obtain the result

$$\Lambda(\phi) = C \exp\left\{\sum_{n=0}^{K-1}\left[\frac{2}{N_0}\int_{nT}^{(n+1)T} r(t)\cos(2\pi f_c t + \phi)\,dt\right]^2\right\} \tag{6-2-48}$$

If we take the logarithm of (6-2-48), differentiate the resulting log-likelihood function, and set the derivative equal to zero, we obtain the condition for the ML estimate as

$$\sum_{n=0}^{K-1}\int_{nT}^{(n+1)T} r(t)\cos(2\pi f_c t + \hat{\phi})\,dt \int_{nT}^{(n+1)T} r(t)\sin(2\pi f_c t + \hat{\phi})\,dt = 0 \tag{6-2-49}$$

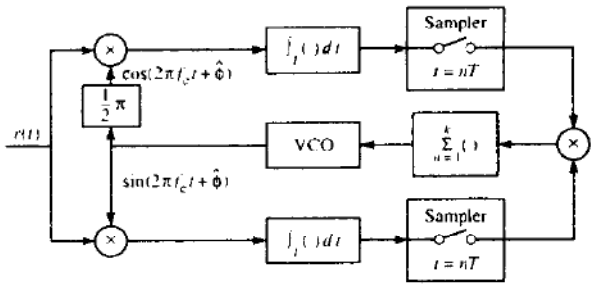


FIGURE 6-2-11 Non-decision-directed PLL for carrier phase estimations of PAM signals.

Although this equation can be manipulated further, its present form suggests the tracking loop configuration illustrated in Fig. 6-2-11. This loop resembles a Costas loop, which is described below. We note that the multiplication of the two signals from the integrators destroys the sign carried by the information symbols. The summer plays the role of the loop filter. In a tracking loop configuration, the summer may be implemented either as a sliding-window digital filter (summer) or as a lowpass digital filter with exponential weighting of the past data.

In a similar manner, one can derive non-decision directed ML phase estimates for QAM and M-PSK. The starting point is to average the likelihood function given by (6-2-9) over the statistical characteristics of the data. Here again, we may use the gaussian approximation (two-dimensional gaussian for complex-valued information symbols) in averaging over the information sequence.

Squaring Loop The squaring loop is a non-decision-directed loop that is widely used in practice to establish the carrier phase of double-sideband suppressed carrier signals such as PAM. To describe its operation, consider the problem of estimating the carrier phase of the digitally modulated PAM signal of the form

$$s(t) = A(t)\cos(2\pi f_c t + \phi) \tag{6-2-50}$$

where $A(t)$ carries the digital information. Note that $E[s(t)] = E[A(t)] = 0$ when the signal levels are symmetric about zero. Consequently, the average value of $s(t)$ does not produce any phase coherent frequency components at any frequency, including the carrier. One method for generating a carrier from the received signal is to square the signal and, thus, to generate a frequency component at $2f_c$, which can be used to drive a phase-locked loop (PLL) tuned to $2f_c$. This method is illustrated in the block diagram shown in Fig. 6-2-12.

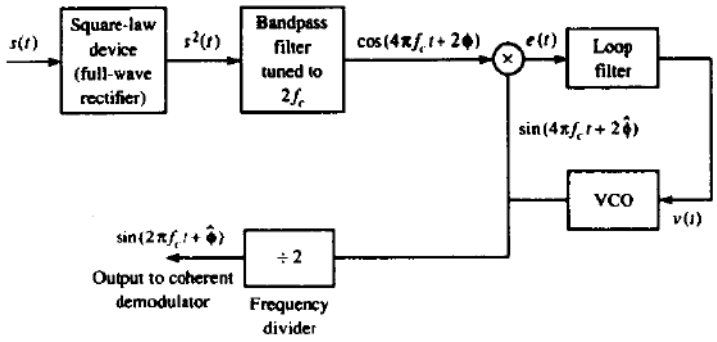


FIGURE 6-2-12 Carrier recovery using a square-law device.

The output of the square-law device is

$$s^2(t) = A^2(t)\cos^2(2\pi f_c t + \phi)$$
$$= \tfrac{1}{2}A^2(t) + \tfrac{1}{2}A^2(t)\cos(4\pi f_c t + 2\phi) \qquad (6\text{-}2\text{-}51)$$

Since the modulation is a cyclostationary stochastic process, the expected value of $s^2(t)$ is

$$E[s^2(t)] = \tfrac{1}{2}E[A^2(t)] + \tfrac{1}{2}E[A^2(t)]\cos(4\pi f_c t + 2\phi) \qquad (6\text{-}2\text{-}52)$$

Hence, there is power at the frequency $2f_c$.

If the output of the square-law device is passed through a bandpass filter tuned to the double-frequency term in (6-2-51), the mean value of the filter is a sinusoid with frequency $2f_c$, phase 2ϕ, and amplitude $\frac{1}{2}E[A^2(t)]H(2f_c)$, where $H(2f_c)$ is the gain of the filter at $f = 2f_c$. Thus, the square-law device has produced a periodic component from the input signal $s(t)$. In effect, the squaring of $s(t)$ has removed the sign information contained in $A(t)$ and, thus, has resulted in phase-coherent frequency components at twice the carrier. The filtered frequency component at $2f_c$ is then used to drive the PLL.

The squaring operation leads to a noise enhancement that increases the noise power level at the input to the PLL and results in an increase in the variance of the phase error.

To elaborate on this point, let the input to the squarer be $s(t) + n(t)$, where $s(t)$ is given by (6-2-50) and $n(t)$ represents the bandpass additive gaussian noise process. By squaring $s(t) + n(t)$, we obtain

$$y(t) = s^2(t) + 2s(t)n(t) + n^2(t) \qquad (6\text{-}2\text{-}53)$$

where $s^2(t)$ is the desired signal component and the other two components are the signal × noise and noise × noise terms. By computing the autocorrelation functions and power density spectra of these two noise components, one can

easily show that both components have spectral power in the frequency band centered at $2f_c$. Consequently, the bandpass filter with bandwidth B_{bp} centered at $2f_c$, which produces the desired sinusoidal signal component that drives the PLL, also passes noise due to these two terms.

Since the bandwidth of the loop is designed to be significantly smaller than the bandwidth B_{bp} of the bandpass filter, the total noise spectrum at the input to the PLL may be approximated as a constant within the loop bandwidth. This approximation allows us to obtain a simple expression for the variance of the phase error as

$$\sigma_\phi^2 = 1/\gamma_L S_L \tag{6-2-54}$$

where S_L is called the squaring loss and is given by

$$S_L = \left(1 + \frac{B_{bp}/2B_{eq}}{\gamma_L}\right)^{-1} \tag{6-2-55}$$

Since $S_L < 1$, S_L^{-1} represents the increase in the variance of the phase error caused by the added noise (noise × noise terms) that results from the squarer. Note, for example, that when $\gamma_L = B_{bp}/2B_{eq}$, the loss is 3 dB.

Finally, we observe that the output of the VCO from the squaring loop must be frequency-divided by 2 to generate the phase-locked carrier for signal demodulation. It should be noted that the output of the frequency divider has a phase ambiguity of 180° relative to the phase of the received signal. For this reason, the binary data must be differentially encoded prior to transmission and differentially decoded at the receiver.

Costas Loop Another method for generating a properly phased carrier for a double-sideband suppressed carrier signal is illustrated by the block diagram shown in Fig. 6-2-13. This scheme was developed by Costas (1956) and is

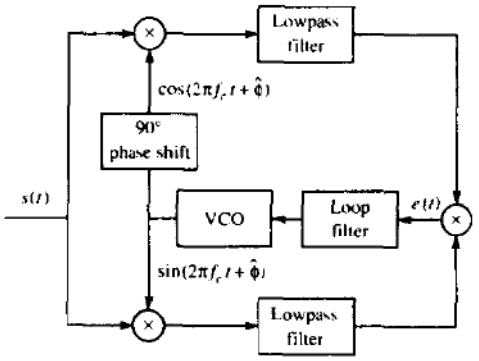


FIGURE 6-2-13 Block diagram of Costas loop.

called the *Costas loop*. The received signal is multiplied by $\cos(2\pi f_c t + \hat{\phi})$ and $\sin(2\pi f_c t + \hat{\phi})$, which are outputs from the VCO. The two products are

$$\begin{aligned} y_c(t) &= [s(t) + n(t)] \cos(2\pi f_c t + \hat{\phi}) \\ &= \tfrac{1}{2}[A(t) + n_c(t)] \cos \Delta\phi + \tfrac{1}{2} n_s(t) \sin \Delta\phi \\ &\quad + \text{double-frequency terms} \\ y_s(t) &= [s(t) + n(t)] \sin(2\pi f_c t + \hat{\phi}) \\ &= \tfrac{1}{2}[A(t) + n_c(t)] \sin \Delta\phi - \tfrac{1}{2} n_s(t) \cos \Delta\phi \\ &\quad + \text{double-frequency terms} \end{aligned} \tag{6-2-56}$$

where the phase error $\Delta\phi = \hat{\phi} - \phi$. The double-frequency terms are eliminated by the lowpass filters following the multiplications.

An error signal is generated by multiplying the two outputs of the lowpass filters. Thus,

$$\begin{aligned} e(t) &= \tfrac{1}{8}\{[A(t) + n_c(t)]^2 - n_s^2(t)\} \sin(2\,\Delta\phi) \\ &\quad - \tfrac{1}{4} n_s(t)[A(t) + n_c(t)] \cos(2\,\Delta\phi) \end{aligned} \tag{6-2-57}$$

This error signal is filtered by the loop filter, whose output is the control voltage that drives the VCO. The reader should note the similarity of the Costas loop to the PLL shown in Fig. 6-2-11.

We note that the error signal into the loop filter consists of the desired term $A^2(t) \sin 2(\hat{\phi} - \phi)$ plus terms that involve signal × noise and noise × noise. These terms are similar to the two noise terms at the input to the PLL for the squaring method. In fact, if the loop filter in the Costas loop is identical to that used in the squaring loop, the two loops are equivalent. Under this condition, the probability density function of the phase error and the performance of the two loops are identical.

It is interesting to note that the optimum lowpass filter for rejecting the double-frequency terms in the Costas loop is a filter matched to the signal pulse in the information-bearing signal. If matched filters are employed for the low pass filters, their outputs could be sampled at the bit rate, at the end of each signal interval, and the discrete-time signal samples could be used to drive the loop. The use of the matched filter results in a smaller noise into the loop.

Finally, we note that, as in the squaring PLL, the output of the VCO contains a phase ambiguity of 180°, necessitating the need for differential encoding of the data prior to transmission and differential decoding at the demodulator.

Carrier Estimation for Multiple Phase Signals When the digital information is transmitted via M-phase modulation of a carrier, the methods described above can be generalized to provide the properly phased carrier for

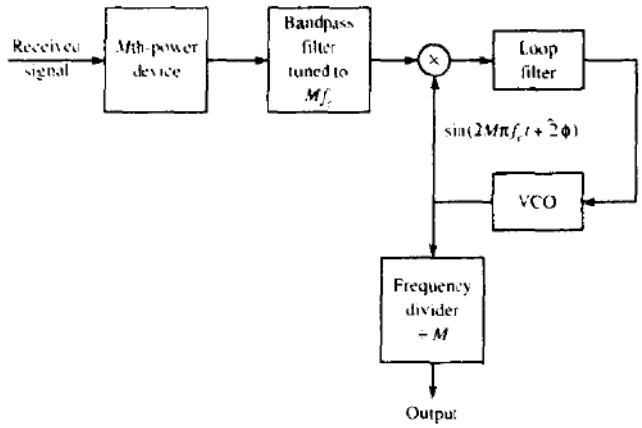


FIGURE 6-2-14 *Carrier recovery with Mth power law device for M-ary PSK.*

demodulation. The received M-phase signal, excluding the additive noise, may be expressed as

$$s(t) = A\cos\left[2\pi f_c t + \phi + \frac{2\pi}{M}(m-1)\right], \quad m = 1, 2, \ldots, M \qquad (6\text{-}2\text{-}58)$$

where $2\pi(m-1)/M$ represents the information-bearing component of the signal phase. The problem in carrier recovery is to remove the information-bearing component and, thus, to obtain the unmodulated carrier $\cos(2\pi f_c t + \phi)$. One method by which this can be accomplished is illustrated in Fig. 6-2-14, which represents a generalization of the squaring loop. The signal is passed through an Mth-power-law device, which generates a number of harmonics of f_c. The bandpass filter selects the harmonic $\cos(2\pi M f_c t + M\phi)$ for driving the PLL. The term

$$\frac{2\pi}{M}(m-1)M = 2\pi(m-1) \equiv 0 \pmod{2\pi}, \quad m = 1, 2, \ldots, M$$

Thus, the information is removed. The VCO output is $\sin(2\pi M f_c t + M\hat{\phi})$, so this output is divided in frequency by M to yield $\sin(2\pi f_c t + \hat{\phi})$, and phase-shifted by $\frac{1}{2}\pi$ rad to yield $\cos(2\pi f_c t + \hat{\phi})$. These components are then fed to the demodulator. Although not explicitly shown, there is a phase ambiguity in these reference sinusoids of $360°/M$, which can be overcome by differential encoding of the data at the transmitter and differential decoding after demodulation at the receiver.

Just as in the case of the squaring PLL, the Mth-power PLL operates in the presence of noise that has been enhanced by the Mth-power-law device, which results in the output

$$y(t) = [s(t) + n(t)]^M$$

The variance of the phase error in the PLL resulting from the additive noise may be expressed in the simple form

$$\sigma_{\hat{\phi}}^2 = \frac{S_{\mathrm{ML}}^{-1}}{\gamma_L} \tag{6-2-59}$$

where γ_L is the loop SNR and S_{ML}^{-1} is the *M-phase power loss*. S_{ML} has been evaluated by Lindsey and Simon (1973) for $M = 4$ and 8.

Another method for carrier recovery in M-ary PSK is based on a generalization of the Costas loop. That method requires multiplying the received signal by M phase-shifted carriers of the form

$$\sin\left[2\pi f_c t + \hat{\phi} + \frac{\pi}{M}(k-1)\right], \quad k = 1, 2, \ldots, M$$

lowpass-filtering each product, and then multiplying the outputs of the lowpass filters to generate the error signal. The error signal excites the loop filter, which, in turn, provides the control signal for the VCO. This method is relatively complex to implement and, consequently, has not been generally used in practice.

Comparison of Decision-Directed with Non-Decision-Directed Loops We note that the decision-feedback phase-locked loop (DFPLL) differs from the Costas loop only in the method by which $A(t)$ is rectified for the purpose of removing the modulation. In the Costas loop, each of the two quadrature signals used to rectify $A(t)$ is corrupted by noise. In the DFPLL, only one of the signals used to rectify $A(t)$ is corrupted by noise. On the other hand, the squaring loop is similar to the Costas loop in terms of the noise effect on the estimate $\hat{\phi}$. Consequently, the DFPLL is superior in performance to both the Costas loop and the squaring loop, provided that the demodulator is operating at error rates below 10^{-2} where an occasional decision error has a negligible effect on $\hat{\phi}$. Quantitative comparisons of the variance of the phase errors in a Costas loop to those in a DFPLL have been made by Lindsey and Simon (1973), and show that the variance of the DFPLL is 4–10 times smaller for signal-to-noise ratios per bit above 0 db.

6-3 SYMBOL TIMING ESTIMATION

In a digital communication system, the output of the demodulator must be sampled periodically at the symbol rate, at the precise sampling time instants $t_m = mT + \tau$, where T is the symbol interval and τ is a nominal time delay that accounts for the propagation time of the signal from the transmitter to the receiver. To perform this periodic sampling, we require a clock signal at the

receiver. The process of extracting such a clock signal at the receiver is usually called *symbol synchronization* or *timing recovery*.

Timing recovery is one of the most critical functions that is performed at the receiver of a synchronous digital communication system. We should note that the receiver must know not only the frequency $(1/T)$ at which the outputs of the matched filters or correlators are sampled, but also where to take the samples within each symbol interval. The choice of sampling instant within the symbol interval of duration T is called the *timing phase*.

Symbol synchronization can be accomplished in one of several ways. In some communication systems, the transmitter and receiver clocks are synchronized to a master clock, which provides a very precise timing signal. In this case, the receiver must estimate and compensate for the relative time delay between the transmitted and received signals. Such may be the case for radio communication systems that operate in the very low frequency (VLF) band (below 30 kHz), where precise clock signals are transmitted from a master radio station.

Another method for achieving symbol synchronization is for the transmitter to simultaneously transmit the clock frequency $1/T$ or a multiple of $1/T$ along with the information signal. The receiver may simply employ a narrowband filter tuned to the transmitted clock frequency and, thus, extract the clock signal for sampling. This approach has the advantage of being simple to implement. There are several disadvantages, however. One is that the transmitter must allocate some of its available power to the transmission of the clock signal. Another is that some small fraction of the available channel bandwidth must be allocated for the transmission of the clock signal. In spite of these disadvantages, this method is frequently used in telephone transmission systems that employ large bandwidths to transmit the signals of many users. In such a case, the transmission of a clock signal is shared in the demodulation of the signals among the many users. Through this shared use of the clock signal, the penalty in transmitter power and in bandwidth allocation is reduced proportionally by the number of users.

A clock signal can also be extracted from the received data signal. There are a number of different methods that can be used at the receiver to achieve self-synchronization. In this section, we treat both decision-directed and non-decision-directed methods.

6-3-1 Maximum-Likelihood Timing Estimation

Let us begin by obtaining the ML estimate of the time delay τ. If the signal is a baseband PAM waveform, it is represented as

$$r(t) = s(t; \tau) + n(t) \tag{6-3-1}$$

where

$$s(t; \tau) = \sum_n I_n g(t - nT - \tau) \tag{6-3-2}$$

As in the case of ML phase estimation, we distinguish between two types of timing estimators, decision-directed timing estimators and non-decision-directed estimators. In the former, the information symbols from the output of the demodulator are treated as the known transmitted sequence. In this case, the log-likelihood function has the form

$$\Lambda_L(\tau) = C_L \int_{T_0} r(t)s(t;\tau)\,dt \tag{6-3-3}$$

If we substitute (6-3-2) into (6-3-3), we obtain

$$\begin{aligned}\Lambda_L(\tau) &= C_L \sum_n I_n \int_{T_0} r(t)g(t-nT-\tau)\,dt \\ &= C_L \sum_n I_n y_n(\tau)\end{aligned} \tag{6-3-4}$$

where $y_n(t)$ is defined as

$$y_n(\tau) = \int_{T_0} r(t)g(t-nT-\tau)\,dt \tag{6-3-5}$$

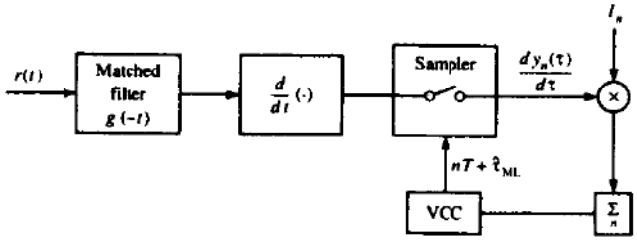
A necessary condition for $\hat{\tau}$ to be the ML estimate of τ is that

$$\begin{aligned}\frac{d\Lambda_L(\tau)}{d\tau} &= \sum_n I_n \frac{d}{d\tau}\int_{T_0} r(t)g(t-nT-\tau)\,dt \\ &= \sum_n I_n \frac{d}{d\tau}[y_n(\tau)] = 0\end{aligned} \tag{6-3-6}$$

The result in (6-3-6) suggests the implementation of the tracking loop shown in Fig. 6-3-1. We should observe that the summation in the loop serves as the loop filter whose bandwidth is controlled by the length of the sliding window in the summation. The output of the loop filter drives the voltage-controlled clock (VCC), or voltage-controlled oscillator, which controls the sampling times for the input to the loop. Since the detected information sequence $\{I_n\}$ is used in the estimation of τ, the estimate is decision-directed.

The techniques described above for ML timing estimation of baseband

FIGURE 6-3-1 Decision-directed ML estimation of timing for baseband PAM.



PAM signals can be extended to carrier modulated signal formats such as QAM and PSK in a straightforward manner, by dealing with the equivalent lowpass form of the signals. Thus, the problem of ML estimation of symbol timing for carrier signals is very similar to the problem formulation for the baseband PAM signal.

6-3-2 Non-Decision-Directed Timing Estimation

A non-decision-directed timing estimate can be obtained by averaging the likelihood ratio $\Lambda(\tau)$ over the pdf of the information symbols, to obtain $\bar{\Lambda}(\tau)$, and then differentiating either $\bar{\Lambda}(\tau)$ or $\ln \bar{\Lambda}(\tau) = \bar{\Lambda}_L(\tau)$ to obtain the condition for the maximum-likelihood estimate $\hat{\tau}_{ML}$.

In the case of binary (baseband) PAM, where $I_n = \pm 1$ with equal probability, the average over the data yields

$$\bar{\Lambda}_L(\tau) = \sum_n \ln \cosh C y_n(\tau) \tag{6-3-7}$$

just as in the case of the phase estimator, Since $\ln \cosh x \approx \frac{1}{2}x^2$ for small x, the square-law approximation

$$\bar{\Lambda}_L(\tau) \approx \tfrac{1}{2}C^2 \sum_n y_n^2(\tau) \tag{6-3-8}$$

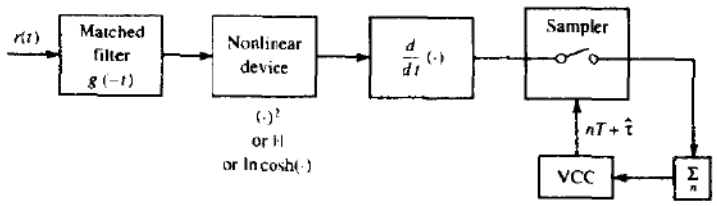
is appropriate for low signal-to-noise ratios. For multilevel PAM, we may approximate the statistical characteristics of the information symbols $\{I_n\}$ by the gaussian pdf, with zero mean and unit variance. When we average $\Lambda(\tau)$ over the gaussian pdf, the logarithm of $\bar{\Lambda}(\tau)$ is identical to $\bar{\Lambda}_L(\tau)$ given by (6-3-8). Consequently, the non-decision-directed estimate of τ may be obtained by differentiating (6-3-8). The result is an approximation to the ML estimate of the delay time. The derivative of (6-3-8) is

$$\frac{d}{d\tau}\sum_n y_n^2(\tau) = 2\sum_n y_n(\tau)\frac{dy_n(\tau)}{d\tau} = 0 \tag{6-3-9}$$

where $y_n(\tau)$ is given by (6-3-5).

An implementation of a tracking loop based on the derivative of $\bar{\Lambda}_L(\tau)$ given by (6-3-7) is shown in Fig. 6-3-2. Alternatively, an implementation of a

FIGURE 6-3-2 Non-decision-directed estimation of timing for binary baseband PAM.



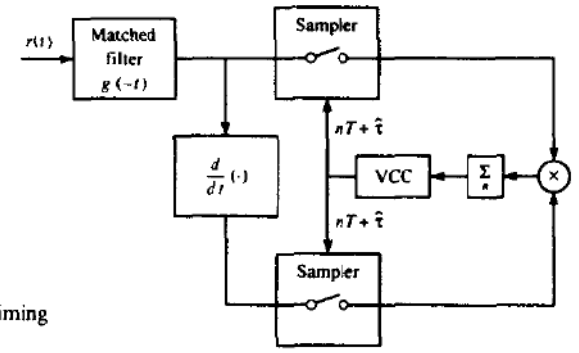


FIGURE 6-3-3 Non-decision-directed estimation of timing for baseband PAM.

tracking loop based on (6-3-9) is illustrated in Fig. 6-3-3. In both structures, we observe that the summation serves as the loop filter that drives the VCC. It is interesting to note the resemblance of the timing loop in Fig. 6-3-3 to the Costas loop for phase estimation.

Early–Late Gate Synchronizers Another non-decision-directed timing estimator exploits the symmetry properties of the signal at the output of the matched filter or correlator. To describe this method, let us consider the rectangular pulse $s(t)$, $0 \leq t \leq T$, shown in Fig. 6-3-4(a). The output of the filter matched to $s(t)$ attains its maximum value at time $t = T$, as shown in Fig. 6-3-4(b). Thus, the output of the matched filter is the time autocorrelation function of the pulse $s(t)$. Of course, this statement holds for any arbitrary pulse shape, so the approach that we describe applies in general to any signal pulse. Clearly, the proper time to sample the output of the matched filter for a maximum output is at $t = T$, i.e., at the peak of the correlation function.

In the presence of noise, the identification of the peak value of the signal is generally difficult. Instead of sampling the signal at the peak, suppose we sample early, at $t = T - \delta$ and late at $t = T + \delta$. The absolute values of the early samples $|y(m(T - \delta))|$ and the late samples $|y(m(T + \delta))|$ will be smaller (on the average in the presence of noise) than the samples of the peak value $|y(mT)|$. Since the autocorrelation function is even with respect to the optimum sampling time $t = T$, the absolute values of the correlation function at $t = T - \delta$ and $t = T + \delta$ are equal. Under this condition, the proper sampling

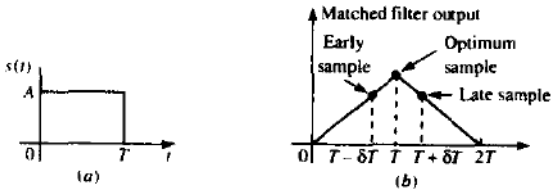


FIGURE 6-3-4 Rectangular signal pulse (a) and its matched filter output (b).

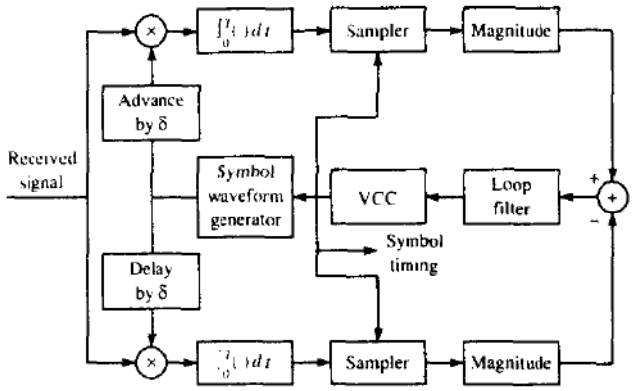


FIGURE 6-3-5 Block diagram of early–late gate synchronizer.

time is the midpoint between $t = T - \delta$ and $t = T + \delta$. This condition forms the basis for the *early–late gate symbol synchronizer.*

Figure 6-3-5 illustrates the block diagram of an early–late gate synchronizer. In this figure, correlators are used in place of the equivalent matched filters. The two correlators integrate over the symbol interval T, but one correlator starts integrating δ seconds early relative to the estimated optimum sampling time and the other integrator starts integrating δ seconds late relative to the estimated optimum sampling time. An error signal is formed by taking the difference between the absolute values of the two correlator outputs. To smooth the noise corrupting the signal samples, the error signal is passed through a lowpass filter. If the timing is off relative to the optimum sampling time, the average error signal at the output of the lowpass filter is nonzero, and the clock signal is either retarded or advanced, depending on the sign of the error. Thus, the smoothed error signal is used to drive a voltage-controlled clock (VCC), whose output is the desired clock signal that is used for sampling. The output of the VCC is also used as a clock signal for a symbol waveform generator that puts out the same basic pulse waveform as that of the transmitting filter. This pulse waveform is advanced and delayed and then fed to the two correlators, as shown in Fig. 6-3-5. Note that if the signal pulses are rectangular, there is no need for a signal pulse generator within the tracking loop.

We observe that the early–late gate synchronizer is basically a closed-loop control system whose bandwidth is relatively narrow compared to the symbol rate $1/T$. The bandwidth of the loop determines the quality of the timing estimate. A narrowband loop provides more averaging over the additive noise and, thus, improves the quality of the estimated sampling instants, provided that the channel propagation delay is constant and the clock oscillator at the transmitter is not drifting with time (or drifting very slowly with time). On the other hand, if the channel propagation delay is changing with time and/or the

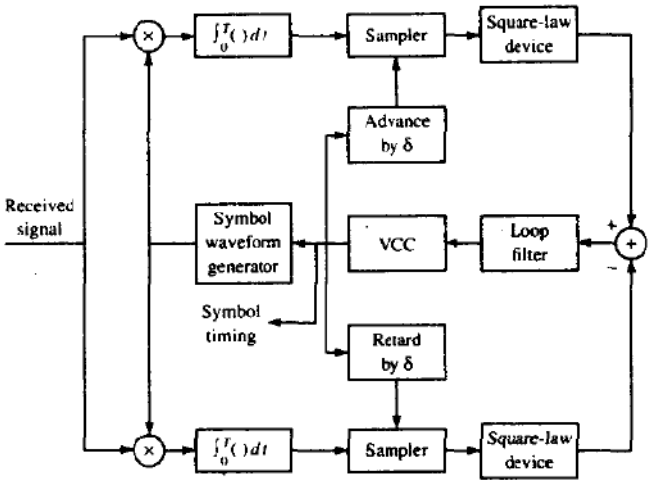


FIGURE 6-3-6 Block diagram of early–late gate synchronizer—an alternative form.

transmitter clock is also drifting with time then the bandwidth of the loop must be increased to provide for faster tracking of time variations in symbol timing.

In the tracking mode, the two correlators are affected by adjacent symbols. However, if the sequence of information symbols has zero mean, as is the case for PAM and some other signal modulations, the contribution to the output of the correlators from adjacent symbols averages out to zero in the lowpass filter.

An equivalent realization of the early–late gate synchronizer that is somewhat easier to implement is shown in Fig. 6-3-6. In this case the clock signal from the VCC is advanced and delayed by δ, and these clock signals are used to sample the outputs of the two correlators.

The early–late gate synchronizer described above is a non-decision-directed estimator of symbol timing that approximates the maximum-likelihood estimator. This assertion can be demonstrated by approximating the derivative of the log-likelihood function by the finite difference, i.e.,

$$\frac{d\bar{\Lambda}_L(\tau)}{d\tau} \approx \frac{\bar{\Lambda}_L(\tau+\delta) - \bar{\Lambda}_L(\tau-\delta)}{2\delta} \tag{6-3-10}$$

If we substitute for $\bar{\Lambda}_L(\tau)$ from (6-3-8) into (6-3-10), we obtain the approximation for the derivative as

$$\begin{aligned}\frac{d\bar{\Lambda}_L(\tau)}{d\tau} &= \frac{C^2}{4\delta}\sum_n \left[y_n^2(\tau+\delta) - y_n^2(\tau-\delta)\right] \\ &\approx \frac{C^2}{4\delta}\sum_n \left\{\left[\int_{T_0} r(t)g(t-nT-\tau-\delta)\,dt\right]^2 \right. \\ &\quad \left. - \left[\int_{T_0} r(t)g(t-nT-\tau+\delta)\,dt\right]^2\right\}\end{aligned} \tag{6-3-11}$$

But the mathematical expression in (6-3-11) basically describes the functions performed by the early–late gate symbol synchronizers illustrated in Figs 6-3-5 and 6-3-6.

6-4 JOINT ESTIMATION OF CARRIER PHASE AND SYMBOL TIMING

The estimation of the carrier phase and symbol timing may be accomplished separately as described above or jointly. Joint ML estimation of two or more signal parameters yields estimates that are as good and usually better than the estimates obtained from separate optimization of the likelihood function. In other words, the variances of the signal parameters obtained from joint optimization are less than or equal to the variance of parameter estimates obtained from separately optimizing the likelihood function.

Let us consider the joint estimation of the carrier phase and symbol timing. The log-likelihood function for these two parameters may be expressed in terms of the equivalent lowpass signals as

$$\Lambda_L(\phi, \tau) = \text{Re}\left[\frac{1}{N_0}\int_{T_0} r(t)s_l^*(t; \phi, \tau)\, dt\right] \tag{6-4-1}$$

where $s_l(t; \phi, \tau)$ is the equivalent lowpass signal, which has the general form

$$s_l(t; \phi, \tau) = e^{-j\phi}\left[\sum_n I_n g(t - nT - \tau) + j\sum_n J_n w(t - nT - \tau)\right] \tag{6-4-2}$$

where $\{I_n\}$ and $\{J_n\}$ are the two information sequences.

We note that, for PAM, we may set $J_n = 0$ for all n, and the sequence $\{I_n\}$ is real. For QAM and PSK, we set $J_n = 0$ for all n and the sequence $\{I_n\}$ is complex-valued. For offset QPSK, both sequences $\{I_n\}$ and $\{J_n\}$ are nonzero and $w(t) = g(t - \frac{1}{2}T)$.

For decision-directed ML estimation of ϕ and τ, the log-likelihood function becomes

$$\Lambda_L(\phi, \tau) = \text{Re}\left\{\frac{e^{j\phi}}{N_0}\sum_n [I_n^* y_n(\tau) + jJ_n^* x_n(\tau)]\right\} \tag{6-4-3}$$

where

$$\begin{aligned} y_n(\tau) &= \int_{T_0} r(t)g^*(t - nT - \tau)\, dt \\ x_n(\tau) &= \int_{T_0} r(t)w^*(t - nT - \tau)\, dt \end{aligned} \tag{6-4-4}$$

Necessary conditions for the estimates of ϕ and τ to be the ML estimates are

$$\frac{\partial \Lambda_L(\phi, \tau)}{\partial \phi} = 0, \qquad \frac{\partial \Lambda_L(\phi, \tau)}{\partial \tau} = 0 \tag{6-4-5}$$

It is convenient to define

$$A(\tau) + jB(\tau) = \frac{1}{N_0} \sum_n [I_n^* y_n(\tau) + jJ_n^* x_n(\tau)] \tag{6-4-6}$$

With this definition, (6-4-3) may be expressed in the simple form

$$\Lambda_L(\phi, \tau) = A(\tau)\cos\phi - B(\tau)\sin\phi \tag{6-4-7}$$

Now the conditions in (6-4-5) for the joint ML estimates become

$$\frac{\partial \Lambda(\phi, \tau)}{\partial \phi} = -A(\tau)\sin\phi - B(\tau)\cos\phi = 0 \tag{6-4-8}$$

$$\frac{\partial \Lambda(\phi, \tau)}{\partial \tau} = \frac{\partial A(\tau)}{\partial \tau}\cos\phi - \frac{\partial B(\tau)}{\partial \tau}\sin\phi = 0 \tag{6-4-9}$$

From (6-4-8), we obtain

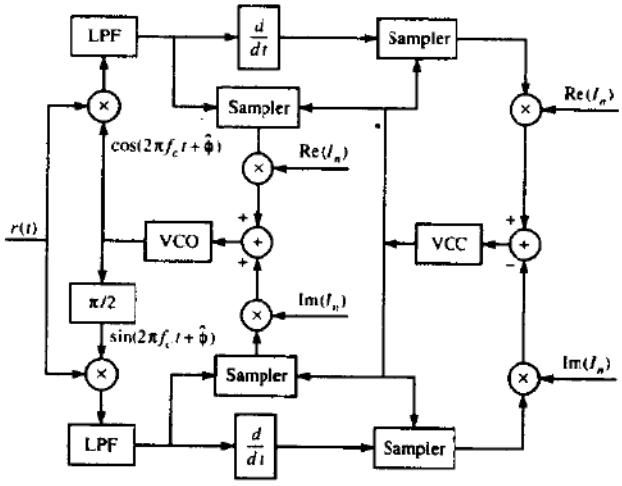
$$\hat{\phi}_{\mathrm{ML}} = -\tan^{-1}\left[\frac{B(\hat{\tau}_{\mathrm{ML}})}{A(\hat{\tau}_{\mathrm{ML}})}\right] \tag{6-4-10}$$

The solution to (6-4-9) that incorporates (6-4-10) is

$$\left[A(\tau)\frac{\partial A(\tau)}{\partial \tau} + B(\tau)\frac{\partial B(\tau)}{\partial \tau}\right]_{\tau=\hat{\tau}_{\mathrm{ML}}} = 0 \tag{6-4-11}$$

The decision-directed tracking loop for QAM (or PSK) obtained from these equations is illustrated in Fig. 6-4-1.

FIGURE 6-4-1 Decision-directed joint tracking loop for carrier phase and symbol timing in QAM and PSK.



Offset QPSK requires a slightly more complex structure for joint estimation of ϕ and τ. The structure is easily derived from (6-4-6)–(6-4-11).

In addition to the joint estimates given above, it is also possible to derive non-decision-directed estimates of the carrier phase and symbol timing, although we shall not pursue this approach.

We should also mention that one can combine the parameter estimation problem with the demodulation of the information sequence $\{I_n\}$. Thus, one can consider the joint maximum-likelihood estimation of $\{I_n\}$, the carrier phase ϕ, and the symbol timing parameter τ. Results on these joint estimation problems have appeared in the technical literature, e.g. Kobayashi (1971), Falconer (1976), and Falconer and Salz (1977).

6-5 PERFORMANCE CHARACTERISTICS OF ML ESTIMATORS

The quality of a signal parameter estimate is usually measured in terms of its bias and its variance. In order to define these terms, let us assume that we have a sequence of observations $[x_1\ x_2\ x_3\ \dots\ x_n] = \mathbf{x}$, with pdf $p(\mathbf{x} \mid \phi)$, from which we extract an estimate of a parameter ϕ. The bias of an estimate, say $\hat{\phi}(\mathbf{x})$, is defined as

$$\text{bias} = E[\hat{\phi}(\mathbf{x})] - \phi \tag{6-5-1}$$

where ϕ is the true value of the parameter. When $E[\hat{\phi}(\mathbf{x})] = \phi$, we say that the estimate is *unbiased*. The variance of the estimate $\hat{\phi}(x)$ is defined as

$$\sigma_{\hat{\phi}}^2 = E\{[\hat{\phi}(\mathbf{x})]^2\} - \{E[\hat{\phi}(\mathbf{x})]\}^2 \tag{6-5-2}$$

In general $\sigma_{\hat{\phi}}^2$ may be difficult to compute. However, a well-known result in parameter estimation (see Helstrom, 1968) is the Cramér–Rao lower bound on the mean square error defined as

$$E\{[\hat{\phi}(\mathbf{x}) - \phi]^2\} \geq \left[\frac{\partial}{\partial \phi} E[\hat{\phi}(\mathbf{x})]\right]^2 \Big/ E\left\{\left[\frac{\partial}{\partial \phi} \ln p(\mathbf{x} \mid \phi)\right]^2\right\} \tag{6-5-3}$$

Note that when the estimate is unbiased, the numerator of (6-5-3) is unity and the bound becomes a lower bound on the variance $\sigma_{\hat{\phi}}^2$ of the estimate $\hat{\phi}(\mathbf{x})$, i.e.,

$$\sigma_{\hat{\phi}}^2 \geq 1 \Big/ E\left\{\left[\frac{\partial}{\partial \phi} \ln p(\mathbf{x} \mid \phi)\right]^2\right\} \tag{6-5-4}$$

Since $\ln p(\mathbf{x} \mid \phi)$ differs from the log-likelihood function by a constant factor

independent of ϕ, it follows that

$$E\left\{\left[\frac{\partial}{\partial\phi}\ln p(\mathbf{x}\mid\phi)\right]^2\right\}=E\left\{\left[\frac{\partial}{\partial\phi}\ln\Lambda(\phi)\right]^2\right\}$$
$$=-E\left\{\frac{\partial^2}{\partial\phi^2}\ln\Lambda(\phi)\right\} \tag{6-5-5}$$

Therefore, the lower bound on the variance is

$$\sigma_{\hat{\phi}}^2\geq 1\Big/E\left\{\left[\frac{\partial}{\partial\phi}\ln\Lambda(\phi)\right]^2\right\}=-1\Big/E\left[\frac{\partial^2}{\partial\phi^2}\ln\Lambda(\phi)\right] \tag{6-5-6}$$

This lower bound is a very useful result. It provides a benchmark for comparing the variance of any practical estimate to the lower bound. Any estimate that is unbiased and whose variance attains the lower bound is called an *efficient estimate.*

In general, efficient estimates are rare. When they exist, they are maximum-likelihood estimates. A well-known result from parameter estimation theory is that any ML parameter estimate is asymptotically (arbitrarily large number of observations) unbiased and efficient. To a large extent, these desirable properties constitute the importance of ML parameter estimates. It also known that an ML estimate is asymptotically gaussian-distributed [with mean ϕ and variance equal to the lower bound given by (6-5-6).]

In the case of the ML estimates described in this chapter for the two signal parameters, their variance is generally inversely proportional to the signal-to-noise ratio, or, equivalently, inversely proportional to the signal power multiplied by the observation interval T_0. Furthermore, the variance of the decision-directed estimates, at low error probabilities, are generally lower than the variance of non-decision-directed estimates. In fact, the performance of the ML decision-directed estimates for ϕ and τ attain the lower bound.

The following example is concerned with the evaluation of the Cramér–Rao lower bound for the ML estimate of the carrier phase.

Example 6-5-1

The ML estimate of the phase of an unmodulated carrier was shown in (6-2-11) to satisfy the condition

$$\int_{T_0} r(t)\sin(2\pi f_c t+\hat{\phi}_{\mathrm{ML}})\,dt=0 \tag{6-5-7}$$

where

$$r(t)=s(t;\phi)+n(t)$$
$$=A\cos(2\pi f_c t+\phi)+n(t) \tag{6-5-8}$$

The condition in (6-5-7) was derived by maximizing the log likelihood function

$$\Lambda_L(\phi)=\frac{2}{N_0}\int_{T_0} r(t)s(t;\phi)\,dt \tag{6-5-9}$$

The variance of $\hat{\phi}_{\mathrm{ML}}$ is lower-bounded as

$$\begin{aligned}\sigma^2_{\hat{\phi}_{\mathrm{ML}}} &\geq \left\{\frac{2A}{N_0}\int_{T_0} E[r(t)]\cos(2\pi f_c t + \phi)\,dt\right\}^{-1} \\ &\geq \left\{\frac{A^2}{N_0}\int_{T_0} dt\right\}^{-1} = \frac{N_0}{A^2 T_0} \\ &\geq \frac{N_0/2T_0}{\frac{1}{2}A^2} = \frac{N_0 B_{\mathrm{eq}}}{\frac{1}{2}A^2}\end{aligned} \tag{6-5-10}$$

The factor $1/2T_0$ is simply the (one-sided) equivalent noise bandwidth of the ideal integrator.

From this example, we observe that the variance of the ML phase estimate is lower-bounded as

$$\sigma^2_{\hat{\phi}_{\mathrm{ML}}} \geq 1/\gamma_L \tag{6-5-11}$$

where $\gamma_L = A^2/2N_0B_{\mathrm{eq}}$ is the loop SNR. This is also the variance obtained for the phase estimate from a PLL with decision-directed estimation. As we have already observed, non-decision-directed estimates do not perform as well due to losses in the nonlinearities required to remove the modulation, e.g., the squaring loss and the Mth-power loss.

Similar results can be obtained on the quality of the symbol timing estimates derived above. In addition to their dependence on the SNR, the quality of symbol timing estimates is a function of the signal pulse shape. For example, a pulse shape that is commonly used in practice is one that has a raised cosine spectrum (see Section 9-2). For such a pulse, the rms timing error (σ_τ) as a function of SNR is illustrated in Fig. 6-5-1, for both decision-directed and

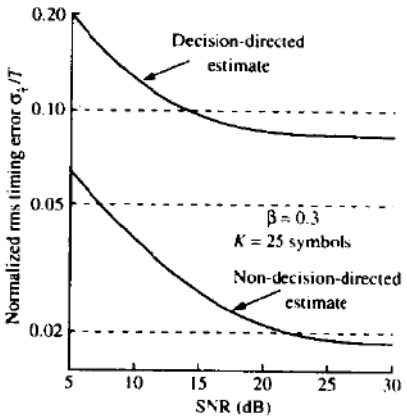


FIGURE 6-5-1 Performance of baseband symbol timing estimate for fixed signal and loop bandwidths. [*From* Synchronization Subsystems: Analysis and Design, *by L. Franks, 1983. Reprinted with permission of the author.*]

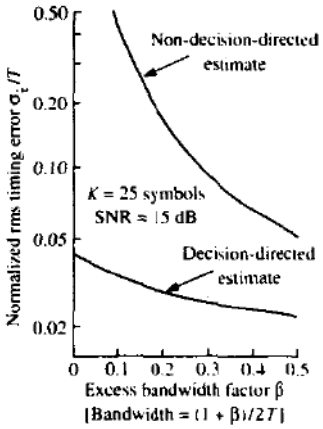


FIGURE 6-5-2 Performance of baseband symbol timing estimate for fixed SNR and fixed loop bandwidth. [*From* Synchronization Subsystems: Analysis and Design, *by L. Franks, 1983. Reprinted with permission of the author.*]

non-decision-directed estimates. Note the significant improvement in performance of the decision-directed estimate compared with the non-decision-directed estimate. Now, if the bandwidth of the pulse is varied, the pulse shape is changed and, hence, the rms value of the timing error also changes. For example, when the bandwidth of the pulse that has a raised cosine spectrum is varied, the rms timing error varies as shown in Fig. 6-5-2. Note that the error decreases as the bandwidth of the pulse increases.

In conclusion, we have presented the ML method for signal parameter estimation and have applied it to the estimation of the carrier phase and symbol timing. We have also described their performance characteristics.

6-6 BIBLIOGRAPHICAL NOTES AND REFERENCES

Carrier recovery and timing synchronization are two topics that have been thoroughly investigated over the past three decades. The Costas loop was invented in 1956 and the decision-directed phase estimation methods were described by Proakis *et al.* (1964) and by Natali and Walbesser (1969). The work on decision-directed estimation was motivated by earlier work of Price (1962a, b). Comprehensive treatments of phase-locked loops first appeared in the books by Viterbi (1966) and Gardner (1979). Books that cover carrier phase recovery and time synchronization techniques have been written by Stiffler (1971), Lindsey (1972), Lindsey and Simon (1973), and Meyr and Ascheid (1990).

A number of tutorial papers have appeared in IEEE journals on the PLL and on time synchronization. We cite, for example, the paper by Gupta (1975), which treats both analog and digital implementation of PLLs, and the paper by Lindsey and Chie (1981), which is devoted to the analysis of digital PLLs. In addition, the tutorial paper by Franks (1980) describes both carrier phase and symbol synchronization methods, including methods based on the maximum-likelihood estimation criterion. The paper by Franks is contained in a special

issue of the *IEEE Transactions on Communications* (August 1980) devoted to synchronization. The paper by Mueller and Muller (1976) describes digital signal processing algorithms for extracting symbol timing.

Application of the maximum-likelihood criterion to parameter estimation was first described in the context of radar parameter estimation (range and range rate). Subsequently, this optimal criterion was applied to carrier phase and symbol timing estimation as well as to joint parameter estimation with data symbols. Papers on these topics have been published by several researchers, including Falconer (1976), Mengali (1977), Falconer and Salz (1977), and Meyers and Franks (1980).

The Cramér–Rao lower bound on the variance of a parameter estimate is derived and evaluated in a number of standard texts on detection and estimation theory, such as Helstrom (1968) and Van Trees (1968). It is also described in several books on mathematical statistics, such as the book by Cramér (1946).

PROBLEMS

6-1 Prove the relation (6-1-7).

6-2 Sketch the equivalent realization of the binary PSK receiver in Fig. 6-1-1 that employs a matched filter instead of a correlator.

6-3 Suppose that the loop filter [see (6-2-14)] for a PLL has the transfer function

$$G(s) = \frac{1}{s + \sqrt{2}}$$

a Determine the closed-loop transfer function $H(s)$ and indicate if the loop is stable.

b Determine the damping factor and the natural frequency of the loop.

6-4 Consider the PLL for estimating the carrier phase of a signal in which the loop filter is specified as

$$G(s) = \frac{K}{1 + \tau_1 s}$$

a Determine the closed-loop transfer function $H(s)$ and its gain at $f = 0$.

b For what range of values of τ_1 and K is the loop stable?

6-5 The loop filter $G(s)$ in a PLL is implemented by the circuit shown in Fig. P6-5. Determine the system function $G(s)$ and express the time constants τ_1 and τ_2 in terms of the circuit parameters.

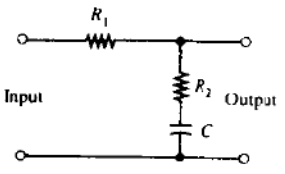


FIGURE P6-5

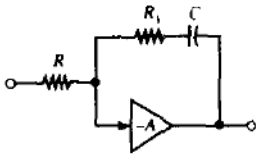


FIGURE P6-6

6-6 The loop filter $G(s)$ in a PLL is implemented with the active filter shown in Fig. P6-6. Determine the system function $G(s)$ and express the time constants τ_1 and τ_2 in terms of the circuit parameters.

6-7 Show that the early–late gate synchronizer illustrated in Fig. 6-3-5 is a close approximation to the timing recovery system illustrated in Fig. P6-7.

6-8 Based on a ML criterion, determine a carrier phase estimation method for binary on–off keying modulation.

6-9 In the transmission and reception of signals to and from moving vehicles, the transmitted signal frequency is shifted in direct proportion to the speed of the vehicle. The so-called *Doppler frequency shift* imparted to a signal that is received in a vehicle traveling at a velocity v relative to a (fixed) transmitter is given by the formula

$$f_D = \pm\frac{v}{\lambda}$$

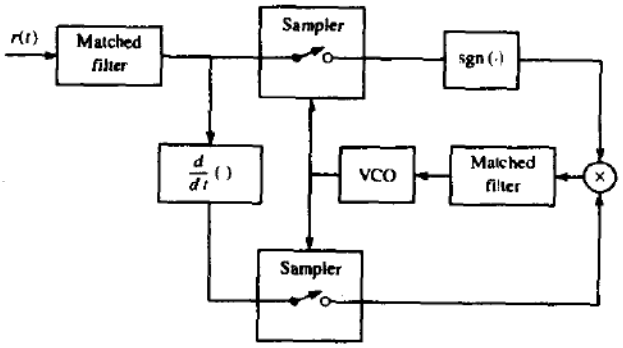
where λ is the wavelength, and the sign depends on the direction (moving toward or moving away) that the vehicle is traveling relative to the transmitter. Suppose that a vehicle is traveling at a speed of 100 km/h relative to a base station in a mobile cellular communication system. The signal is a narrowband signal transmitted at a carrier frequency of 1 GHz.

a Determine the Doppler frequency shift.

b What should be the bandwidth of a Doppler frequency tracking loop if the loop is designed to track Doppler frequency shifts for vehicles traveling at speeds up to 100 km/h?

c Suppose the transmitted signal bandwidth is 2 MHz centered at 1 GHz.

FIGURE P6-7



Determine the Doppler frequency spread between the upper and lower frequencies in the signal.

6-10 Show that the mean value of the ML estimate in (6-2-38) is ϕ, i.e., that the estimate is unbiased.

6-11 Determine the pdf of the ML phase estimate in (6-2-38).

6-12 Determine the ML phase estimate for offset QPSK.

6-13 A single-sideband PAM signal may be represented as

$$u_m(t) = A_m[g_I(t)\cos 2\pi f_c t - \hat{g}_I(t)\sin 2\pi f_c t]$$

where $\hat{g}_I(t)$ is the Hilbert transform of $g_I(t)$ and A_m is the amplitude level that conveys the information. Demonstrate mathematically that a Costas loop can be used to demodulate the SSB PAM signal.

6-14 A carrier component is transmitted on the quadrature carrier in a communication system that transmits information via binary PSK. Hence, the received signal has the form

$$r(t) = \pm\sqrt{2P_s}\cos(2\pi f_c + \phi) + \sqrt{2P_c}\sin(2\pi f_c + \phi) + n(t)$$

where ϕ is the carrier phase and $n(t)$ is AWGN. The unmodulated carrier component is used as a pilot signal at the receiver to estimate the carrier phase.

a Sketch a block diagram of the receiver, including the carrier phase estimator.

b Illustrate mathematically the operations involved in the estimation of the carrier phase ϕ.

c Express the probability of error for the detection of the binary PSK signal as a function of the total transmitted power $P_T = P_s + P_c$. What is the loss in performance due to the allocation of a portion of the transmitted power to the pilot signal? Evaluate the loss for $P_c/P_T = 0.1$.

6-15 Determine the signal and noise components at the input to a fourth-power ($M = 4$) PLL that is used to generate the carrier phase for demodulation of QPSK. By ignoring all noise components except those that are linear in the noise $n(t)$, determine the variance of the phase estimate at the output of the PLL.

6-16 The probability of error for binary PSK demodulation and detection when there is a carrier phase error ϕ_e is

$$P_2(\phi_e) = Q\left(\sqrt{\frac{2\mathcal{E}_b}{N_0}\cos^2\phi_e}\right)$$

Suppose that the phase error from the PLL is modeled as a zero-mean gaussian random variable with variance $\sigma_\phi^2 \ll \pi$. Determine the expression for the average probability of error (in integral form).

7

CHANNEL CAPACITY AND CODING

In Chapter 5, we considered the problem of digital modulation by means of $M = 2^k$ signal waveforms, where each waveform conveys k bits of information. We observed that some modulation methods provide better performance than others. In particular, we demonstrated that orthogonal signaling waveforms allow us to make the probability of error arbitrarily small by letting the number of waveforms $M \to \infty$, provided that the SNR per bit $\gamma_b \geq -1.6$ dB. Thus, we can operate at the capacity of the additive, white gaussian noise channel in the limit as the bandwidth expansion factor $B_e = W/R \to \infty$. This is a heavy price to pay, because B_e grows exponentially with the block length k. Such inefficient use of channel bandwidth is highly undesirable.

In this and the following chapter, we consider signal waveforms generated from either binary or nonbinary sequences. The resulting waveforms are generally characterized by a bandwidth expansion factor that grows only linearly with k. Consequently, coded waveforms offer the potential for greater bandwidth efficiency than orthogonal M-ary waveforms. We shall observe that, in general, coded waveforms offer performance advantages not only in power-limited applications where $R/W < 1$, but also in bandwidth-limited systems where $R/W > 1$.

We begin by establishing several channel models that will be used to evaluate the benefits of channel coding, and we shall introduce the concept of channel capacity for the various channel models. Then, we treat the subject of code design for efficient communications.

7-1 CHANNEL MODELS AND CHANNEL CAPACITY

In the model of a digital communication system described in Section 1-1, we recall that the transmitter building blocks consist of the discrete-input, discrete-output channel encoder followed by the modulator. The function of the discrete channel encoder is to introduce, in a controlled manner, some redundancy in the binary information sequence, which can be used at the receiver to overcome the effects of noise and interference encountered in the transmission of the signal through the channel. The encoding process generally involves taking k information bits at a time and mapping each k-bit sequence into a unique n-bit sequence, called a *code word*. The amount of redundancy introduced by the encoding of the data in this manner is measured by the ratio n/k. The reciprocal of this ratio, namely k/n, is called the *code rate*.

The binary sequence at the output of the channel encoder is fed to the modulator, which serves as the interface to the communication channel. As we have discussed, the modulator may simply map each binary digit into one of two possible waveforms, i.e., a 0 is mapped into $s_1(t)$ and a 1 is mapped into $s_2(t)$. Alternatively, the modulator may transmit q-bit blocks at a time by using $M = 2^q$ possible waveforms.

At the receiving end of the digital communication system, the demodulator processes the channel-corrupted waveform and reduces each waveform to a scalar or a vector that represents an estimate of the transmitted data symbol (binary or M-ary). The detector, which follows the demodulator, may decide on whether the transmitted bit is a 0 or a 1. In such a case, the detector has made a *hard decision*. If we view the decision process at the detector as a form of quantization, we observe that a hard decision corresponds to binary quantization of the demodulator output. More generally, we may consider a detector that quantizes to $Q > 2$ levels, i.e., a Q-ary detector. If M-ary signals are used then $Q \geq M$. In the extreme case when no quantization is performed, $Q = \infty$. In the case where $Q > M$, we say that the detector has made a *soft decision*.

The quantized output from the detector is then fed to the channel decoder, which exploits the available redundancy to correct for channel disturbances.

In the following sections, we describe three channel models that will be used to establish the maximum achievable bit rate for the channel.

7-1-1 Channel Models

In this section we describe channel models that will be useful in the design of codes. The simplest is the *binary symmetric channel* (BSC), which corresponds to the case with $M = 2$ and hard decisions at the detector.

Binary Symmetric Channel Let us consider an additive noise channel and let the modulator and the demodulator/detector be included as parts of the

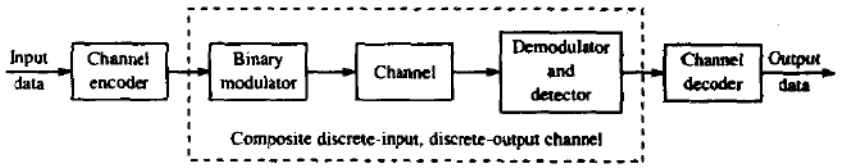


FIGURE 7-1-1 A composite discrete-input, discrete-output channel formed by including the modulator and the demodulator/detector as part of the channel.

channel. If the modulator employs binary waveforms and the detector makes hard decisions, then the composite channel, shown in Fig. 7-1-1, has a discrete-time binary input sequence and a discrete-time binary output sequence. Such a composite channel is characterized by the set $X = \{0, 1\}$ of possible inputs, the set of $Y = \{0, 1\}$ of possible outputs, and a set of conditional probabilities that relate the possible outputs to the possible inputs. If the channel noise and other disturbances cause statistically independent errors in the transmitted binary sequence with average probability p then

$$\begin{aligned} P(Y = 0 \mid X = 1) &= P(Y = 1 \mid X = 0) = p \\ P(Y = 1 \mid X = 1) &= P(Y = 0 \mid X = 0) = 1 - p \end{aligned} \tag{7-1-1}$$

Thus, we have reduced the cascade of the binary modulator, the waveform channel, and the binary demodulator and detector into an equivalent discrete-time channel which is represented by the diagram shown in Fig. 7-1-2. This binary-input, binary-output, symmetric channel is simply called a *binary symmetric channel* (BSC). Since each output bit from the channel depends only on the corresponding input bit, we say that the channel is memoryless.

Discrete Memoryless Channels The BSC is a special case of a more general discrete-input, discrete-output channel. Suppose that the output from the channel encoder are q-ary symbols, i.e., $X = \{x_0, x_1, \ldots, x_{q-1}\}$ and the output of the detector consists of Q-ary symbols, where $Q \geq M = 2^q$. If the

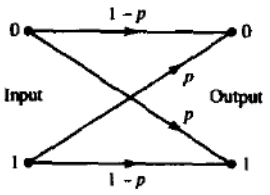


FIGURE 7-1-2 Binary symmetric channel.

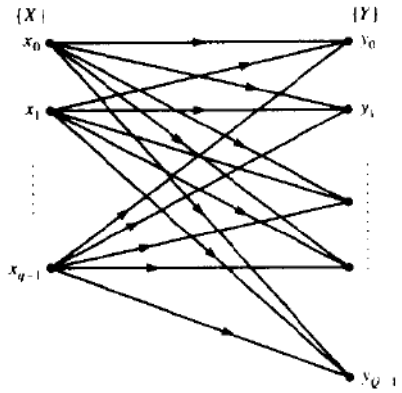


FIGURE 7-1-3 Discrete q-ary input, Q-ary output channel.

channel and the modulation are memoryless, then the input–output characteristics of the composite channel, shown in Fig. 7-1-1, are described by a set of qQ conditional probabilities

$$P(Y = y_i \mid X = x_j) \equiv P(y_i \mid x_j) \tag{7-1-2}$$

where $i = 0, 1, \ldots, Q-1$ and $j = 0, 1, \ldots, q-1$. Such a channel is called a *dicrete memoryless channel* (DMC), and its graphical representation is shown in Fig. 7-1-3. Hence, if the input to a DMC is a sequence of n symbols $u_1, u_2, \ldots, u_n$ selected from the alphabet X and the corresponding output is the sequence $v_1, v_2, \ldots, v_n$ of symbols from the alphabet Y, the joint conditional probability is

$$\begin{aligned} P(Y_1 = v_1, Y_2 = v_2, \ldots, Y_n = v_n \mid X = u_1, \ldots, X = u_n) \\ = \prod_{k=1}^{n} P(Y = v_k \mid X = u_k) \end{aligned} \tag{7-1-3}$$

This expression is simply a mathematical statement of the memoryless condition.

In general, the conditional probabilities $\{P(y_i \mid x_j)\}$ that characterize a DMC can be arranged in the matrix form $\mathbf{P} = [p_{ji}]$, where, by definition, $p_{ji} \equiv P(y_i \mid x_j)$. $\mathbf{P}$ is called the *probability transition matrix* for the channel.

Discrete-Input, Continuous-Output Channel Now, suppose that the input to the modulator comprises symbols selected from a finite and discrete input alphabet $X = \{x_0, x_1, \ldots, x_{q-1}\}$ and the output of the detector is unquantized ($Q = \infty$). Then, the input to the channel decoder can assume any value on the real line, i.e., $Y = \{-\infty, \infty\}$. This leads us to define a composite discrete-time

memoryless channel that is characterized by the discrete input X, the continuous output Y, and the set of conditional probability density functions

$$p(y \mid X = x_k), \quad k = 0, 1, \ldots, q-1$$

The most important channel of this type is the additive white gaussian noise channel (AWGN), for which

$$Y = X + G \tag{7-1-4}$$

where G is a zero-mean gaussian random variable with variance σ^2 and $X = x_k$, $k = 0, 1, \ldots, q-1$. For a given X, it follows that Y is gaussian with mean x_k and variance σ^2. That is,

$$p(y \mid X = x_k) = \frac{1}{\sqrt{2\pi}\sigma} e^{-(y-x_k)^2/2\sigma^2} \tag{7-1-5}$$

For any given input sequence, X_i, $i = 1, 2, \ldots, n$, there is a corresponding output sequence

$$Y_i = X_i + G_i, \quad i = 1, 2, \ldots, n \tag{7-1-6}$$

The condition that the channel is memoryless may be expressed as

$$p(y_1, y_2, \ldots, y_n \mid X_1 = u_1, X_2 = u_2, \ldots, X_n = u_n) = \prod_{i=1}^{n} p(y_i \mid X_i = u_i) \tag{7-1-7}$$

Waveform Channels We may separate the modulator and demodulator from the physical channel, and consider a channel model in which the inputs are waveforms and the outputs are waveforms. Let us assume that such a channel has a given bandwidth W, with ideal frequency response $C(f) = 1$ within the bandwidth W, and the signal at its output is corrupted by additive white gaussian noise. Suppose, that $x(t)$ is a band-limited input to such a channel and $y(t)$ is the corresponding output. Then,

$$y(t) = x(t) + n(t) \tag{7-1-8}$$

where $n(t)$ represents a sample function of the additive noise process. A suitable method for defining a set of probabilities that characterize the channel is to expand $x(t)$, $y(t)$, and $n(t)$ into a complete set of orthonormal functions. That is, we express $x(t)$, $y(t)$, and $n(t)$ in the form

$$\begin{aligned} y(t) &= \sum_i y_i f_i(t) \\ x(t) &= \sum_i x_i f_i(t) \\ n(t) &= \sum_i n_i f_i(t) \end{aligned} \tag{7-1-9}$$

where $\{y_i\}$, $\{x_i\}$, and $\{n_i\}$ are the sets of coefficients in the corresponding expansions, e.g.,

$$\begin{aligned} y_i &= \int_0^T y(t) f_i^*(t)\, dt \\ &= \int_0^T [x(t) + n(t)] f_i^*(t)\, dt \\ &= x_i + n_i \end{aligned} \tag{7-1-10}$$

The functions $\{f_i(t)\}$ form a complete orthonormal set over the interval $(0, T)$, i.e.,

$$\int_0^T f_i(t) f_j^*(t)\, dt = \delta_{ij} = \begin{cases} 1 & (i = j) \\ 0 & (i \neq j) \end{cases} \tag{7-1-11}$$

where δ_{ij} is the Kronecker delta function. Since the gaussian noise is white, any complete set of orthonormal functions may be used in the expansions (7-1-9).

We may now use the coefficients in the expansion for characterizing the channel. Since

$$y_i = x_i + n_i$$

where n_i is gaussian, it follows that

$$p(y_i \mid x_i) = \frac{1}{\sqrt{2\pi}\,\sigma_i} e^{-(y_i - x_i)^2/2\sigma_i^2}, \qquad i = 1, 2, \ldots \tag{7-1-12}$$

Since the functions $\{f_i(t)\}$ in the expansion are orthonormal, it follows that the $\{n_i\}$ are uncorrelated. Since they are gaussian, they are also statistically independent. Hence,

$$p(y_1, y_2, \ldots, y_N \mid x_1, x_2, \ldots, x_N) = \prod_{i=1}^{N} p(y_i \mid x_i) \tag{7-1-13}$$

for any N. In this manner, the waveform channel is reduced to an equvalent discrete-time channel characterized by the conditional pdf given in (7-1-12).

When the additive noise is white and gaussian with spectral density $\frac{1}{2}N_0$, the variances $\sigma_i^2 = \frac{1}{2}N_0$ for all i in (7-1-12). In this case, samples of $x(t)$ and $y(t)$ may be taken at the Nyquist rate of $2W$ samples/s, so that $x_i = x(i/2W)$ and $y_i = y(i/2W)$. Since the noise is white, the noise samples are statistically independent. Thus, (7-1-12) and (7-1-13) describe the statistics of the sampled signal. We note that in a time interval of length T, there are $N = 2WT$ samples. This parameter is used below in obtaining the capacity of the band-limited AWGN waveform channel.

The choice of which channel model to use at any one time depends on our objectives. If we are interested in the design and analysis of the performance

of the discrete channel encoder and decoder, it is appropriate to consider channel models in which the modulator and demodulator are a part of the composite channel. On the other hand, if our intent is to design and analyze the performance of the digital modulator and digital demodulator, we use a channel model for the waveform channel.

7-1-2 Channel Capacity

Now let us consider a DMC having an input alphabet $X = \{x_0, x_1, \ldots, x_{q-1}\}$, an output alphabet $Y = \{y_0, y_1, \ldots, y_{Q-1}\}$, and the set of transition probabilities $P(y_i \mid x_j)$ as defined in (7-1-2). Suppose that the symbol x_j is transmitted and the symbol y_i is received. The mutual information provided about the event $X = x_j$ by the occurrence of the event $Y = y_i$ is $\log [P(y_i \mid x_j)/P(y_i)]$, where

$$P(y_i) \equiv P(Y = y_i) = \sum_{k=0}^{q-1} P(x_k)P(y_i \mid x_k) \tag{7-1-14}$$

Hence, the average mutual information provided by the output Y about the input X is

$$I(X; Y) = \sum_{j=0}^{q-1} \sum_{i=0}^{Q-1} P(x_j)P(y_i \mid x_j) \log \frac{P(y_i \mid x_j)}{P(y_i)} \tag{7-1-15}$$

The channel characteristics determine the transition probabilities $P(y_i \mid x_j)$, but the probabilities of the input symbols are under the control of the discrete channel encoder. The value of $I(X; Y)$ maximized over the set of input symbol probabilities $P(x_j)$ is a quantity that depends only on the characteristics of the DMC through the conditional probabilities $P(y_i \mid x_j)$. This quantity is called the *capacity* of the channel and is denoted by C. That is, the capacity of a DMC is defined as

$$\begin{aligned} C &= \max_{P(x_j)} I(X; Y) \\ &= \max_{P(x_j)} \sum_{j=0}^{q-1} \sum_{i=0}^{Q-1} P(x_j)P(y_i \mid x_j) \log \frac{P(y_i \mid x_j)}{P(y_i)} \end{aligned} \tag{7-1-16}$$

The maximization of $I(X; Y)$ is performed under the constraints that

$$P(x_j) \geq 0$$

$$\sum_{j=0}^{q-1} P(x_j) = 1$$

The units of C are bits per input symbol into the channel (bits/channel use)

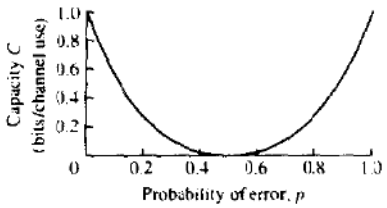


FIGURE 7-1-4 The capacity of a BSC as a function of the error probability p.

when the logarithm is base 2, and nats/input symbol when the natural logarithm (base e) is used. If a symbol enters the channel every τ_s seconds, the channel capacity in bits/s or nats/s is C/τ_s.

Example 7-1-1

For the BSC with transition probabilities

$$P(0 \mid 1) = P(1 \mid 0) = p$$

the average mutual information is maximized when the input probabilities $P(0) = P(1) = \frac{1}{2}$. Thus, the capacity of the BSC is

$$C = p \log 2p + (1 - p) \log 2(1 - p) = 1 - H(p) \qquad (7\text{-}1\text{-}17)$$

where $H(p)$ is the binary entropy function. A plot of C versus p is illustrated in Fig. 7-1-4. Note that for $p = 0$, the capacity is 1 bit/channel use. On the other hand, for $p = \frac{1}{2}$, the mutual information between input and output is zero. Hence, the channel capacity is zero. For $\frac{1}{2} < p \leq 1$, we may reverse the position of 0 and 1 at the output of the BSC, so that C becomes symmetric with respect to the point $p = \frac{1}{2}$. In our treatment of binary modulation and demodulation given in Chapter 5, we showed that p is a monotonic function of the signal-to-noise ratio (SNR) as illustrated in Fig. 7-1-5(a). Consequently when C is plotted as a function of the SNR, it increases monotonically as the SNR increases. This characteristic behavior of C versus SNR is illustrated in Fig. 7-1-5(b).

Next let us consider the discrete-time AWGN memoryless channel described by the transition probability density functions defined by (7-1-5). The

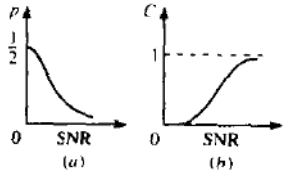


FIGURE 7-1-5 General behavior of error probability and channel capacity as a function of SNR.

average mutual information between the discrete input $X = \{x_0, x_1, \ldots, x_{q-1}\}$ and the output $Y = \{-\infty, \infty\}$ is given by the capacity of this channel in bits/channel use is

$$C = \max_{P(x_i)} \sum_{i=0}^{q-1} \int_{-\infty}^{\infty} p(y \mid x_i) P(x_i) \log_2 \frac{p(y \mid x_i)}{p(y)} dy \tag{7-1-18}$$

where

$$p(y) = \sum_{k=0}^{q-1} p(y \mid x_k) P(x_k) \tag{7-1-19}$$

Example 7-1-2

Let us consider a binary-input AWGN memoryless channel with possible inputs $X = A$ and $X = -A$. The average mutual information $I(X; Y)$ is maximized when the input probabilities are $P(X = A) = P(X = -A) = \frac{1}{2}$. Hence, the capacity of this channel in bits/channel use is

$$\begin{aligned} C = {} & \tfrac{1}{2} \int_{-\infty}^{\infty} p(y \mid A) \log_2 \frac{p(y \mid A)}{p(y)} dy \\ & + \tfrac{1}{2} \int_{-\infty}^{\infty} p(y \mid -A) \log_2 \frac{p(y \mid -A)}{p(y)} dy \end{aligned} \tag{7-1-20}$$

Figure 7-1-6 illustrates C as a function of the ratio $A^2/2\sigma^2$. Note that C increases monotonically from 0 to 1 bit/symbol as this ratio increases.

It is interesting to note that in the two channel models described above, the choice of equally probable input symbols maximizes the average mutual information. Thus, the capacity of the channel is obtained when the input symbols are equally probable. This is not always the solution for the capacity formulas given in (7-1-16) and (7-1-18), however. Nothing can be said in general about the input probability assignment that maximizes the average mutual information. However, in the two channel models considered above,

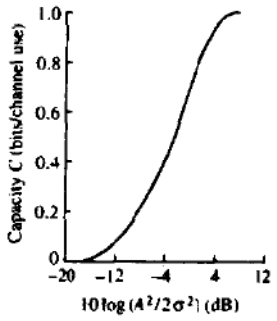


FIGURE 7-1-6 Channel capacity as a function of $A^2/2\sigma^2$ for binary-input AWGN memoryless channel.

the channel transition probabilities exhibit a form of symmetry that results in the maximum of $I(X; Y)$ being obtained when the input symbols are equally probable. The symmetry condition can be expressed in terms of the elements of the probability transition matrix $\mathbf{P}$ of the channel. When each row of this matrix is a permutation of any other row and each column is a permutation of any other column, the probability transition matrix is symmetric and input symbols with equal probability maximize $I(X; Y)$.

In general, necessary and sufficient conditions for the set of input probabilities $\{P(x_j)\}$ to maximize $I(X; Y)$ and, thus, to achieve capacity on a DMC are that (Problem 7-1)

$$\begin{aligned} I(x_j; Y) &= C \quad \text{for all } j \text{ with } P(x_j) > 0 \\ I(x_j; Y) &\leq C \quad \text{for all } j \text{ with } P(x_j) = 0 \end{aligned} \tag{7-1-21}$$

where C is the capacity of the channel and

$$I(x_j; Y) = \sum_{i=0}^{Q-1} P(y_i \mid x_j) \log \frac{P(y_i \mid x_j)}{P(y_i)} \tag{7-1-22}$$

Usually, it is relatively easy to check if the equally probable set of input symbols satisfy the conditions (7-1-21). If they do not, then one must determine the set of unequal probabilities $\{P(x_j)\}$ that satisfy (7-1-21).

Now let us consider a band-limited waveform channel with additive white gaussian noise. Formally, the capacity of the channel per unit time has been defined by Shannon (1948b) as

$$C = \lim_{T \to \infty} \max_{p(x)} \frac{1}{T} I(X; Y) \tag{7-1-23}$$

where the average mutual information $I(X; Y)$ is given in (3-2-17). Alternatively, we may use the samples or the coefficients $\{y_i\}$, $\{x_i\}$, and $\{n_i\}$ in the series expansions of $y(t)$, $x(t)$, and $n(t)$, respectively, to determine the average mutual information between $\mathbf{x}_N = [x_1\ x_2\ \dots\ x_N]$ and $\mathbf{y}_N = [y_1\ y_2\ \dots\ y_N]$, where $N = 2WT$, $y_i = x_i + n_i$, and $p(y_i \mid x_i)$ is given by (7-1-12). The average mutual information between $\mathbf{x}_N$ and $\mathbf{y}_N$ for the AWGN channel is

$$\begin{aligned} I(\mathbf{X}_N; \mathbf{Y}_N) &= \int_{\mathbf{x}_N} \cdots \int \int_{\mathbf{y}_N} \cdots \int p(\mathbf{y}_N \mid \mathbf{x}_N) p(\mathbf{x}_N) \log \frac{p(\mathbf{y}_N \mid \mathbf{x}_N)}{p(\mathbf{y}_N)} \, d\mathbf{x}_N \, d\mathbf{y}_N \\ &= \sum_{i=1}^{N} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} p(y_i \mid x_i) p(x_i) \log \frac{p(y_i \mid x_i)}{p(y_i)} \, dy_i \, dx_i \end{aligned} \tag{7-1-24}$$

where

$$p(y_i \mid x_i) = \frac{1}{\sqrt{\pi N_0}} e^{-(y_i - x_i)^2 / N_0} \tag{7-1-25}$$

The maximum of $I(X; Y)$ over the input pdfs $p(x_i)$ is obtained when the $\{x_i\}$ are statistically independent zero-mean gaussian random variables, i.e.,

$$p(x_i) = \frac{1}{\sqrt{2\pi}\,\sigma_x} e^{-x_i^2/2\sigma_x^2} \tag{7-1-26}$$

where σ_x^2 is the variance of each x_i. Then, it follows from (7-1-24) that

$$\begin{aligned} \max_{p(x)} I(\mathbf{X}_N; \mathbf{Y}_N) &= \sum_{i=1}^{N} \tfrac{1}{2} \log\left(1 + \frac{2\sigma_x^2}{N_0}\right) \\ &= \tfrac{1}{2} N \log\left(1 + \frac{2\sigma_x^2}{N_0}\right) \\ &= WT \log\left(1 + \frac{2\sigma_x^2}{N_0}\right) \end{aligned} \tag{7-1-27}$$

Suppose that we put a constraint on the average power in $x(t)$. That is,

$$\begin{aligned} P_{\text{av}} &= \frac{1}{T}\int_0^T E[x^2(t)]\,dt \\ &= \frac{1}{T}\sum_{i=1}^{N} E(x_i^2) \\ &= \frac{N\sigma_x^2}{T} \end{aligned} \tag{7-1-28}$$

Hence,

$$\begin{aligned} \sigma_x^2 &= \frac{TP_{\text{av}}}{N} \\ &= \frac{P_{\text{av}}}{2W} \end{aligned} \tag{7-1-29}$$

Substitution of this result into (7-1-27) for σ_x^2 yields

$$\max_{p(x)} I(\mathbf{X}_N; \mathbf{Y}_N) = WT \log\left(1 + \frac{P_{\text{av}}}{WN_0}\right) \tag{7-1-30}$$

Finally, the channel capacity per unit time is obtained by dividing the result in (7-1-30) by T. Thus

$$C = W \log\left(1 + \frac{P_{\text{av}}}{WN_0}\right) \tag{7-1-31}$$

This is the basic formula for the capacity of the band-limited AWGN

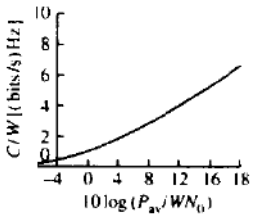


FIGURE 7-1-7 Normalized channel capacity as a function of SNR for band-limited AWGN channel.

waveform channel with a band-limited and average power-limited input. It was originally derived by Shannon (1948b).

A plot of the capacity in bits/s normalized by the bandwidth W is plotted in Fig. 7-1-7 as a function of the ratio of signal power P_{av} to noise power WN_0. Note that the capacity increases monotonically with increasing SNR. Thus, for a fixed bandwidth, the capacity of the waveform channel increases with an increase in the transmitted signal power. On the other hand, if P_{av} is fixed, the capacity can be increased by increasing the bandwidth W. Figure 7-1-8 illustrates a graph of C versus W. Note that as W approaches infinity, the capacity of the channel approaches the asymptotic value

$$C_\infty = \frac{P_{av}}{N_0} \log_2 e = \frac{P_{av}}{N_0 \ln 2} \text{ bits/s} \tag{7-1-32}$$

It is instructive to express the normalized channel capacity C/W as a function of the SNR per bit. Since P_{av} represents the average transmitted power and C is the ratio in bits/s, it follows that

$$P_{av} = C\mathscr{E}_b \tag{7-1-33}$$

where $\mathscr{E}_b$ is the energy per bit. Hence, (7-1-31) may be expressed as

$$\frac{C}{W} = \log_2 \left(1 + \frac{C}{W} \frac{\mathscr{E}_b}{N_0}\right) \tag{7-1-34}$$

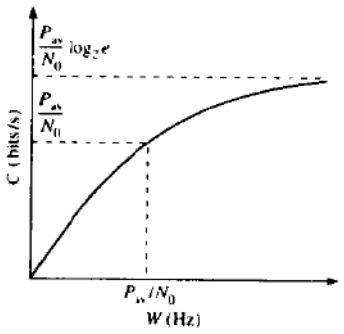


FIGURE 7-1-8 Channel capacity as a function of bandwidth with a fixed transmitted average power.

Consequently,

$$\frac{\mathscr{E}_b}{N_0} = \frac{2^{C/W} - 1}{C/W} \tag{7-1-35}$$

When $C/W = 1$, $\mathscr{E}_b/N_0 = 1$ (0 dB). As $C/W \to \infty$,

$$\begin{aligned} \frac{\mathscr{E}_b}{N_0} &\approx \frac{2^{C/W}}{C/W} \\ &\approx \exp\left(\frac{C}{W}\ln 2 - \ln\frac{C}{W}\right) \end{aligned} \tag{7-1-36}$$

Thus, $\mathscr{E}_b/N_0$ increases exponentially as $C/W \to \infty$. On the other hand, as $C/W \to 0$,

$$\frac{\mathscr{E}_b}{N_0} = \lim_{C/W \to 0} \frac{2^{C/W} - 1}{C/W} = \ln 2 \tag{7-1-37}$$

which is -1.6 dB. A plot of C/W versus $\mathscr{E}_b/N_0$ is shown in Fig. 5-2-17.

Thus, we have derived the channel capacities of three important channel models that are considered in this book. The first is the discrete-input, discrete-output channel, of which the BSC is a special case. The second is a discrete-input, continuous-output memoryless additive white gaussian noise channel. From these two channel models, we can obtain benchmarks for the coded performance with hard- and soft-decision decoding in digital communications systems.

The third channel model focuses on the capacity in bits/s of a waveform channel. In this case, we assumed that we have a bandwidth limitation on the channel, an additive gaussian noise that corrupts the signal, and an average power constraint at the transmitter. Under these conditions, we derived the result given in (7-1-31).

The major significance of the channel capacity formulas given above is that they serve as upper limits on the transmission rate for reliable communication over a noisy channel. The fundamental rate that the channel capacity plays is given by the *noisy channel coding theorem* due to Shannon (1948a).

Noisy Channel Coding Theorem

There exist channel codes (and decoders) that make it possible to achieve reliable communication, with as small an error probability as desired, if the transmission rate $R < C$, where C is the channel capacity. If $R > C$, it is not possible to make the probability of error tend toward zero with any code.

In the following section, we explore the benefits of coding for the additive

noise channel models described above, and use the channel capacity as the benchmark for accessing code performance.

7-1-3 Achieving Channel Capacity with Orthogonal Signals

In Section 5-2, we used a simple union bound to show that, for orthogonal signals, the probability of error can be made as small as desired by increasing the number M of waveforms, provided that $\mathscr{E}_b/N_0 > 2 \ln 2$. We indicated that the simple union bound does not produce the smallest lower bound on the SNR per bit. The problem is that the upper bound used on $Q(x)$ is very loose for small x.

An alternative approach is to use two different upper bounds for $Q(x)$, depending on the value of x. Beginning with (5-2-21), we observe that

$$1 - [1 - Q(y)]^{M-1} \leq (M-1)Q(y) < Me^{-y^2/2} \tag{7-1-38}$$

This is just the union bound, which is tight when y is large, i.e., for $y > y_0$, where y_0 depends on M. When y is small, the union bound exceeds unity for large M. Since

$$1 - [1 - Q(y)]^{M-1} \leq 1 \tag{7-1-39}$$

for all y, we may use this bound for $y < y_0$ because it is tighter than the union bound. Thus (5-2-21) may be upper-bounded as

$$P_M < \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{y_0} e^{-(y-\sqrt{2\gamma})^2/2}\, dy + \frac{M}{\sqrt{2\pi}} \int_{y_0}^{\infty} e^{-y^2/2} e^{-(y-\sqrt{2\gamma})^2/2}\, dy \tag{7-1-40}$$

The value of y_0 that minmizes this upper bound is found by differentiating the right-hand side of (7-1-40) and setting the derivative equal to zero. It is easily verified that the solution is

$$e^{y_0^2/2} = M \tag{7-1-41}$$

or, equivalently,

$$\begin{aligned} y_0 &= \sqrt{2 \ln M} = \sqrt{2 \ln 2 \log_2 M} \\ &= \sqrt{2k \ln 2} \end{aligned} \tag{7-1-42}$$

Having determined y_0, let us now compute simple exponential upper bounds for the integrals in (7-1-40). For the first integral, we have

$$\begin{aligned} \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{y_0} e^{-(y-\sqrt{2\gamma})^2/2}\, dy &= \frac{1}{\sqrt{\pi}} \int_{-\infty}^{-(\sqrt{2\gamma}-y_0)/\sqrt{2}} e^{-x^2}\, dx \\ &= Q(\sqrt{2\gamma} - y_0), \qquad y_0 \leq \sqrt{2\gamma} \\ &< e^{-(\sqrt{2\gamma}-y_0)^2/2}, \qquad y_0 \leq \sqrt{2\gamma} \end{aligned} \tag{7-1-43}$$

The second integral is upper-bounded as follows:

$$\frac{M}{\sqrt{2\pi}}\int_{y_0}^{\infty} e^{-y^2/2}e^{-(y-\sqrt{2\gamma})^2/2}\,dy = \frac{M}{\sqrt{2\pi}}e^{-\gamma/2}\int_{y_0-\sqrt{\gamma/2}}^{\infty} e^{-x^2}\,dx$$

$$< \begin{cases} Me^{-\gamma/2} & (y_0 \le \sqrt{\frac{1}{2}\gamma}) \\ Me^{-\gamma/2}e^{-(y_0-\sqrt{\gamma/2})^2} & (y_0 \ge \sqrt{\frac{1}{2}\gamma}) \end{cases} \quad (7\text{-}1\text{-}44)$$

Combining the bounds for the two integrals and substituting $e^{y_0^2/2}$ for M, we obtain

$$P_M < \begin{cases} e^{-(\sqrt{2\gamma}-y_0)^2/2} + e^{(y_0^2-\gamma)/2} & (0 \le y_0 \le \sqrt{\frac{1}{2}\gamma}) \\ e^{-(\sqrt{2\gamma}-y_0)^2/2} + e^{(y_0^2-\gamma)/2}e^{-(y_0-\sqrt{\gamma/2})^2} & (\sqrt{\frac{1}{2}\gamma} \le y_0 \le \sqrt{2\gamma}) \end{cases} \quad (7\text{-}1\text{-}45)$$

In the range $0 \le y_0 \le \sqrt{\frac{1}{2}\gamma}$, the bound may be expressed as

$$P_M < e^{(y_0^2-\gamma)/2}(1 + e^{-(y_0-\sqrt{\gamma/2})^2})$$

$$< 2e^{(y_0^2-\gamma)/2}, \quad 0 \le y_0 \le \sqrt{\tfrac{1}{2}\gamma} \quad (7\text{-}1\text{-}46)$$

In the range $\sqrt{\frac{1}{2}\gamma} \le y_0 \le \sqrt{2\gamma}$, the two terms in (7-1-45) are identical. Hence,

$$P_M < 2e^{-(\sqrt{2\gamma}-y_0)^2/2}, \quad \sqrt{\tfrac{1}{2}\gamma} \le y_0 \le \sqrt{2\gamma} \quad (7\text{-}1\text{-}47)$$

Now we substitute for y_0 and γ. Since $y_0 = \sqrt{2 \ln M} = \sqrt{2k \ln 2}$ and $\gamma = k\gamma_b$, the bounds in (7-1-46) and (7-1-47) may be expresed as

$$P_M < \begin{cases} 2e^{-k(\gamma_b - 2\ln 2)/2} & (\ln M \le \frac{1}{4}\gamma) \\ 2e^{-k(\sqrt{\gamma_b}-\sqrt{\ln 2})^2} & (\frac{1}{4}\gamma \le \ln M \le \gamma) \end{cases} \quad (7\text{-}1\text{-}48)$$

The first upper bound coincides with the union bound presented earlier, but it is loose for large values of M. The second upper bound is better for large values of M. We note that $P_M \to 0$ as $k \to \infty$ ($M \to \infty$) provided that $\gamma_b > \ln 2$. But, $\ln 2$ is the limiting value of the SNR per bit required for reliable transmission when signaling at a rate equal to the capacity of the infinite-bandwidth AWGN channel as shown in Section 7-1-2. In fact, when the substitutions

$$y_0 = \sqrt{2k \ln 2} = \sqrt{2RT \ln 2}$$

$$\gamma = \frac{TP_{av}}{N_0} = TC_\infty \ln 2 \quad (7\text{-}1\text{-}49)$$

are made into the two upper bounds given in (7-1-46) and (7-1-47), where $C_\infty = P_{av}/(N_0 \ln 2)$ is the capacity of the infinite-bandwidth AWGN channel, the result is

$$P_M < \begin{cases} 2 \cdot 2^{-T(\frac{1}{4}C_\infty - R)} & (0 \le R \le \frac{1}{4}C_\infty) \\ 2 \cdot 2^{-T(\sqrt{C_\infty}-\sqrt{R})^2} & (\frac{1}{4}C_\infty \le R \le C_\infty) \end{cases} \quad (7\text{-}1\text{-}50)$$

Thus we have expressed the bounds in terms of C_∞ and the bit rate in the channel. The first upper bound is appropriate for rates below $\frac{1}{4}C_\infty$, while the second is tighter than the first for rates between $\frac{1}{4}C_\infty$ and C_∞. Clearly, the probability of error can be made arbitrarily small by making $T \to \infty$ ($M \to \infty$ for fixed R), provided that $R < C_\infty = P_{av}/(N_0 \ln 2)$. Furthermore, we observe that the set of orthogonal waveforms achieves the channel capacity bound as $M \to \infty$, when the rate $R < C_\infty$.

7-1-4 Channel Reliability Functions

The exponential bounds on the error probability for M-ary orthogonal signals on an infinite-bandwidth AWGN channel given by (7-1-50) may be expressed as

$$P_M < 2 \cdot 2^{-TE(R)} \tag{7-1-51}$$

The exponential factor

$$E(R) = \begin{cases} \frac{1}{2}C_\infty - R & (0 \le R \le \frac{1}{4}C_\infty) \\ (\sqrt{C_\infty} - \sqrt{R})^2 & (\frac{1}{4}C_\infty \le R \le C_\infty) \end{cases} \tag{7-1-52}$$

in (7-1-51) is called the *channel reliability function* for the infinite-bandwidth AWGN channel. A plot of $E(R)/C_\infty$ is shown in Fig. 7-1-9. Also shown is the exponential factor for the union bound on P_M, given by (5-2-27), which may be expressed as

$$P_M \le \tfrac{1}{2} \cdot 2^{-T(\frac{1}{2}C_\infty - R)}, \qquad 0 \le R \le \tfrac{1}{2}C_\infty \tag{7-1-53}$$

Clearly, the exponential factor in (7-1-53) is not as tight as $E(R)$, due to the looseness of the union bound.

The bound given by (7-1-51) and (7-1-52) has been shown by Gallager (1965) to be *exponentially tight*. This means that there does not exist another reliability function, say $E_1(R)$, satisfying the condition $E_1(R) > E(R)$ for any R. Consequently, the error probability is bounded from above and below as

$$K_l 2^{-TE(R)} \le P_e \le K_u 2^{-TE(R)} \tag{7-1-54}$$

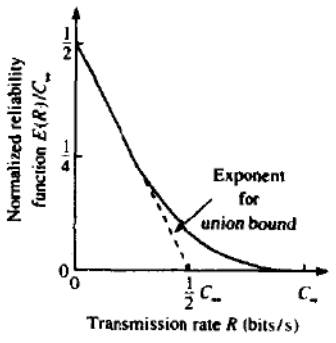


FIGURE 7-1-9 Channel reliability function for the infinite-bandwidth AWGN channel.

where the constants have only a weak dependence on T, i.e., they vary slowly with T.

Since orthogonal signals provide essentially the same performance as the optimum simplex signals for large M, the lower bound in (7-1-54) applies for any signal set. Hence, the reliability function $E(R)$ given by (7-1-52) determines the exponential characteristics of the error probability for digital signaling over the infinite-bandwidth AWGN channel.

Although the error probability can be made arbitrarily small by increasing the number of either orthogonal, biorthogonal, or simplex signals, with $R < C_\infty$, for a relatively modest number of signals, there is a large gap between the actual performance and the best achievable performance given by the channel capacity formula. For example, from Fig. 5-2-17, we observe that a set of $M = 16$ orthogonal signals detected coherently requires a SNR per bit of approximately 7.5 dB to achieve a bit error rate of $P_e = 10^{-5}$. In contrast, the channel capacity formula indicates that for a $C/W = 0.5$, reliable transmission is possible with a SNR of -0.8 dB. This represents a rather large difference of 8.3 dB/bit and serves as a motivation for searching for more efficient signaling waveforms. In this chapter and in Chapter 8, we demonstrate that coded waveforms can reduce this gap considerably.

Similar gaps in performance also exist in the bandwidth-limited region of Fig. 5-2-17, where $R/W > 1$. In this region, however, we must be more clever in how we use coding to improve performance, because we cannot expand the bandwidth as in the power-limited region. The use of coding techniques for bandwidth-efficient communication is also treated in Chapter 8.

7-2 RANDOM SELECTION OF CODES

The design of coded modulation for efficient transmission of information may be divided into two basic approaches. One is the algebraic approach, which is primarily concerned with the design of coding and decoding techniques for specific classes of codes, such as cyclic block codes and convolutional codes. The second is the probabilistic approach, which is concerned with the analysis of the performance of a general class of coded signals. This approach yields bounds on the probability of error that can be attained for communication over a channel having some specified characteristic.

In this section, we adopt the probabilistic approach to coded modulation. The algebraic approach, based on block codes and on convolutional codes, is treated in Chapter 8.

7-2-1 Random Coding Based on M-ary Binary Coded Signals

Let us consider a set of M coded signal waveforms constructed from a set of n-dimensional binary code words of the form

$$\mathbf{C}_i = [c_{i1} c_{i2} \ldots c_{in}], \quad i = 1, 2, \ldots, M \tag{7-2-1}$$

where $c_{ij} = 0$ or 1. Each bit in the code word is mapped into a binary PSK waveform, so that the signal waveform corresponding to the code word $\mathbf{C}_i$ may be expressed as

$$s_i(t) = \sum_{j=1}^{n} s_{ij} f_j(t), \qquad i = 1, 2, \ldots, M \tag{7-2-2}$$

where

$$s_{ij} = \begin{cases} \sqrt{\mathscr{E}_c} & \text{when } c_{ij} = 1 \\ -\sqrt{\mathscr{E}_c} & \text{when } c_{ij} = 0 \end{cases} \tag{7-2-3}$$

and $\mathscr{E}_c$ is the energy per code bit. Thus, the waveforms $s_i(t)$ are equivalent to the n-dimensional vectors

$$\mathbf{s}_i = [s_{i1} \quad s_{i2} \quad \ldots \quad s_{in}], \qquad i = 1, 2, \ldots, M \tag{7-2-4}$$

which correspond to the vertices of a hypercube in n-dimensional space.

Now, suppose that the information rate into the encoder is R bits/s and we encode blocks of k bits at a time into one of the M waveforms. Hence, $k = RT$ and $M = 2^k = 2^{RT}$ signals are required. It is convenient to define a parameter D as

$$D = \frac{n}{T} \text{ dimensions/s} \tag{7-2-5}$$

Thus, $n = DT$ is the dimensionality of the signal space.

The hypercube has $2^n = 2^{DT}$ vertices, of which $M = 2^{RT}$ may be used to transmit the information. If we impose the condition that $D > R$, the fraction of the vertices that we use as signal points is

$$F = \frac{2^k}{2^n} = \frac{2^{RT}}{2^{DT}} = 2^{-(D-R)T} \tag{7-2-6}$$

Clearly, if $D > R$, we have $F \to 0$ as $T \to \infty$.

The question that we wish to pose is the following. Can we choose a subset $M = 2^{RT}$ vertices out of the $2^n = 2^{DT}$ available vertices such that the probability of error $P \to 0$ as $T \to \infty$ or, equivalently, as $n \to \infty$? Since the fraction F of vertices used approaches zero as $T \to \infty$, it should be possible to select M signal waveforms having a minimum distance that increases as $T \to \infty$ and, thus, $P_e \to 0$.

Instead of attempting to find a single set of M coded waveforms for which we compute the error probability, let us consider the ensemble of $(2^n)^M$ distinct ways in which we can select M vertices from the 2^n available vertices of the hypercube. Associated with each of the 2^{nM} selections, there is a communication system, consisting of a modulator, a channel, and a demodulator, that is optimum for the selected set of M waveforms. Thus, there are 2^{nM}

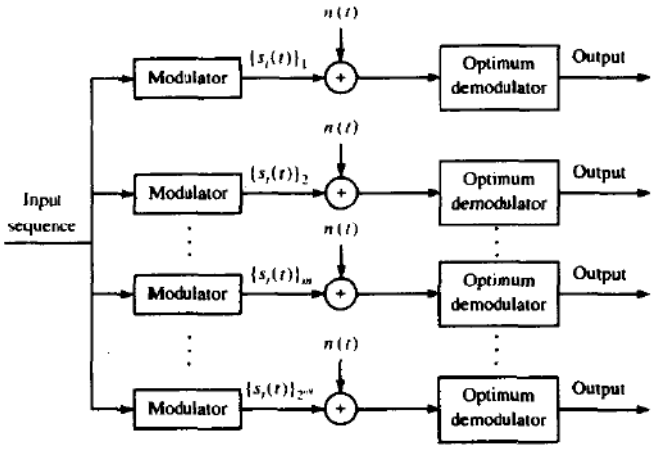


FIGURE 7-2-1 An ensemble at 2^{nM} communications system. Each system employs a different set of M signals from the set of 2^{nM} possible choices.

communication systems, one for each choice of the M coded waveforms, as illustrated in Fig. 7-2-1. Each communication system is characterized by its probability of error.

Suppose that our choice of M coded waveforms is based on random selection from the set of 2^{nM} possible sets of codes. Thus, the random selection of the mth code, denoted by $\{\mathbf{s}_i\}_m$, occurs with probability

$$P(\{\mathbf{s}_i\}_m) = 2^{-nM} \tag{7-2-7}$$

and the corresponding conditional probability of error for this choice of coded signals is $P_e(\{\mathbf{s}_i\}_m)$. Then, the average probability of error over the ensemble of codes is

$$\begin{aligned} \bar{P}_e &= \sum_{m=1}^{2^{nM}} P_e(\{\mathbf{s}_i\}_m) P(\{\mathbf{s}_i\}_m) \\ &= 2^{-nM} \sum_{m=1}^{2^{nM}} P_e(\{\mathbf{s}_i\}_m) \end{aligned} \tag{7-2-8}$$

where the overbar on P_e denotes an average over the ensemble of codes.

It is clear that some choices of codes will result in large probability of error. For example, the code that assigns all M k-bit sequences to the same vertex of the hypercube will result in a large probability of error. In such a case, $P_e(\{\mathbf{s}_i\}_m) > \bar{P}_e$. However, there will also be choices of codes for which $P_e(\{\mathbf{s}_i\}_m) < \bar{P}_e$. Consequently, if we obtain an upper bound on $\bar{P}_e$, this bound will also hold for those codes for which $P_e(\{\mathbf{s}_i\}_m) < \bar{P}_e$. Furthermore, if $\bar{P}_e \to 0$ as $T \to \infty$ then we conclude that, for these codes, $P(\{\mathbf{s}_i\}_m) \to 0$ as $T \to \infty$.

In order to determine an upper bound on $\bar{P}_e$, we consider the transmission

of a k-bit message $\mathbf{X}_k \equiv [x_1 x_2 x_3 \ldots x_k]$, where $x_j = 0$ or 1 for $j = 1, 2, \ldots, k$. The conditional probability of error averaged over the ensemble of codes is

$$\overline{P_e(\mathbf{X}_k)} = \sum_{\text{all codes}} P_e(\mathbf{X}_k, \{\mathbf{s}_i\}_m) P(\{\mathbf{s}_i\}_m) \tag{7-2-9}$$

where $P_e(\mathbf{X}_k, \{\mathbf{s}_i\}_m)$ is the conditional probability of error for a given k-bit message $\mathbf{X}_k$, which is transmitted by use of the code $\{\mathbf{s}_i\}_m$. For the mth code, the probability of error $P_e(\mathbf{X}_k, \{\mathbf{s}_i\}_m)$ is upper-bounded as

$$P_e(\mathbf{X}_k, \{\mathbf{s}_i\}_m) \leq \sum_{\substack{l=1 \\ l \neq k}}^{M} P_{2m}(\mathbf{s}_l, \mathbf{s}_k) \tag{7-2-10}$$

where $P_{2m}(\mathbf{s}_l, \mathbf{s}_k)$ is the probability of error for a binary communication system that employs the signal vectors $\mathbf{s}_l$ and $\mathbf{s}_k$ to communicate one of two equally likely k-bit messages. Hence,

$$\overline{P_e(\mathbf{X}_k)} \leq \sum_{\text{all codes}} P_e(\{\mathbf{s}_i\}_m) \sum_{\substack{l=1 \\ l \neq k}}^{M} P_{2m}(\mathbf{s}_l, \mathbf{s}_k) \tag{7-2-11}$$

If we interchange the order of the summations in (7-2-11) we obtain

$$\begin{aligned} \overline{P_e(\mathbf{X}_k)} &\leq \sum_{\substack{l=1 \\ l \neq k}}^{M} \left[\sum_{\text{all codes}} P_e(\{\mathbf{s}_i\}_m) P_{2m}(\mathbf{s}_l, \mathbf{s}_k) \right] \\ &\leq \sum_{\substack{l=1 \\ l \neq k}}^{M} \overline{P_2(\mathbf{s}_l, \mathbf{s}_k)} \end{aligned} \tag{7-2-12}$$

where $\overline{P_2(\mathbf{s}_l, \mathbf{s}_k)}$ represents the ensemble average of $P_{2m}(\mathbf{s}_l, \mathbf{s}_k)$ over the 2^{nM} codes or the 2^{nM} communication systems.

For the additive white gaussian noise channel, the binary error probability $P_{2m}(\mathbf{s}_l, \mathbf{s}_k)$ is

$$P_{2m}(\mathbf{s}_l, \mathbf{s}_k) = Q\left(\sqrt{\frac{d_{lk}^2}{2N_0}} \right) \tag{7-2-13}$$

where $d_{lk}^2 = |\mathbf{s}_l - \mathbf{s}_k|^2$. If $\mathbf{s}_l$ and $\mathbf{s}_k$ differ in d coordinates,

$$d_{lk}^2 = |\mathbf{s}_l - \mathbf{s}_k|^2 = \sum_{j=1}^{n} (s_{lj} - s_{kj})^2 = d(2\sqrt{\mathscr{E}_c})^2 = 4d\mathscr{E}_c \tag{7-2-14}$$

Hence,

$$P_{2m}(\mathbf{s}_l, \mathbf{s}_k) = Q\left(\sqrt{\frac{2d\mathscr{E}_c}{N_0}} \right) \tag{7-2-15}$$

Now, we can average $P_{2m}(\mathbf{s}_l, \mathbf{s}_k)$ over the ensemble of codes. Since all the codes are equally probable, the signal vector $\mathbf{s}_l$ is equally likely to be any of the 2^n possible vertices of the hypercube and it is statistically independent

of the signal vector $\mathbf{s}_k$. Therefore, $P(s_{li} = s_{ki}) = \frac{1}{2}$ and $P(s_{li} \neq s_{ki}) = \frac{1}{2}$, independently for all $i = 1, 2, \ldots, n$. Consequently, the probability that $\mathbf{s}_l$ and $\mathbf{s}_k$ differ in d positions is simply

$$P(d) = (\tfrac{1}{2})^n \binom{n}{d} \tag{7-2-16}$$

Hence, the expected value of $P_{2m}(\mathbf{s}_l, \mathbf{s}_k)$ over the ensemble of codes may be expressed as

$$\begin{aligned}\overline{P_2(\mathbf{s}_l, \mathbf{s}_k)} &= \sum_{d=0}^{n} P(d) Q\left(\sqrt{\frac{2d\mathscr{E}_c}{N_0}}\right) \\ &= \frac{1}{2^n} \sum_{d=0}^{n} \binom{n}{d} Q\left(\sqrt{\frac{2d\mathscr{E}_c}{N_0}}\right)\end{aligned} \tag{7-2-17}$$

The result (7-2-17) can be simplified if we upper-bound the Q-function as

$$Q\left(\sqrt{\frac{2d\mathscr{E}_c}{N_0}}\right) < e^{-d\mathscr{E}_c/N_0}$$

Thus,

$$\begin{aligned}\overline{P_2(\mathbf{s}_l, \mathbf{s}_k)} &\leq 2^{-n} \sum_{d=0}^{n} \binom{n}{d} e^{-d\mathscr{E}_c/N_0} \\ &\leq 2^{-n}(1 + e^{-\mathscr{E}_c/N_0})^n \\ &\leq [\tfrac{1}{2}(1 + e^{-\mathscr{E}_c/N_0})]^n\end{aligned} \tag{7-2-18}$$

We observe that the right-hand side of (7-2-18) is independent of the indices l and k. Hence, when we substitute the bound (7-2-18) into (7-2-12), we obtain

$$\begin{aligned}\overline{P_e(\mathbf{X}_k)} &\leq \sum_{\substack{l=1 \\ l \neq k}}^{M} \overline{P_2(\mathbf{s}_l, \mathbf{s}_k)} = (M-1)[\tfrac{1}{2}(1 + e^{-\mathscr{E}_c/N_0})]^n \\ &< M[\tfrac{1}{2}(1 + e^{-\mathscr{E}_c/N_0})]^n\end{aligned}$$

Finally, the unconditional average error probability $\bar{P}_e$ is obtained by averaging $\overline{P_e(X_k)}$ over all possible k-bit information sequences. Thus,

$$\begin{aligned}\bar{P}_e &= \sum_k \overline{P_e(\mathbf{X}_k)} P(\mathbf{X}_k) < M[\tfrac{1}{2}(1 + e^{-\mathscr{E}_c/N_0})]^n \sum_k P(\mathbf{X}_k) \\ &< M[\tfrac{1}{2}(1 + e^{-\mathscr{E}_c/N_0})]^n\end{aligned} \tag{7-2-19}$$

This result can be expressed in a more convenient form by first defining a parameter R_0, which is called the *cutoff rate* and has units of bits/dimension, as

$$\begin{aligned}R_0 &= \log_2 \frac{2}{1 + e^{-\mathscr{E}_c/N_0}} \\ &= 1 - \log_2 (1 + e^{-\mathscr{E}_c/N_0}), \quad \text{antipodal signaling}\end{aligned} \tag{7-2-20}$$

Then, (7-2-19) becomes

$$\bar{P}_e < M 2^{-nR_0} = 2^{RT} 2^{-nR_0} \tag{7-2-21}$$

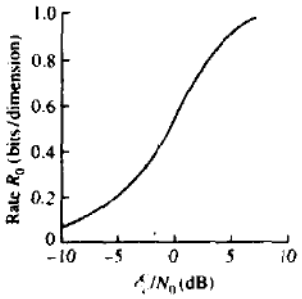


FIGURE 7-2-2 The cutoff rate R_0 as a function of the SNR per dimension in decibels.

Since $n = DT$, (7-2-21) may be expressed as

$$\bar{P}_e < 2^{-T(DR_0 - R)} \tag{7-2-22}$$

The parameter R_0 is plotted as a function of $\mathscr{E}_c/N_0$ in Fig. 7-2-2. We observe that $0 \leq R_0 \leq 1$. Consequently, $\bar{P}_e \to 0$ as $T \to \infty$, provided that the information rate $R < DR_0$.

Alternatively, (7-2-21) may be expressed as

$$\bar{P}_e < 2^{-n(R_0 - R/D)} \tag{7-2-23}$$

The ratio R/D also has units of bits/dimension and may be defined as

$$R_c = \frac{R}{D} = \frac{R}{n/T} = \frac{RT}{n} = \frac{k}{n} \tag{7-2-24}$$

Hence, R_c is the code rate and

$$\bar{P}_e < 2^{-n(R_0 - R_c)} \tag{7-2-25}$$

We conclude that when $R_c < R_0$, the average probability of error $\bar{P}_e \to 0$ as the code block length $n \to \infty$. Since the average value of the probability error can be made arbitrarily small as $n \to \infty$, it follows that there exist codes in the ensemble of 2^{nM} codes that have a probability of error no larger than $\bar{P}_e$.

From the derivation of the average error probability given above, we conclude that good codes exist. Although we do not normally select codes at random, it is interesting to consider the question of whether or not a randomly selected code is likely to be a good code. In fact, we can easily show that there are many good codes in the ensemble. First, we note that $\bar{P}_e$ is an ensemble average of error probabilities over all codes and that all these probabilities are obviously positive quantities. If a code is selected at random, the probability that its error probability $P_e > \alpha \bar{P}_e$ is less than $1/\alpha$. Consequently, no more than 10% of the codes have an error probability that exceeds $10\bar{P}_e$ and no more than 1% of the codes have an error probability that exceeds $100\bar{P}_e$.

We should emphasize that codes with error probabilities exceeding $\bar{P}_e$ are not necessarily poor codes. For example, suppose that an average error rate of $\bar{P}_e < 10^{-10}$ can be attained by using codes with dimensionality n_0 when $R_0 > R_c$. Then, if we select a code with error probability $1000\bar{P}_e = 10^{-7}$, we may compensate for this reduction in error probability by increasing n from n_0 to $n = 10n_0/7$. Thus, by a modest increase in dimensionality, we have a code with $\bar{P}_e < 10^{-10}$. In summary, good codes are abundant and, hence, they are easily found even by random selection.

It is also interesting to express the average error probability in (7-2-25) in terms of the SNR per bit, γ_b. To accomplish this, we express the energy per signal waveform as

$$\mathscr{E} = n\mathscr{E}_c = k\mathscr{E}_b \tag{7-2-26}$$

Hence, $n = k\mathscr{E}_b/\mathscr{E}_c$. We also note that $R_c\mathscr{E}_b/\mathscr{E}_c = 1$. Therefore, (7-2-25) may be expressed as

$$\bar{P}_e < 2^{-k(\gamma_b/\gamma_0 - 1)} \tag{7-2-27}$$

where γ_0 is a normalized SNR parameter, defined as

$$\gamma_0 = \frac{R_c}{R_0}\gamma_b$$

$$= \frac{R_c\gamma_b}{1 - \log_2(1 + e^{-R_c\gamma_b})} \tag{7-2-28}$$

Now, we note that $\bar{P}_e \to 0$ as $k \to \infty$, provided that the SNR per bit, $\gamma_b > \gamma_0$.

The parameter γ_0 is plotted in Fig. 7-2-3 as a function of $R_c\gamma_b$. Note that as $R_c\gamma_b \to 0$, $\gamma_0 \to 2\ln 2$. Consequently, the error probability for M-ary binary coded signals is equivalent to the error probability obtained from the union bound for M-ary orthogonal signals, provided that the signal dimensionality is sufficiently large so that $\gamma_0 \approx 2\ln 2$.

The dimensionality parameter D that we introduced in (7-2-5) is proportional to the channel bandwidth required to transmit the signals. Recall from the sampling theorem that a signal of bandwidth W may be represented by samples taken at a rate of $2W$ samples/s. Thus, in the time interval of length T

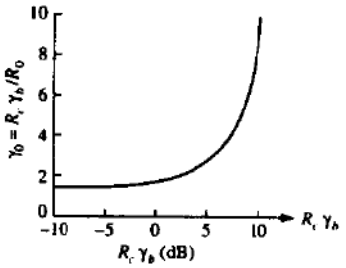


FIGURE 7-2-3 Lower bound on SNR per bit, γ_b, for binary antipodal signals.

there are $n = 2WT$ samples or, equivalently, n degrees of freedom (dimensions). Consequently, D may be equated with $2W$.

Finally, we note that the binary coded signals considered in this section are appropriate when the SNR per dimension is small, e.g., $\mathscr{E}_c/N_0 < 10$. However, when $\mathscr{E}_c/N_0 > 10$, R_0 saturates at 1 bit/dimension. Since the code rate is restricted to be less than R_0, binary coded signals become inefficient at $\mathscr{E}_c/N_0 > 10$. In such a case, we may use nonbinary-coded signals to achieve an increase in the number of bits per dimension. For example, multiple-amplitude coded signal sets can be constructed from nonbinary codes by mapping each code element into one of q possible amplitude levels (as in PAM). Such codes are considered below.

7-2-2 Random Coding Based on M-ary Multiamplitude Signals

Instead of constructing binary-coded signals, suppose we employ nonbinary codes with code words of the form given by (7-2-1), where the code elements c_{ij} are selected from the set $\{0, 1, \ldots, q-1\}$. Each code element is mapped into one of q possible amplitude levels. Thus, we construct signals corresponding to n-dimensional vectors $\{\mathbf{s}_i\}$ as in (7-2-4), where the components $\{s_{ij}\}$ are selected from a multiamplitude set of q possible values. Now, we have q^n possible signals, from which we select $M = 2^{RT}$ signals to transmit k-bit blocks of information. The q amplitudes corresponding to the code elements $\{0, 1, \ldots, q-1\}$ may be denoted by $\{a_1, a_2, \ldots, a_q\}$, and they are assumed to be selected according to some specified probabilities $\{p_i\}$. The amplitude levels are assumed to be equally spaced over the interval $[-\sqrt{\mathscr{E}_c}, \sqrt{\mathscr{E}_c}]$. For example, Fig. 7-2-4 illustrates the amplitude values for $q = 4$. In general, adjacent amplitude levels are separated by $2\sqrt{\mathscr{E}_c}/(q-1)$. This assignment guarantees not only that each component s_{ij} is peak-energy-limited to $\sqrt{\mathscr{E}_c}$, but, also, each code word is constrained in average energy to satisfy the condition

$$|\mathbf{s}_i|^2 < n\mathscr{E}_c \tag{7-2-29}$$

By repeating the derivation given above for random selection of codes in an AWGN channel, we find that the average probability of error is upper-bounded as

$$\bar{P}_e < M2^{-nR_0} = 2^{RT}2^{-nT_0} = 2^{-n(R_0 - R/D)} \tag{7-2-30}$$

where R_0 is defined as

$$R_0 = -\log_2\left(\sum_{l=1}^{q}\sum_{m=1}^{q} p_l p_m e^{-d_{lm}^2/4N_0}\right) \tag{7-2-31}$$

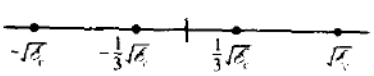


FIGURE 7-2-4 Signal alphabet consisting of four amplitude levels.

and

$$d_{lm} = |a_l - a_m|, \quad l, m = 1, 2, \ldots, q \tag{7-2-32}$$

In the special case where all the amplitude levels are equally likely, $p_l = p_m = 1/q$ and (7-2-31) reduces to

$$R_0 = -\log_2 \left(\frac{1}{q^2} \sum_{l=1}^{q} \sum_{m=1}^{q} e^{-d_{lm}^2/4N_0} \right) \tag{7-2-33}$$

For example, where $q = 2$ and $a_1 = -\sqrt{\mathscr{E}_c}$, $a_2 = \sqrt{\mathscr{E}_c}$, we have $d_{11} = d_{22} = 0$, $d_{12} = d_{21} = 2\sqrt{\mathscr{E}_c}$, and, hence,

$$R_0 = \log_2 \frac{2}{1 + e^{-\mathscr{E}_c/N_0}}, \quad q = 2$$

which agrees with our previous result. When $q = 4$, $a_1 = -\sqrt{\mathscr{E}_c}$, $a_2 = -\sqrt{\mathscr{E}_c}/3$, $a_3 = \sqrt{\mathscr{E}_c}/3$, and $a_4 = \sqrt{\mathscr{E}_c}$, we have $d_{mm} = 0$ for $m = 1, 2, 3, 4$, $d_{12} = d_{23} = d_{34} = d_{21} = d_{32} = d_{43} = 2\sqrt{\mathscr{E}_c}/3$, $d_{13} = d_{31} = d_{24} = d_{42} = 4\sqrt{\mathscr{E}_c}/3$, and $d_{14} = d_{41} = 2\sqrt{\mathscr{E}_c}$. Hence,

$$R_0 = \log_2 \frac{8}{2 + 3e^{-\mathscr{E}_c/9N_0} + 2e^{-4\mathscr{E}_c/9N_0} + e^{-\mathscr{E}_c/N_0}}, \quad q = 4 \tag{7-2-34}$$

Clearly, R_0 now saturates at 2 bits/dimension as $\mathscr{E}_c/N_0$ increases.

The graphs of R_0 as a function of $\mathscr{E}_c/N_0$ for equally spaced and equally likely amplitude levels are shown in Fig. 7-2-5 for $q = 2$, 3, 4, 8, 16, 32, and 64. Note that the saturation level now occurs at $\log_2 q$ bits/dimension. Consequently, for high SNR, $\bar{P}_e \to 0$ as $n \to \infty$, provided that $R < DR_0 = 2WR_0$ bits/s.

If we remove the peak energy constraint on each of the elements, but retain

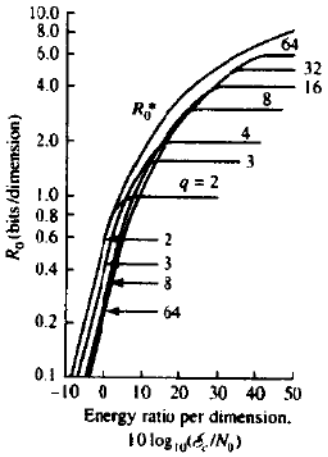


FIGURE 7-2-5 Cutoff rate R_0 for equally spaced q-level amplitude modulation with equal probabilities $p_c = 1/q$. [*From* Principles of Communication Engineering, *by J. M. Wozencraft and I. M. Jacobs.*© *1965 by John Wiley and Sons, Inc. Reprinted with permission of the publisher.*]

the average energy constraint per code word as given by (7-2-29) it is possible to obtain a larger upper bound on the number of bits per dimension. For this case, the result obtained by Shannon (1959b) is

$$R_0^* = \frac{1}{2}\left[1 + \frac{\mathscr{E}_c}{N_0} - \sqrt{1 + \left(\frac{\mathscr{E}_c}{N_0}\right)^2}\right]\log_2 e + \tfrac{1}{2}\log_2\left[\frac{1}{2}\left(1 + \sqrt{1 + \left(\frac{\mathscr{E}_c}{N_0}\right)^2}\right)\right] \tag{7-2-35}$$

The graph of R_0^* as a function of the SNR per dimension, $\mathscr{E}_c/N_0$, is also shown in Fig. 7-2-5. It is clear that our selection of the equally spaced, equally likely amplitude levels that result in R_0 is suboptimum. However, these coded signals are easily generated and implemented in practice. This is an important advantage that justifies their use.

7-2-3 Comparison of R_0^* with the Capacity of the AWGN Channel

The channel capacity of the band-limited additive white gaussian noise channel with an average power constraint on the input signal was derived in Section 7-1-2, and is given by

$$C = W\log_2\left(1 + \frac{P_{av}}{WN_0}\right) \text{ bits/s} \tag{7-2-36}$$

where P_{av} is the average power of the input signal and W is the channel bandwidth. It is interesting to express the capacity of this channel in terms of bits/dimension and the average power in terms of energy/dimension. With $D = 2W$ and

$$\mathscr{E}_c = \frac{\mathscr{E}}{n} = \frac{P_{av}T}{n}$$

we have

$$P_{av} = \frac{n}{T}\mathscr{E}_c = D\mathscr{E}_c \tag{7-2-37}$$

By defining $C_n = C/2W = C/D$ and substituting for W and P_{av}, (7-2-36) may be expressed as

$$C_n = \tfrac{1}{2}\log_2\left(1 + 2\frac{\mathscr{E}_c}{N_0}\right) = \tfrac{1}{2}\log_2(1 + 2R_c\gamma_b) \text{ bits/dimension} \tag{7-2-38}$$

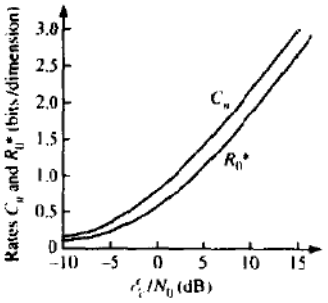


FIGURE 7-2-6 Comparison of cutoff rate R_0^* with the channel capacity for an AWGN channel.

This expression for the normalized capacity may be compared with R_0^*, as shown in Fig. 7-2-6. Since C_n is the ultimate upper limit on the transmission rate R/D, $R_0^* < C_n$ as expected. We also observe that for small values of $\mathscr{E}_c/N_0$, the difference between R_0^* and C_n is approximately 3 dB. Therefore, the use of randomly selected, optimum average power-limited, multiamplitude signals yields a rate function R_0^* that is within 3 dB of the channel capacity. More elaborate bounding techniques are required to show that the probability of error can be made arbitrarily small when $R < DC_n = 2WC_n = C$.

7-3 COMMUNICATION SYSTEM DESIGN BASED ON THE CUTOFF RATE

In the foregoing discussion, we characterized coding and modulation performance in terms of the error probability, which is certainly a meaningful criterion for system design. However, in many cases, the computation of the error probability is extremely difficult, especially if nonlinear operations such as signal quantization are performed in processing the signal at the receiver, or if the additive noise is nongaussian.

Instead of attempting to compute the exact probability of error for specific codes, we may use the ensemble average probability of error for randomly selected code words. The channel is assumed to have q input symbols $\{0, 1, \ldots, q-1\}$ and Q output symbols $\{0, 1, \ldots, Q-1\}$, and to be characterized by the transition probabilities $P(i \mid j)$, where $j = 0, 1, \ldots, q-1$ and $i = 0, 1, \ldots, Q-1$, with $Q \geq q$. The input symbols occur with probabilities $\{p_j\}$ and are assumed to be statistically independent. In addition, the noise on the channel is assumed to be statistically independent in time, so that there is no dependence among successive received symbols. Under these conditions, the ensemble average probability of error for random selected code words may be derived by applying the Chernoff bound (see Viterbi and Omura, 1979).

The general result that is obtained for the discrete memoryless channel is

$$\bar{P}_e < 2^{-n(R_0 - RD)} \tag{7-3-1}$$

where n is the block length of the code, R is the information rate in bits/s, D is

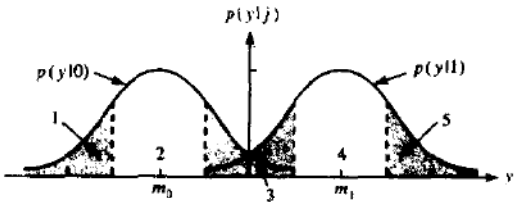


FIGURE 7-3-1 Example of quantization of the demodulator output into five levels.

the number of dimensions per second, and R_Q is the cutoff rate for a quantizer with Q levels, defined as

$$R_Q = \max_{\{p_j\}} \left\{ -\log_2 \sum_{i=0}^{Q-1} \left[\sum_{j=0}^{q-1} p_j \sqrt{P(i \mid j)} \right]^2 \right\} \tag{7-3-2}$$

From the viewpoint of code design, the combination of modulator, waveform channel, and demodulator constitutes a discrete-time channel with q inputs and Q outputs. The transition probabilities $\{P(i \mid j)\}$ depend on the channel noise characteristics, the number of quantization levels, and the type of quantizer, e.g., uniform or nonuniform. For example, in the binary-input AWGN channel, the output of the correlator at the sampling instant may be expressed as

$$p(y \mid j) = \frac{1}{\sqrt{2\pi}\sigma} e^{-(y-m_j)^2/2\sigma^2}, \quad j = 0, 1 \tag{7-3-3}$$

where $m_0 = -\sqrt{\mathscr{E}_c}$, $m_1 = \sqrt{\mathscr{E}_c}$, and $\sigma^2 = \frac{1}{2}N_0$. These two pdfs are shown in Fig. 7-3-1. Also illustrated in the figure is a quantization scheme that subdivides the real line into five regions. From such a subdivision, we may compute the transition probabilities and optimally select the thresholds that subdivide the regions in a way that maximizes R_Q for any given Q. Thus,

$$P(i \mid j) = \int_{r_i} p(y \mid j)\, dy \tag{7-3-4}$$

where the integral of $p(y \mid j)$ is evaluated over the region r_i that corresponds to the transition probability $P(i \mid j)$.

The value of the rate R_Q in the limit as $Q \to \infty$ yields the cutoff rate for the unquantized decoder. It is relatively straightforward to show that as $Q \to \infty$, the first summation (sum from $i = 0$ to $Q - 1$) in (7-3-2) becomes an integral and the transition probabilities are replaced by the corresponding pdfs. Thus, when the channel consists of q discrete inputs and one continuous output y, which represents the unquantized output from a matched filter or a cross-correlator in a system that employs either PSK or a multiamplitude (PAM) modulation, the cutoff rate is given by

$$R_0 = \max_{\{p_j\}} \left\{ -\log_2 \int_{-\infty}^{\infty} dy \left[\sum_{j=0}^{q-1} p_j \sqrt{p(y \mid j)} \right]^2 \right\} \tag{7-3-5}$$

where p_j, $0 \leq j \leq q - 1$, is the probability of transmitting the jth symbol and

$p(y \mid j)$ is the conditional probability density function of the output y from the matched filter or cross-correlator when the jth signal is transmitted. This is the desired expression for unquantized (soft-decision) decoding.

We observe that when the input signal is binary PSK with $p_0 = p_1 = \frac{1}{2}$ and the noise is additive, white, and gaussian, (7-3-5) reduces to the familiar result given previously in (7-2-20).

The general expressions in (7-3-5) and (7-3-2) allow us to compare the performance of various receiver implementations based on a different number of quantization levels.

Example 7-3-1

Let us compare the performance of a binary PSK input signal in an AWGN channel when the receiver quantizes the output to $Q = 2$, 4, and 8 levels. To simplify the optimization problem for the quantization of the signal at the output of the demodulator, the quantization levels are placed at $0, \pm\tau_h, \pm 2\tau_h, \ldots, \pm(2^{b-1} - 1)\tau_h$, where τ_h is the *quantizer step-size parameter*, which is to be selected, and b is the number of bits of the quantizer. A good strategy for the selection of τ_h is to choose it to minimize the SNR per bit γ_b that is required for operation at a code rate R_0. This implies that the step-size parameter must be optimized for every SNR, which in a practical implementation of the receiver means that the SNR must be measured. Fortunately, τ_h does not exhibit high sensitivity to small changes in SNR, so that it is possible to optimize τ_h for one SNR and obtain good performance for a wide range of SNRs about this nominal value by using a fixed τ_h.

Based on this approach, the expression for R_Q given by (7-3-2) was evaluated for $b = 1$ (hard-decision decoding), 2, and 3 bits, corresponding to $Q = 2$, 4, and 8 levels of quantization. The results are plotted in Fig. 7-3-2. The value of R_0 for unquantized soft-decision decoding, obtained by evaluating (7-3-5) is also shown in Fig. 7-3-2. We observe that two-bit quantization with $\tau_h = 1.0$ gains about 1.4 dB over hard-decision decoding, and three-bit quantization with $\tau_h = 0.5$ yields an additional 0.4 dB improvement. Thus, with a three-bit quantizer, we are within 0.2 dB of the

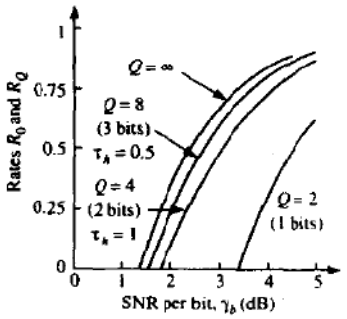


FIGURE 7-3-2 Effect of quantization on the performance of a coded communications system operating at a rate $R = R_0$ or $R = R_Q$, with binary PSK modulation on an AWGN channel.

unquantized soft-decision decoding limit. Clearly, there is little to be gained by increasing the precision any further.

When a nonbinary code is used in conjunction with M-ary ($M = q$) signaling, the received signal at the output of the M matched filters may be represented by the vector $\mathbf{y} = [y_1 \ y_2 \ \cdots \ y_M]$. The cutoff rate for this M-input, M-output (unquantized) channel is

$$R_0 = \max_{\{p_i\}} \left\{ -\log_2 \sum_{j=0}^{M-1} \sum_{i=0}^{M-1} p_i p_j \int_{-\infty}^{\infty} \sqrt{p(\mathbf{y} \mid j) p(\mathbf{y} \mid i)}\, d\mathbf{y} \right\} \tag{7-3-6}$$

where $p(\mathbf{y} \mid j)$ is the conditional probability density function of the output vector $\mathbf{y}$ from the demodulator given that the jth signal was transmitted. Note that (7-3-6) is similar in form to (7-3-5) except that we now have an M-fold integral to perform because there are M outputs from the demodulator.

Let us assume that the M signals are orthogonal so that the M outputs conditioned on a particular input signal are statistically independent. As a consequence,

$$p(\mathbf{y} \mid j) = p_{s+n}(y_j) \prod_{\substack{i=0 \\ i \neq j}}^{M-1} p_n(y_i) \tag{7-3-7}$$

where $p_{s+n}(y_j)$ is the pdf of the matched filter output corresponding to the transmitted signal and $\{p_n(y_i)\}$ corresponds to the noise-only outputs from the other $M - 1$ matched filters. When (7-3-7) is incorporated into (7-3-6) we obtain

$$R_0 = \max_{\{p_i\}} \left\{ -\log_2 \left[\sum_{j=0}^{M-1} p_j^2 + \sum_{j=0}^{M-1} \sum_{\substack{i=0 \\ i \neq j}}^{M-1} p_i p_j \left(\int_{-\infty}^{\infty} dy \sqrt{p_{s+n}(y) p_n(y)} \right)^2 \right] \right\} \tag{7-3-8}$$

The maximization of R_0 over the set of input probabilities yields $p_j = 1/M$ for $1 \leq j \leq M$. Consequently, (7-3-8) reduces to

$$\begin{aligned} R_0 &= \log_2 \left\{ \frac{M}{1 + (M-1)\left[\int_{-\infty}^{\infty} \sqrt{p_{s+n}(y) p_n(y)}\, dy\right]^2} \right\} \\ &= \log_2 M - \log_2 \left\{ 1 + (M-1) \left[\int_{-\infty}^{\infty} \sqrt{p_{s+n}(y) p_n(y)}\, dy \right]^2 \right\} \end{aligned} \tag{7-3-9}$$

This is the desired result for the cutoff rate of an M-ary input, M-ary vector output unquantized channel.

For phase coherent detection of the M-ary orthogonal signals the appropriate pdfs are

$$\begin{aligned} p_{s+n}(y) &= \frac{1}{\sqrt{2\pi}\,\sigma} e^{-(y-m)^2/2\sigma^2} \\ p_n(y) &= \frac{1}{\sqrt{2\pi}\,\sigma} e^{-y^2/2\sigma^2} \end{aligned} \tag{7-3-10}$$

where $m = \sqrt{\mathscr{E}}$ and $\sigma^2 = \frac{1}{2}N_0$. Substituting these relations into (7-3-9) and evaluating the integral yields

$$R_0 = \log_2 \left[\frac{M}{1 + (M-1)e^{-\mathscr{E}/2N_0}} \right]$$

$$= \log_2 \left[\frac{M}{1 + (M-1)e^{-R_w \gamma_b/2}} \right] \tag{7-3-11}$$

where $\mathscr{E}$ is the received energy per waveform, R_w is the information rate in bits/waveform, and $\gamma_b = \mathscr{E}_b/N_0$ is the SNR per bit.

We should emphasize that the rate parameter R_w has imbedded in it the code rate R_c. For example, if $M = 2$ and the code is binary then $R_w = R_c$. More generally, if the code is binary and $M = 2^\nu$ then each M-ary waveform conveys $R_w = \nu R_c$ bits of information. It is also interesting to note that if the code is binary and $M = 2$ then (7-3-11) reduces to

$$R_0 = \log_2 \left(\frac{2}{1 + e^{-R_c \gamma_b/2}} \right), \quad M = 2 \text{ orthogonal signals} \tag{7-3-12}$$

which is 3 dB worse than the cutoff rate for antipodal signals. If we set $R_w = R_0$ in (7-3-11) and solve for γ_b, we obtain

$$\gamma_b = \frac{2}{R_0} \ln \left(\frac{M-1}{2^{-R_0}M - 1} \right) \tag{7-3-13}$$

Graphs of R_0 versus γ_b for several values of M are illustrated in Fig. 7-3-3. Note that the curve for any value of M saturates at $R_0 = \log_2 M$.

It is also interesting to consider the limiting form of (7-3-11) in the limit as $M \to \infty$. This yields

$$\lim_{M \to \infty} R_0 = \frac{\mathscr{E}}{2N_0 \ln 2} \text{ bits/waveform} \tag{7-3-14}$$

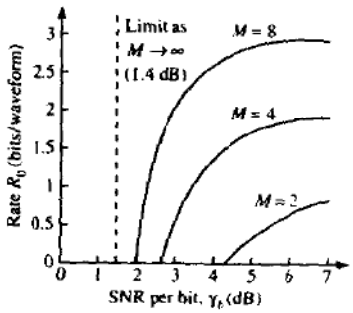


FIGURE 7-3-3 SNR per bit required to operate at a rate R_0 with M-ary orthogonal signals detected coherently in an AWGN channel.

Since $\mathscr{E} = P_{av}T$, where T is the time interval per waveform, it follows that

$$\lim_{M\to\infty} \frac{R_0}{T} = \frac{P_{av}}{2N_0 \ln 2} = \tfrac{1}{2}C_\infty \tag{7-3-15}$$

Hence, in the limit as $M \to \infty$, the cutoff rate is one-half of the capacity for the infinite bandwidth AWGN channel. Alternatively, the substitution of $\mathscr{E} = R_0\mathscr{E}_b$ into (7-3-14) yields $\gamma_b = 2 \ln 2$ (1.4 dB), which is the minimum SNR required to operate at R_0 (as $M \to \infty$). Hence, signaling at a rate R_0 requires 3 dB more power than the Shannon limit.

The value of R_0 given in (7-3-11) is based on the use of M-ary orthogonal signals, which are clearly suboptimal when M is small. If we attempt to maximize R_0 by selecting the best set of M waveforms, we should not be surprised to find that the simplex set of waveforms is optimum. In fact, R_0 for these optimum waveforms is simply given as

$$R_0 = \log_2\left[\frac{M}{1 + (M-1)e^{-M\mathscr{E}/2(M-1)N_0}}\right] \tag{7-3-16}$$

If we compare this expression with (7-3-11) we observe that R_0 in (7-3-16) simply reflects the fact that the simplex set is more energy-efficient by a factor $M/(M-1)$.

In the case of noncoherent detection, the probability density functions corresponding to signal-plus-noise and noise alone may be expressed as

$$\begin{aligned} p_{s+n}(y) &= ye^{-(y^2+a^2)/2}I_0(ay), \quad y \geq 0 \\ p_n(y) &= ye^{-y^2/2}, \quad y \geq 0 \end{aligned} \tag{7-3-17}$$

where, by definition, $a = \sqrt{2\mathscr{E}/N_0}$. The computation of R_0 given by (7-3-9) does not yield a closed-form solution. Instead, the integral in (7-3-9) must be evaluated numerically. Results for this case have been given by Jordan (1966) and Bucher (1980). For example, the (normalized) cutoff rate R_0 for M-ary orthogonal signals with noncoherent detection is shown in Fig. 7-3-4 for

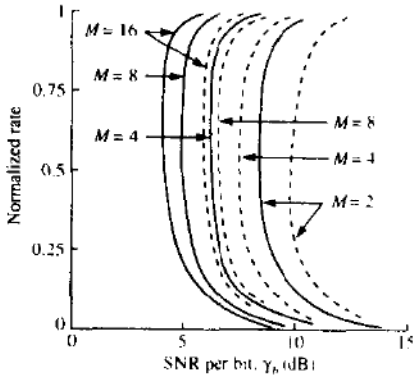


FIGURE 7-3-4 SNR per bit required to operate at a rate R_0 with M-ary orthogonal signals detected noncoherently in an AWGN channel.

$M = 2, 4, 8$, and 16. For purposes of comparison we also plot the cutoff rate for hard-decision decoding ($Q = M$) of the M-ary symbols. In this case, we have

$$R_Q = \log_2 \left\{ \frac{M}{[\sqrt{(1 - P_M)} + \sqrt{(M-1)P_M}]^2} \right\}, \quad Q = M \quad (7\text{-}3\text{-}18)$$

where P_M is the probability of a symbol error. For a relatively broad range of rates, the difference between soft- and hard-decision decoding is approximately 2 dB.

The most striking characteristic of the performance curves in Fig. 7-3-4 is that there is an optimum code rate for any given M. Unlike the case of coherent detection, where the SNR per bit decreases monotonically with a decrease in code rate, the SNR per bit for noncoherent detection reaches a minimum in the vicinity of a normalized rate of 0.5, and increases for both high and low rates. The minimum is rather broad, so there is really a range of rates from 0.2 to 0.9 where the SNR per bit is within 1 dB of the minimum. This characteristic behavior in the performance with noncoherent detection is attributed to the nonlinear characteristic of the detector.

7-4 BIBLIOGRAPHICAL NOTES AND REFERENCES

The pioneering work on channel characterization in terms of channel capacity and random coding was done by Shannon (1948a, b, 1949). Additional contributions were subsequently made by Gilbert (1952), Elias (1955), Gallager (1965), Wyner (1965), Shannon *et al.* (1967), Forney (1968) and Viterbi (1969). All of these early publications are contained in the IEEE Press book entitled *Key Papers in the Development of Information Theory*, edited by Slepian (1974).

The use of the cutoff rate parameter as a design criterion was proposed and developed by Wozencraft and Kennedy (1966) and by Wozencraft and Jacobs (1965). It was used by Jordan (1966) in the design of coded waveforms for M-ary orthogonal signals with coherent and noncoherent detection. Following these pioneering works, the cutoff rate has been widely used as a design criterion for coded signals in a variety of different channel conditions.

PROBLEMS

7-1 Show that the following two relations are necessary and sufficient conditions for the set of input probabilities $\{P(x_j)\}$ to maximize $I(X; Y)$ and, thus, to achieve capacity for a DMC:

$$I(x_j; Y) = C \quad \text{for all } j \text{ with } P(x_j) > 0$$
$$I(x_j; Y) \leq C \quad \text{for all } j \text{ with } P(x_j) = 0$$

where C is the capacity of the channel and

$$I(x_j; Y) = \sum_{i=0}^{Q-1} P(y_i \mid x_j) \log \frac{P(y_i \mid x_j)}{P(y_i)}$$

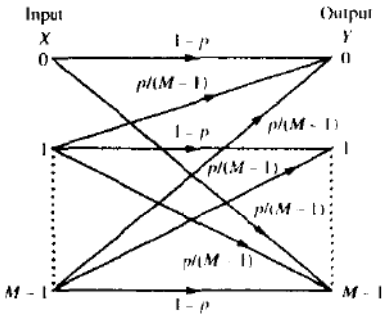


FIGURE P7-2

7-2 Figure P7-2 illustrates an M-ary symmetric DMC with transition probabilities $P(y \mid x) = 1 - p$ when $x = y = k$ for $k = 0, 1, \ldots, M-1$, and $P(y \mid x) = p/(M-1)$ when $x \neq y$.

a Show that this channel satisfies the condition given in Problem 7-1 when $P(x_k) = 1/M$.

b Determine and plot the channel capacity as a function of p.

7-3 Determine the capacities of the channels shown in Fig. P7-3.

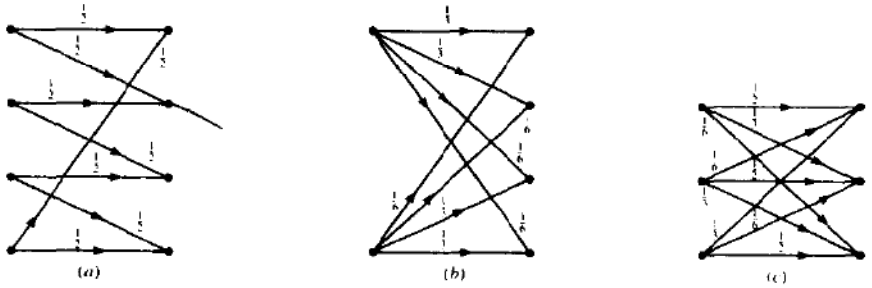


FIGURE P7-3

7-4 Consider the two channels with the transition probabilities as shown in Fig. P7-4. Determine if equally probable input symbols maximize the information rate through the channel.

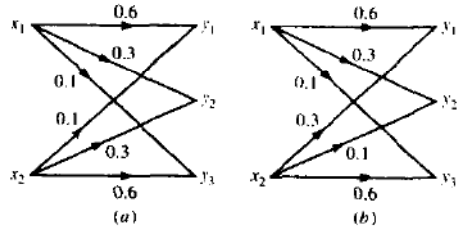


FIGURE P7-4

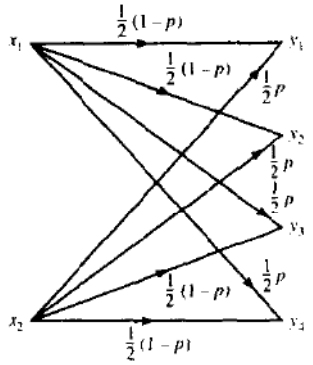


FIGURE P7-6

7-5 A telephone channel has a bandwidth $W = 3000$ Hz and a signal-to-noise power ratio of 400 (26 dB). Suppose we characterize the channel as a band-limited AWGN waveform channel with $P_{av}/WN_0 = 400$.

a Determine the capacity of the channel in bits/s.

b Is the capacity of the channel sufficient to support the transmission of a speech signal that has been sampled and encoded by means of logarithmic PCM?

c Usually, channel impairments other than additive noise limit the transmission rate over the telephone channel to less than the channel capacity of the equivalent band-limited AWGN channel considered in (a). Suppose that a transmission rate of $0.7C$ is achievable in practice without channel encoding. Which of the speech source encoding methods described in Section 3-5 provide sufficient compression to fit the bandwidth restrictions of the telephone channel?

7-6 Consider the binary-input, quaternary-output DMC shown in Fig. P7-6.

a Determine the capacity of the channel.

b Show that this channel is equivalent to a BSC.

7-7 Determine the channel capacity for the channel shown in Fig. P7-7.

7-8 Consider a BSC with crossover probability of error p. Suppose that R is the number of bits in a source code word that represents one of 2^R possible levels at the output of a quantizer. Determine

a the probability that a code word transmitted over the BSC is received correctly;

b the probability of having at least one bit error in a code word transmitted over the BSC;

c the probability of having n_e or less bit errors in a code word;

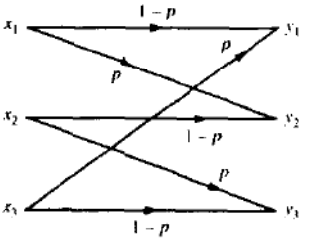


FIGURE P7-7

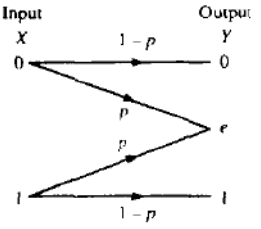


FIGURE P7-10

d Evaluate the probability in (a), (b), and (c) for $R = 5$, $p = 0.01$, and $n_c = 5$.

7-9 Show that, for a DMC, the average mutual information between a sequence $X_1X_2 \cdots X_n$ of channel inputs and the corresponding channel outputs satisfy the condition

$$I(X_1X_2 \cdots X_n; Y_1, Y_2, \ldots, Y_n) \leq \sum_{i=1}^{n} I(X_i; Y_i)$$

with equality if and only if the set of input symbols is statistically independent.

7-10 Figure P7-10 illustrates a binary erasure channel with transition probabilities $P(0 \mid 0) = P(1 \mid 1) = 1 - p$ and $P(e \mid 0) = P(e \mid 1) = p$. The probabilities for the input symbols are $P(X = 0) = \alpha$ and $P(X = 1) = 1 - \alpha$.

a Determine the average mutual information $I(X; Y)$ in bits.

b Determine the value of α that maximizes $I(X; Y)$, i.e., the channel capacity C in bits/channel use, and plot C as a function of p for the optimum value of α.

c For the value of α found in (b), determine the mutual information $I(x; y) = I(0; 0)$, $I(1; 1)$, $I(0; e)$, and $I(1; e)$.

7-11 Consider the binary-input, ternary-output channel with transition probabilities shown in Fig. P7-11, where e denotes an erasure. For the AWGN channel, α and p are defined as

$$\alpha = \frac{1}{\sqrt{\pi N_0}} \int_{-\beta}^{\beta} e^{-(x+\sqrt{\mathscr{E}_c})^2/N_0}\, dx$$

$$p = \frac{1}{\sqrt{\pi N_0}} \int_{\beta}^{\infty} e^{-(x+\sqrt{\mathscr{E}_c})^2/N_0}\, dx$$

a Determine R_Q for $Q = 3$ as a function of the probabilities α and p.

b The rate parameter R_Q depends on the choice of the threshold β through the probabilities α and p. For any $\mathscr{E}_c/N_0$, the value of β that maximizes R_Q can be determined by trial and error. For example, it can be shown that for $\mathscr{E}_c/N_0$ below 0 dB, $\beta_{\text{opt}} \approx 0.65\sqrt{\frac{1}{2}N_0}$; for $1 \leq \mathscr{E}_c/N_0 \leq 10$, β_{opt} varies approximately

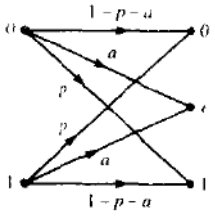


FIGURE P7-11

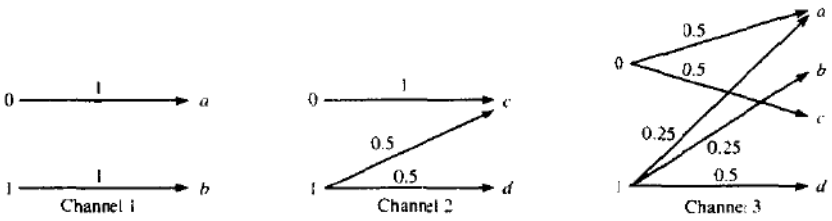


FIGURE P7-13

linearly between $0.65\sqrt{\frac{1}{2}N_0}$ and $1.0\sqrt{\frac{1}{2}N_0}$. By using $\beta = 0.65\sqrt{\frac{1}{2}N_0}$ for the entire range of $\mathscr{E}_c/N_0$, plot R_Q versus $\mathscr{E}_c/N_0$ and compare this result with R_Q ($Q = \infty$).

7-12 Find the capacity of the cascade connection of n binary-symmetric channels with the same crossover probability ϵ. What is the capacity when the number of channels goes to infinity?

7-13 Channels 1, 2, and 3 are shown in Fig. P7-13.

a Find the capacity of channel 1. What input distribution achieves capacity?

b Find the capacity of channel 2. What input distribution achieves capacity?

c Let C denote the capacity of the third channel and C_1 and C_2 represent the capacities of the first and second channel. Which of the following relations holds true and why?

$$C < \tfrac{1}{2}(C_1 + C_2) \quad \text{(i)}$$

$$C = \tfrac{1}{2}(C_1 + C_2) \quad \text{(ii)}$$

$$C > \tfrac{1}{2}(C_1 + C_2) \quad \text{(iii)}$$

7-14 Let C denote the capacity of a discrete memoryless channel with input alphabet $\mathscr{X} = \{x_1, x_2, \ldots, x_N\}$ and output alphabet $\mathscr{Y} = \{y_1, y_2, \ldots, y_M\}$. Show that $C \leq \min\{\log M, \log N\}$.

7-15 The channel C (known as the Z channel) is shown in Fig. P7-15.

a Find the input probability distribution that achieves capacity.

b What is the input distribution and capacity for the special cases $\epsilon = 0$, $\epsilon = 1$, and $\epsilon = 0.5$?

c Show that if n such channels are cascaded, the resulting channel will be equivalent to a Z channel with $\epsilon_1 = \epsilon^n$.

d What is the capacity of the equivalent Z channel when $n \to \infty$.

7-16 Find the capacity of an additive white Gaussian noise channel with a bandwidth 1 MHz, power 10 W, and noise power spectral density $\frac{1}{2}N_0 = 10^{-9}$ W/Hz.

7-17 Channel C_1 is an additive white gaussian noise channel with a bandwidth W, average transmitter power P, and noise power spectral density $\frac{1}{2}N_0$. Channel C_2 is

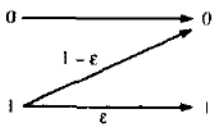


FIGURE P7-15

an additive gaussian noise channel with the same bandwidth and power as channel C_1 but with noise power spectral density $\Phi_n(f)$. It is further assumed that the total noise power for both channels is the same; that is,

$$\int_{-W}^{W} \Phi_n(f)\,df = \int_{-W}^{W} \tfrac{1}{2}N_0\,df = N_0 W$$

Which channel do you think has a larger capacity? Give an intuitive reasoning.

7-18 A discrete-time memoryless gaussian source with mean 0 and variance σ^2 is to be transmitted over a binary-symmetric channel with crossover probability p.

a What is the minimum value of the distortion attainable at the destination (distortion is measured in mean-squared error)?

b If the channel is a discrete-time memoryless additive gaussian noise channel with input power P and noise power P_n, what is the minimum attainable distortion?

c Now assume that the source has the same basic properties but is not memoryless. Do you expect the distortion in transmission over the binary-symmetric channel to be decreased or increased? Why?

7-19 X is a binary memoryless source with $p(X = 0) = 0.3$. This source is transmitted over a binary-symmetric channel with crossover probability $p = 0.1$.

a Assume that the source is directly connected to the channel, i.e., no coding is employed. What is the error probability at the destination?

b If coding is allowed, what is the minimum possible error probability in the reconstruction of the source?

c For what values of p is reliable transmission possible (with coding, of course)?

7-20 Plot the capacity of an AWGN channel that employs binary antipodal signaling, with optimal bit-by-bit detection at the receiver, as a function of $\mathscr{E}_b/N_0$. On the same axis, plot the capacity of the same channel when binary orthogonal signaling is employed.

7-21 In a coded communication system, M messages $1, 2, \ldots, M = 2^k$ are transmitted by M *baseband* signals $x_1(t), x_2(t), \ldots, x_M(t)$, each of duration nT. The general form of $x_i(t)$ is given by

$$x_i(t) = \sum_{j=0}^{n-1} f_{ij}(t - jT)$$

where $f_{ij}(t)$ can be either of the two signals $f_1(t)$ or $f_2(t)$, where $f_1(t) = f_2(t) \equiv 0$ for all $t \notin [0, T]$. We further assume that $f_1(t)$ and $f_2(t)$ have equal energy $\mathscr{E}$ and the channel is ideal (no attenuation) with additive white gaussian noise of power spectral density $\frac{1}{2}N_0$. This means that the received signal is $r(t) = x(t) + n(t)$, where $x(t)$ is one of the $x_i(t)$ and $n(t)$ represents the noise.

a With $f_1(t) = -f_2(t)$, show that N, the dimensionality of the signal space, satisfies $N \leq n$.

b Show that, in general, $N \leq 2n$.

c With $M = 2$, show that, for general $f_1(t)$ and $f_2(t)$,

$$p(\text{error} \mid x_1(t) \text{ sent}) \leq \int_{R^N} \cdots \int \sqrt{p(\mathbf{r} \mid \mathbf{x}_1)p(\mathbf{r} \mid \mathbf{x}_2)}\,d\mathbf{r}$$

where $\mathbf{r}$, $\mathbf{x}_1$, and $\mathbf{x}_2$ are the vector representations of $r(t)$, $x_1(t)$, and $x_2(t)$ in the N-dimensional space.

d Using the result of (c), show that, for general M,

$$p(\text{error} \mid x_m(t) \text{ sent}) \leq \sum_{\substack{1 \leq m' \leq M \\ m' \neq m}} \int \cdots \int_{R^N} \sqrt{p(\mathbf{r} \mid \mathbf{x}_m) p(\mathbf{r} \mid \mathbf{x}_{m'})}\, d\mathbf{r}$$

e Show that

$$\int \cdots \int_{R^N} \sqrt{p(\mathbf{r} \mid \mathbf{x}_m) p(\mathbf{r} \mid \mathbf{x}_{m'})}\, d\mathbf{r} = \exp\left(-\frac{|\mathbf{x}_m - \mathbf{x}_{m'}|^2}{4N_0}\right)$$

and, therefore,

$$p(\text{error} \mid x_m(t) \text{ sent}) \leq \sum_{\substack{1 \leq m' \leq M \\ m' \neq m}} \exp\left(-\frac{|\mathbf{x}_m - \mathbf{x}_{m'}|^2}{4N_0}\right)$$

8

BLOCK AND CONVOLUTIONAL CHANNEL CODES

In Chapter 7, we treated channel coding and decoding from a general viewpoint, and showed that even randomly selected codes on the average yield performances close to the capacity of a channel. In the case of orthogonal signals, we demonstrated that the channel capcity limit can be achieved as the number of signals approaches infinity.

In this chapter, we describe specific codes and evaluate their performance for the additive white gaussian noise channel. In particular, we treat two classes of codes, namely, linear block codes and convolutional codes. The code performance is evaluated for both hard-decision decoding and soft-decision decoding.

8-1 LINEAR BLOCK CODES

A block code consists of a set of fixed-length vectors called *code words*. The length of a code word is the number of elements in the vector and is denoted by n. The elements of a code word are selected from an alphabet of q elements. When the alphabet consists of two elements, 0 and 1, the code is a binary code and the elements of any code word are called bits. When the elements of a code word are selected from an alphabet having q elements ($q > 2$), the code is nonbinary. It is interesting to note that when q is a power of 2, i.e., $q = 2^b$ where b is a positive integer, each q-ary element has an equivalent binary representation consisting of b bits, and, thus, a nonbinary code of block length N can be mapped into a binary code of block length $n = bN$.

There are 2^n possible code words in a binary block code of length n. From

these 2^n code words, we may select $M = 2^k$ code words ($k < n$) to form a code. Thus, a block of k information bits is mapped into a code word of length n selected from the set of $M = 2^k$ code words. We refer to the resulting block code as an (n, k) code, and the ratio $k/n \equiv R_c$ is defined to be the *rate* of the code. More generally, in a code having q elements, there are q^n possible code words. A subset of $M = 2^k$ code words may be selected to transmit k-bit blocks of information.

Besides the code rate parameter R_c, an important parameter of a code word is its *weight*, which is simply the number of nonzero elements that it contains. In general, each code word has its own weight. The set of all weights in a code constitutes the *weight distribution* of the code. When all the M code words have equal weight, the code is called a *fixed-weight code* or a *constant-weight code*.

The encoding and decoding functions involve the arithmetic operations of addition and multiplication performed on code words. These arithmetic operations are performed according to the conventions of the algebraic field that has as its elements the symbols contained in the alphabet. For example, the symbols in a binary alphabet are 0 and 1; hence, the field has two elements. In general, a field F consists of a set of elements that has two arithmetic operations defined on its elements, namely, addition and multiplication, that satisfy the following properties (axioms).

Addition

1 The set F is closed under addition, i.e., if $a, b \in F$ than $a + b \in F$.

2 Addition is associative, i.e., if a, b, and c are elements of F then $a + (b + c) = (a + b) + c$.

3 Addition is commutative, i.e., $a + b = b + a$.

4 The set contains an element called *zero* that satisfies the condition $a + 0 = a$.

5 Every element in the set has its own negative element. Hence, if b is an element, its negative is denoted by $-b$. The subtraction of two elements, such as $a - b$, is defined as $a + (-b)$.

Multiplication

1 The set F is closed under multiplication, i.e., if $a, b \in F$ then $ab \in F$.

2 Multiplication is associative, i.e., $a(bc) = (ab)c$.

3 Multiplication is commutative, i.e., $ab = ba$.

4 Multiplication is distributive over addition, i.e., $(a + b)c = ac + bc$.

5 The set F contains an element, called the *identity*, that satisfies the condition $a(1) = a$, for any element $a \in F$.

6 Every element of F, except zero, has an inverse. Hence, if $b \in F$ ($b \neq 0$)

then its inverse is defined as b^{-1}, and $bb^{-1} = 1$. The division of two elements, such as $a \div b$, is defined as ab^{-1}

We are very familiar with the field of real numbers and the field of complex numbers. These fields have an infinite number of elements. However, as indicated above, codes are constructed from fields with a finite number of elements. A finite field with q elements is generally called a *Galois field* and denoted by GF(q).

Every field must have a *zero* element and a *one* element. Hence, the simplest field is GF(2). In general, when q is a prime, we can construct the finite field GF(q) consisting of the elements $\{0, 1, \ldots, q-1\}$. The addition and multiplication operations on the elements of GF(q) are defined modulo q and denoted as (mod q). For example, the addition and multiplication tables for GF(2) are

+	0	1
0	0	1
1	1	0

·	0	1
0	0	0
1	0	1

which are operations (mod 2). Similarly, the field GF(5) is a set consisting of the elements {0, 1, 2, 3, 4}. The addition and multiplication tables for GF(5) are

+	0	1	2	3	4
0	0	1	2	3	4
1	1	2	3	4	0
2	2	3	4	0	1
3	3	4	0	1	2
4	4	0	1	2	3

·	0	1	2	3	4
0	0	0	0	0	0
1	0	1	2	3	4
2	0	2	4	1	3
3	0	3	1	4	2
4	0	4	3	2	1

In general, the finite field GF(q) can be constructed only if q is a prime or a power of a prime. When q is a prime, multiplication and addition are based on modulo-q arithmetic as illustrated above. If $q = p^m$ where p is a prime and m is any positive integer, it is possible to extend the field GF(p) to the field GF(p^m). This is called the *extension field* of GF(p). Multiplication and addition of the elements in the extension field are based on modulo-p arithmetic.

With this brief introduction to the arithmetic operations that may be performed on the elements of code words, let us now consider some basic characteristics of block codes.

Suppose $\mathbf{C}_i$ and $\mathbf{C}_j$ are any two code words in an (n, k) block code. A measure of the difference between the code words is the number of corresponding elements or positions in which they differ. This measure is called the *Hamming distance* between the two code words and is denoted as d_{ij}.

Clearly, d_{ij} for $i \neq j$ satisfies the condition $0 < d_{ij} \leq n$. The smallest value of the set $\{d_{ij}\}$ for the M code words is called the *minimum distance* of the code and is denoted as $d_{\min}$. Since the Hamming distance is a measure of the separation between pairs of code words, it is intimately related to the cross-correlation coefficient between corresponding pairs of waveforms generated from the code words. The relationship is discussed in Section 8-1-4.

Besides characterizing a code as being binary or nonbinary, one can also describe it as either linear or nonlinear. Suppose $\mathbf{C}_i$ and $\mathbf{C}_j$ are two code words in an (n, k) block code and let α_1 and α_2 be any two elements selected from the alphabet. Then the code is said to be linear if and only if $\alpha_1\mathbf{C}_i + \alpha_2\mathbf{C}_j$ is also a code word. This definition implies that a linear code must contain the all-zero code word. Consequently a constant-weight code is nonlinear.

Suppose we have a binary linear block code, and let $\mathbf{C}_i$, $i = 1, 2, \ldots, M$, denote the M code words. For convenience, let $\mathbf{C}_1$ denote the all-zero code word, i.e., $\mathbf{C}_1 = [0\,0 \ldots 0]$, and let w_r denote the weight of the rth code word. It follows that w_r is the Hamming distance between the code words $\mathbf{C}_r$ and $\mathbf{C}_1$. Thus, the distance $d_{1r} = w_r$. In general, the distance d_{ij} between any pair of code words $\mathbf{C}_i$ and $\mathbf{C}_j$ is simply equal to the weight of the code word formed by taking the difference between $\mathbf{C}_i$ and $\mathbf{C}_j$. Since the code is linear, the difference (equivalent to taking the modulo-2 sum for a binary code) between $\mathbf{C}_i$ and $\mathbf{C}_j$ is also a code word having a weight included in the set $\{w_r\}$. Hence, the weight distribution of a linear code completely characterizes the distance properties of the code. The minimum distance of the code is, therefore,

$$d_{\min} = \min_{r, r \neq 1} \{w_r\} \tag{8-1-1}$$

A number of elementary concepts from linear algebra are particularly useful in dealing with linear block codes. Specifically, the set of all n-tuples (vectors with n elements) form a vector space S. If we select a set of $k < n$ linearly independent vectors from S and from these construct the set of all linear combinations of these vectors, the resulting set forms a subspace of S, say S_c, of dimension k. Any set of k linearly independent vectors in the subspace S_c constitutes a basis. Now consider the set of vectors in S that are orthogonal to every vector in a basis for S_c (and, hence, orthogonal to all vectors in S_c). This set of vectors is also a subspace of S and is called the *null space* of S_c. If the dimension of S_c is k, the dimension of the null space is $n - k$.

Expressed in terms appropriate for binary block codes, the vector space S consists of the 2^n binary valued n-tuples. The linear (n, k) code is a set of 2^k n-tuples called *code words*, which forms a subspace S_c over the field of two elements. Since there are 2^k code words in S_c, a basis for S_c has k code words. That is, k linearly independent code words are required to construct 2^k linear combinations, thus generating the entire code. The null space of S_c is another linear code, which consists of 2^{n-k} code words of block length n and $n - k$ information bits. Its dimension is $n - k$. In Section 8-1-1, we consider these relationships in greater detail.

8-1-1 The Generator Matrix and the Parity Check Matrix

Let $x_{m1}, x_{m2}, \ldots, x_{mk}$ denote the k information bits encoded into the code word $\mathbf{C}_m$. Throughout this chapter, we follow the established convention in coding of representing code words as row vectors. Thus, the vector of k information bits into the encoder is denoted by

$$\mathbf{X}_m = [x_{m1} \quad x_{m2} \quad \ldots \quad x_{mk}]$$

and the output of the encoder is the vector

$$\mathbf{C}_m = [c_{m1} \quad c_{m2} \quad \ldots \quad c_{mn}]$$

The encoding operation performed in a linear binary block encoder can be represented by a set of n equations of the form

$$c_{mj} = x_{m1}g_{1j} + x_{m2}g_{2j} + \ldots + x_{mk}g_{kj}, \qquad j = 1, 2, \ldots, n \tag{8-1-2}$$

where $g_{ij} = 0$ or 1 and $x_{mi}g_{ij}$ represents the product of x_{mi} and g_{ij}. The linear equations (8-1-2) may also be represented in a matrix form as

$$\mathbf{C}_m = \mathbf{X}_m\mathbf{G} \tag{8-1-3}$$

where $\mathbf{G}$, called the *generator matrix* of the code, is

$$\mathbf{G} = \begin{bmatrix} \leftarrow \mathbf{g}_1 \rightarrow \\ \leftarrow \mathbf{g}_2 \rightarrow \\ \vdots \\ \leftarrow \mathbf{g}_k \rightarrow \end{bmatrix} = \begin{bmatrix} g_{11} & g_{12} & \cdots & g_{1n} \\ g_{21} & g_{22} & \cdots & g_{2n} \\ \vdots & \vdots & & \vdots \\ g_{k1} & g_{k2} & \cdots & g_{kn} \end{bmatrix} \tag{8-1-4}$$

Note that any code word is simply a linear combination of the vectors $\{\mathbf{g}_i\}$ of $\mathbf{G}$, i.e.,

$$\mathbf{C}_m = x_{m1}\mathbf{g}_1 + x_{m2}\mathbf{g}_2 + \ldots + x_{mk}\mathbf{g}_k \tag{8-1-5}$$

Since the linear (n, k) code with 2^k code words is a subspace of dimension k, the row vectors $\{\mathbf{g}_i\}$ of the generator matrix $\mathbf{G}$ must be linearly independent, i.e., they must span a subspace of k dimensions. In other words, the $\{\mathbf{g}_i\}$ must be a basis for the (n, k) code. We note that the set of basis vectors is not unique, and, hence, $\mathbf{G}$ is not unique. We also note that, since the subspace has dimension k, the rank of $\mathbf{G}$ is k.

Any generator matrix of an (n, k) code can be reduced by row operations (and column permutations) to the "systematic form."

$$\mathbf{G} = [\mathbf{I}_k \mid \mathbf{P}] = \left[\begin{array}{ccccc:ccc} 1 & 0 & 0 & \ldots & 0 & p_{11} & \cdots & p_{1n-k} \\ 0 & 1 & 0 & \ldots & 0 & p_{21} & \cdots & p_{2n-k} \\ \vdots & \vdots & \vdots & & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \ldots & 1 & p_{k1} & \cdots & p_{kn-k} \end{array}\right] \tag{8-1-6}$$

where $\mathbf{I}_k$ is the $k \times k$ identity matrix and $\mathbf{P}$ is a $k \times (n-k)$ matrix that

determines the $n - k$ redundant bits or parity check bits. Note that a generator matrix of the systematic form generates a linear block code in which the first k bits of each code word are identical to the information bits to be transmitted, and the remaining $n - k$ bits of each code word are linear combinations of the k information bits. These $n - k$ redundant bits are called *parity check bits.* The resulting (n, k) code is called a *systematic code.*

An (n, k) code generated by a generator matrix that is not in the systematic form (8-1-6) is called *nonsystematic.* However, such a generator matrix is equivalent to a generator matrix of the systematic form in the sense that one can be obtained from the other by elementary row operations and column permutations. The two (n, k) linear codes generated by the two equivalent generator matrices are said to be *equivalent*, and one can be obtained from the other by a permutation of the places of every element. Thus, every linear (n, k) code is equivalent to a linear systematic (n, k) code.

Example 8-1-1

Consider a (7, 4) code with generator matrix

$$\mathbf{G} = \left[\begin{array}{cccc:ccc} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{array}\right] = [\mathbf{I}_4 \vdots \mathbf{P}] \tag{8-1-7}$$

A typical code word may be expressed as

$$\mathbf{C}_m = [x_{m1} \quad x_{m2} \quad x_{m3} \quad x_{m4} \quad c_{m5} \quad c_{m6} \quad c_{m7}]$$

where the $\{x_{mj}\}$ represents the four information bits and the $\{c_{mj}\}$ represent the three parity check bits given by

$$\begin{aligned} c_{m5} &= x_{m1} + x_{m2} + x_{m3} \\ c_{m6} &= x_{m2} + x_{m3} + x_{m4} \\ c_{m7} &= x_{m1} + x_{m2} + x_{m4} \end{aligned} \tag{8-1-8}$$

A linear systematic (n, k) binary block encoder may be implemented by using a k-bit shift register and $n - k$ modulo-2 adders tied to the appropriate stages of the shift register. The $n - k$ adders generate the parity check bits, which are subsequently stored temporarily in a second shift register of length $n - k$. The k-bit block of information bits shifted into the k-bit shift register and the $n - k$ parity check bits are computed. Then the k information bits followed by the $n - k$ parity check bits are shifted out of the two shift registers

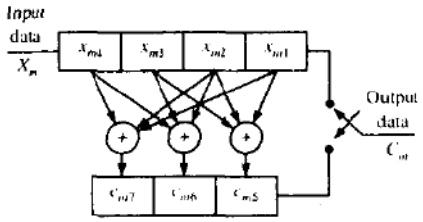


FIGURE 8-1-1 A linear shift register for generating a (7, 4) binary code.

and fed to the modulator. This encoding is illustrated in Fig. 8-1-1 for the (7, 4) code of Example 8-1-1.

Associated with any linear (n, k) code is the dual code of dimension $n - k$. The dual code is a linear $(n, n-k)$ code with 2^{n-k} code vectors, which is the null space of the (n, k) code. The generator matrix for the dual code, denoted by $\mathbf{H}$, consists of $n - k$ linearly independent code vectors selected from the null space. Any code word $\mathbf{C}_m$ of the (n, k) code is orthogonal to any code word in the dual code. Hence, any code word of the (n, k) code is orthogonal to every row of the matrix $\mathbf{H}$, i.e.,

$$\mathbf{C}_m\mathbf{H}' = \mathbf{0} \tag{8-1-9}$$

where $\mathbf{0}$ denotes an all-zero row vector with $n - k$ elements, and $\mathbf{C}_m$ is a code word of the (n, k) code. Since (8-1-9) holds for every code word of the (n, k) code, it follows that

$$\mathbf{G}\mathbf{H}' = \mathbf{0} \tag{8-1-10}$$

where $\mathbf{0}$ is now a $k \times (n - k)$ matrix with all-zero elements.

Now suppose that the linear (n, k) code is systematic and its generator matrix $\mathbf{G}$ is given by the systematic form (8-1-6). Then, since $\mathbf{G}\mathbf{H}' = \mathbf{0}$, it follows that

$$\mathbf{H} = [-\mathbf{P}' \mid \mathbf{I}_{n-k}] \tag{8-1-11}$$

The negative sign in (8-1-11) may be dropped when dealing with binary codes, since modulo-2 subtraction is identical to modulo-2 addition.

Example 8-1-2

For the systematic (7, 4) code generated by matrix $\mathbf{G}$ given by (8-1-7), we have, according to (8-1-11), the matrix $\mathbf{H}$ in the form

$$\mathbf{H} = \left[\begin{array}{cccc|ccc} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{array}\right] \tag{8-1-12}$$

Now, the product $\mathbf{C}_m\mathbf{H}'$ yields the three equations

$$\begin{aligned} x_{m1} + x_{m2} + x_{m3} + c_{m5} &= 0 \\ x_{m2} + x_{m3} + x_{m4} + c_{m6} &= 0 \\ x_{m1} + x_{m2} + x_{m4} + c_{m7} &= 0 \end{aligned} \tag{8-1-13}$$

Thus, we observe that the product $\mathbf{C}_m\mathbf{H}'$ is equivalent to adding the parity check bits to the corresponding linear combinations of the information bits used to compute c_{mj}, $j = 5, 6, 7$. That is, (8-1-13) are equivalent to (8-1-8). The matrix $\mathbf{H}$ may be used by the decoder to check that a received code word $\mathbf{Y}$ satisfies the condition (8-1-13), i.e., $\mathbf{YH}' = \mathbf{0}$. In so doing, the decoder checks the received parity check bits with the corresponding linear combination of the bits y_1, y_2, y_3, and y_4 that formed the parity check bits at the transmitter. It is, therefore, appropriate to call $\mathbf{H}$ the *parity check matrix* associated with the (n, k) code.

We make the following observation regarding the relation of the minimum distance of a code to its parity check matrix $\mathbf{H}$. The product $\mathbf{C}_m\mathbf{H}'$ with $\mathbf{C}_m \neq \mathbf{0}$ represents a linear combination of the n columns of $\mathbf{H}'$. Since $\mathbf{C}_m\mathbf{H}' = \mathbf{0}$, the column vectors of $\mathbf{H}$ are linearly dependent. Suppose $\mathbf{C}_j$ denotes the minimum weight code word of a linear (n, k) code. It must satisfy the condition $\mathbf{C}_j\mathbf{H}' = \mathbf{0}$. Since the minimum weight is equal to the minimum distance, it follows that $d_{\min}$ of the columns of $\mathbf{H}$ are linearly dependent. Alternatively, we may say that no more than $d_{\min} - 1$ columns of $\mathbf{H}$ are linearly independent. Since the rank of $\mathbf{H}$ is at most $n - k$, we have $n - k \geq d_{\min} - 1$. Therefore, $d_{\min}$ is upper-bounded as

$$d_{\min} \leq n - k + \tag{8-1-14}$$

Given a linear binary (n, k) code with minimum distance $d_{\min}$, we can construct a linear binary $(n + 1, k)$ code by appending one additional parity check bit to each code word. The check bit is usually selected to be a check bit on all the bits in the code word. Thus the added check bit is a 0 if the original code word has an even number of 1s and it is a 1 if the code word has an odd number of 1s. Consequently, if the minimum weight and, hence, the minimum distance of the code is odd, the added parity check bit increases the minimum distance by 1. We call the $(n + 1, k)$ code an *extended code*. Its parity check matrix is

$$\boldsymbol{H}_e = \left[\begin{array}{ccc:c} & & & 0 \\ & \mathbf{H} & & 0 \\ & & & \vdots \\ & & & 0 \\ \hdashline 1 & 1 & 1 \ \ldots \ 1 & 1 \end{array}\right] \tag{8-1-15}$$

where $\mathbf{H}$ is the parity check matrix of the original code.

A systematic (n, k) code can also be shortened by setting a number of the information bits to zero. That is, a linear (n, k) code consisting of k information bits and $n-k$ check bits can be shortened into a $(n-l, k-l)$ linear code by setting the first l bits to zero. These l bits are not transmitted. The $n-k$ check bits are computed in the usual manner, as in the original code. Since

$$\mathbf{C}_m = \mathbf{X}_m \mathbf{G}$$

the effect of setting the first l bits of $\mathbf{X}_m$ to 0 is equivalent to reducing the number of rows of $\mathbf{G}$ by removing the first l rows. Equivalently, since

$$\mathbf{C}_m \mathbf{H}' = \mathbf{0}$$

we may remove the first l columns of $\mathbf{H}$. The shortened $(n-l, k-l)$ code consists of 2^{k-l} code words. The minimum distance of these 2^{k-l} code words is at least as large as the minimum distance of the original (n, k) code.

8-1-2 Some Specific Linear Block Codes

In this subsection, we shall briefly describe three types of linear block codes that are frequently encountered in practice and list their important parameters.

Hamming Codes There are both binary and nonbinary Hamming codes. We limit our discussion to the properties of binary Hamming codes. These comprise a class of codes with the property that

$$(n, k) = (2^m - 1, 2^m - 1 - m) \tag{8-1-16}$$

where m is any positive integer. For example, if $m = 3$, we have a (7, 4) code.

The parity check matrix $\mathbf{H}$ of a Hamming code has a special property that allows us to describe the code rather easily. Recall that the parity check matrix of an (n, k) code has $n-k$ rows and n columns. For the binary (n, k) Hamming code, the $n = 2^m - 1$ columns consist of all possible binary vectors with $n-k = m$ elements, except the all-zero vector. For example, the (7, 4) code considered in Examples 8-1-1 and 8-1-2 is a Hamming code. Its parity check matrix consists of the seven column vectors (001), (010), (011), (100), (101), (110), (111).

If we desire to generate a systematic Hamming code, the parity check matrix $\mathbf{H}$ can be easily arranged in the systematic form (8-1-11). Then the corresponding generator matrix $\mathbf{G}$ can be obtained from (8-1-11).

We make the observation that no two columns of $\mathbf{H}$ are linearly dependent, for otherwise the two columns would be identical. However, for $m > 1$, it is possible to find three columns of $\mathbf{H}$ that add to zero. Consequently, $d_{\min} = 3$ for an (n, k) Hamming code.

By adding an overall parity bit, a Hamming (n, k) code can be modified to yield an $(n+1, k)$ code with $d_{\min} = 4$. On the other hand, an (n, k) Hamming code may be shortened to $(n-l, k-l)$ by removing l rows of its generator matrix $\mathbf{G}$ or, equivalently, by removing l columns of its parity check matrix $\mathbf{H}$.

The weight distribution for the class of Hamming (n, k) codes is known and is expressed in compact form by the weight enumerating polynomial

$$A(z) = \sum_{i=0}^{n} A_i z^i = \frac{1}{n+1}[(1+z)^n + n(1+z)^{(n-1)/2}(1-z)^{(n+1)/2}] \tag{8-1-17}$$

where A_i is the number of code words of weight i.

Hadamard Codes A Hadamard code is obtained by selecting as code words the rows of a Hadamard matrix. A Hadamard matrix $\mathbf{M}_n$ is an $n \times n$ matrix (n an even integer) of 1s and 0s with the property that any row differs from any other row in exactly $\frac{1}{2}n$ positions.† One row of the matrix contains all zeros. The other rows contain $\frac{1}{2}n$ zeros and $\frac{1}{2}n$ ones.

For $n = 2$, the Hadamard matrix is

$$\mathbf{M}_2 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \tag{8-1-18}$$

Furthermore, from $\mathbf{M}_n$, we can generate the Hadamard matrix $\mathbf{M}_{2n}$ according to the relation

$$\boldsymbol{M}_{2n} = \begin{bmatrix} \mathbf{M}_n & \mathbf{M}_n \\ \mathbf{M}_n & \bar{\mathbf{M}}_n \end{bmatrix} \tag{8-1-19}$$

where $\bar{\mathbf{M}}_n$ denotes the complement (0s replaced by 1s and vice versa) of $\mathbf{M}_n$. Thus, by substituting (8-1-18) into (8-1-19), we obtain

$$\mathbf{M}_4 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} \tag{8-1-20}$$

The complement of $\mathbf{M}_4$ is

$$\bar{\mathbf{M}}_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix} \tag{8-1-21}$$

Now the rows of $\mathbf{M}_4$ and $\bar{\mathbf{M}}_4$ form a linear binary code of block length $n = 4$ having $2n = 8$ code words. The minimum distance of the code is $d_{\min} = \frac{1}{2}n = 2$.

By repeated application of (8-1-19), we can generate Hadamard codes with block length $n = 2^m$, $k = \log_2 2n = \log_2 2^{m+1} = m + 1$, and $d_{\min} = \frac{1}{2}n = 2^{m-1}$, where m is a positive integer. In addition to the important special case where $n = 2^m$, Hadamard codes of other block lengths are possible, but the codes are not linear.

†Sometimes the elements of the Hadamard matrix are denoted by $+1$ and -1. Then the rows of the Hadamard matrix are mutually orthogonal. We also note that the $M = 2^k$ signal waveforms, constructed from Hadamard code words by mapping each bit in a code word into a binary PSK signal, are orthogonal.

TABLE 8-1-1 WEIGHT DISTRIBUTION OF GOLAY (23, 12) AND EXTENDED GOLAY (24, 12) CODES

	Number of code words	
Weight	**(23, 12) code**	**(24, 12) code**
0	1	1
7	253	0
8	506	759
11	1288	0
12	1288	2576
15	506	0
16	253	759
23	1	0
24	0	1

Source: Peterson and Weldon (1972).

Golay Code The Golay code is a binary linear (23, 12) code with $d_{\min} = 7$. The extended Golay code obtained by adding an overall parity to the (23, 12) is a binary linear (24, 12) code with $d_{\min} = 8$. Table 8-1-1 lists the weight distribution of the code words in the Golay (23, 12) and the extended Golay (24, 12) codes. We discuss the generation of the Golay code in Section 8-1-3.

8-1-3 Cyclic Codes

Cyclic codes are a subset of the class of linear codes that satisfy the following cyclic shift property: if $\mathbf{C} = [c_{n-1}c_{n-2} \ldots c_1c_0]$ is a code word of a cyclic code then $[c_{n-2}c_{n-3} \ldots c_0c_{n-1}]$, obtained by a cyclic shift of the elements of $\mathbf{C}$, is also a code word. That is, all cyclic shifts of $\mathbf{C}$ are code words. As a consequence of the cyclic property, the codes possess a considerable amount of structure which can be exploited in the encoding and decoding operations. A number of efficient encoding and hard-decision decoding algorithms have been devised for cyclic codes that make it possible to implement long block codes with a large number of code words in practical communications systems. A description of specific algorithms is beyond the scope of this book. Our primary objective is to briefly describe a number of characteristics of cyclic codes.

In dealing with cyclic codes, it is convenient to associate with a code word $\mathbf{C} = [c_{n-1}c_{n-2} \ldots c_1c_0]$ a polynomial $C(p)$ of degree $\leq n-1$, defined as

$$C(p) = c_{n-1}p^{n-1} + c_{n-2}p^{n-2} + \ldots + c_1p + c_0 \tag{8-1-22}$$

For a binary code, each of the coefficients of the polynomial is either zero or one.

Now suppose we form the polynomial

$$pC(p) = c_{n-1}p^n + c_{n-2}p^{n-1} + \ldots + c_1p^2 + c_0p$$

This polynomial cannot represent a code word, since its degree may be equal to n (when $c_{n-1} = 1$). However, if we divide $pC(p)$ by $p^n + 1$, we obtain

$$\frac{pC(p)}{p^n + 1} = c_{n-1} + \frac{C_1(p)}{p^n + 1} \tag{8-1-23}$$

where

$$C_1(p) = c_{n-2}p^{n-1} + c_{n-3}p^{n-2} + \ldots + c_0 p + c_{n-1}$$

Note that the polynomial $C_1(p)$ represents the code word $\mathbf{C}_1 = [c_{n-2} \cdots c_0 c_{n-1}]$, which is just the code word $\mathbf{C}$ shifted cyclicly by one position. Since $C_1(p)$ is the remainder obtained by dividing $pC(p)$ by $p^n + 1$, we say that

$$C_1(p) = pC(p) \mod (p^n + 1) \tag{8-1-24}$$

In a similar manner, if $C(p)$ represents a code word in a cyclic code then $p^i C(p) \mod (p^n + 1)$ is also a code word of the cyclic code. Thus we may write

$$p^i C(p) = Q(p)(p^n + 1) + C_i(p) \tag{8-1-25}$$

where the remainder polynomial $C_i(p)$ represents a code word of the cyclic code and $Q(p)$ is the quotient.

We can generate a cyclic code by using a *generator polynomial* $g(p)$ of degree $n - k$. The generator polynomial of an (n, k) cyclic code is a factor of $p^n + 1$ and has the general form

$$g(p) = p^{n-k} + g_{n-k-1}p^{n-k-1} + \ldots + g_1 p + 1 \tag{8-1-26}$$

We also define a *message polynomial* $X(p)$ as

$$X(p) = x_{k-1}p^{k-1} + x_{k-2}p^{k-2} + \ldots + x_1 p + x_0 \tag{8-1-27}$$

where $[x_{k-1}x_{k-2} \cdots x_1 x_0]$ represent the k information bits. Clearly, the product $X(p)g(p)$ is a polynomial of degree less than or equal to $n - 1$, which may represent a code word. We note that there are 2^k polynomials $\{X_i(p)\}$, and, hence, there are 2^k possible code words that can be formed from a given $g(p)$.

Suppose we denote these code words as

$$C_m(p) = X_m(p)g(p), \quad m = 1, 2, \ldots, 2^k \tag{8-1-28}$$

To show that the code words in (8-1-28) satisfy the cyclic property, consider any code word $C(p)$ in (8-1-28). A cyclic shift of $C(p)$ produces

$$C_1(p) = pC(p) + c_{n-1}(p^n + 1) \tag{8-1-29}$$

and, since $g(p)$ divides both $p^n + 1$ and $C(p)$, it also divides $C_1(p)$, i.e., $C_1(p)$ can be represented as

$$C_1(p) = X_1(p)g(p)$$

Therefore, a cyclic shift of any code word $C(p)$ generated by (8-1-28) yields another code word.

From the above, we see that code words possessing the cyclic property can

be generated by multiplying the 2^k message polynomials with a unique polynomial $g(p)$, called the generator polynomial of the (n, k) cyclic code, which divides $p^n + 1$ and has degree $n - k$. The cyclic code generated in this manner is a subspace S_c of the vector space S. The dimension of S_c is k.

Example 8-1-3

Consider a code with block length $n = 7$. The polynomial $p^7 + 1$ has the following factors:

$$p^7 + 1 = (p + 1)(p^3 + p^2 + 1)(p^3 + p + 1) \tag{8-1-30}$$

To generate a (7, 4) cyclic code, we may take as a generator polynomial one of the following two polynomials:

$$\begin{aligned} g_1(p) &= p^3 + p^2 + 1 \\ g_2(p) &= p^3 + p + 1 \end{aligned} \tag{8-1-31}$$

The codes generated by $g_1(p)$ and $g_2(p)$ are equivalent. The code words in the (7, 4) code generated by $g_1(p) = p^3 + p^2 + 1$ are given in Table 8-1-2.

In general, the polynomial $p^n + 1$ may be factored as

$$p^n + 1 = g(p)h(p)$$

where $g(p)$ denotes the generator polynomial for the (n, k) cyclic code and

TABLE 8-1-2 (7, 4) CYCLIC CODE
Generator Polynomial: $g_1(p) = p^3 + p^2 + 1$

Information bits				Code words						
p^3	p^2	p^1	p^0	p^6	p^5	p^4	p^3	p^2	p^1	p^0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1	1	0	1
0	0	1	0	0	0	1	1	0	1	0
0	0	1	1	0	0	1	0	1	1	1
0	1	0	0	0	1	1	0	1	0	0
0	1	0	1	0	1	1	1	0	0	1
0	1	1	0	0	1	0	1	1	1	0
0	1	1	1	0	1	0	0	0	1	1
1	0	0	0	1	1	0	1	0	0	0
1	0	0	1	1	1	0	0	1	0	1
1	0	1	0	1	1	1	0	0	1	0
1	0	1	1	1	1	1	1	1	1	1
1	1	0	0	1	0	1	1	1	0	0
1	1	0	1	1	0	1	0	0	0	1
1	1	1	0	1	0	0	0	1	1	0
1	1	1	1	1	0	0	1	0	1	1

$h(p)$ denotes the *parity polynomial* that has degree k. The latter may be used to generate the dual code.

For this purpose, we define the *reciprocal polynomial* of $h(p)$ as

$$\begin{aligned} p^k h(p^{-1}) &= p^k(p^{-k} + h_{k-1}p^{-k+1} + h_{k-2}p^{-k+2} + \ldots + h_1 p^{-1} + 1) \\ &= 1 + h_{k-1}p + h_{k-2}p^2 + \ldots + h_1 p^{k-1} + p^k \end{aligned} \tag{8-1-32}$$

Clearly, the reciprocal polynomial is also a factor of $p^n + 1$. Hence, $p^k h(p^{-1})$ is the generator polynomial of an $(n, n-k)$ cyclic code. This cyclic code is the dual code to the (n, k) code generated from $g(p)$. Thus, the $(n, n-k)$ dual code constitutes the null space of the (n, k) cyclic code.

Example 8-1-4

Let us consider the dual code to the (7, 4) cyclic code generated in Example 8-1-3. This dual code is a (7, 3) cyclic code associated with the parity polynomial

$$\begin{aligned} h_1(p) &= (p+1)(p^3 + p + 1) \\ &= p^4 + p^3 + p^2 + 1 \end{aligned} \tag{8-1-33}$$

The reciprocal polynomial is

$$p^4 h_1(p^{-1}) = 1 + p + p^2 + p^4$$

This polynomial generates the (7, 3) dual code given in Table 8-1-3. The reader can verify that the code words in the (7, 3) dual code are orthogonal to the code words in the (7, 4) cyclic code of Example 8-1-3. Note that neither the (7, 4) nor the (7, 3) codes are systematic.

It is desirable to show how a generator matrix can be obtained from the generator polynomial of a cyclic (n, k) code. As previously indicated, the generator matrix for an (n, k) code can be constructed from any set of k

TABLE 8-1-3 (7, 3) DUAL CODE
Generator Polynomial $p^4 h_1(p^{-1}) = p^4 + p^2 + p + 1$

Information bits			Code words						
p^2	p^1	p^0	p^6	p^5	p^4	p^3	p^2	p^1	p^0
0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	1	0	1	1	1
0	1	0	0	1	0	1	1	1	0
0	1	1	0	1	1	1	0	0	1
1	0	0	1	0	1	1	1	0	0
1	0	1	1	0	0	1	0	1	1
1	1	0	1	1	1	0	0	1	0
1	1	1	1	1	0	0	1	0	1

linearly independent code words. Hence, given the generator polynomial $g(p)$, an easily generated set of k linearly independent code words is the code words corresponding to the set of k linearly independent polynomials

$$p^{k-1}g(p), \quad p^{k-2}g(p), \quad \ldots, \quad pg(p), \quad g(p)$$

Since any polynomial of degree less than or equal to $n-1$ and divisible by $g(p)$ can be expressed as a linear combination of this set of polynomials, the set forms a basis of dimension k. Consequently, the code words associated with these polynomials form a basis of dimension k for the (n, k) cyclic code.

Example 8-1-5

The four rows of the generator matrix for the (7, 4) cyclic code with generator polynomial $g_1(p) = p^3 + p^2 + 1$ are obtained from the polynomials

$$p^i g_1(p) = p^{3+i} + p^{2+i} + p^i, \quad i = 3, 2, 1, 0$$

It is easy to see that the generator matrix is

$$\mathbf{G}_1 = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix} \tag{8-1-34}$$

Similarly, the generator matrix for the (7, 4) cyclic code generated by the polynomial $g_2(p) = p^3 + p + 1$ is

$$\mathbf{G}_2 = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} \tag{8-1-35}$$

The parity check matrices corresponding to $\mathbf{G}_1$ and $\mathbf{G}_2$ can be constructed in the same manner by using the respective reciprocal polynomials (Problem 8-8).

Note that the generator matrix obtained by this construction is not in systematic form. We can construct the generator matrix of a cyclic code in the systematic form $\mathbf{G} = [\mathbf{I}_k \mid \mathbf{P}]$ from the generator polynomial as follows. First, we observe that the lth row of $\mathbf{G}$ corresponds to a polynomial of the form $p^{n-l} + R_l(p)$, $l = 1, 2, \ldots, k$, where $R_l(p)$ is a polynomial of degree less than $n-k$. This form can be obtained by dividing p^{n-l} by $g(p)$. Thus, we have

$$\frac{p^{n-l}}{g(p)} = Q_l(p) + \frac{R_l(p)}{g(p)}, \quad l = 1, 2, \ldots, k$$

or, equivalently,

$$p^{n-l} = Q_l(p)g(p) + R_l(p), \quad l = 1, 2, \ldots, k \tag{8-1-36}$$

where $Q_l(p)$ is the quotient. But $p^{n-l} + R_l(p)$ is a code word of the cyclic code since $p^{n-l} + R_l(p) = Q_l(p)g(p)$. Therefore the desired polynomial corresponding to the lth row of $\mathbf{G}$ is $p^{n-l} + R_l(p)$.

Example 8-1-6

For the (7, 4) cyclic code with generator polynomial $g_2(p) = p^3 + p + 1$, previously discussed in Example 8-1-5, we have

$$p^6 = (p^3 + p + 1)g_2(p) + p^2 + 1$$
$$p^5 = (p^2 + 1)g_2(p) + p^2 + p + 1$$
$$p^4 = pg_2(p) + p^2 + p$$
$$p^3 = g_2(p) + p + 1$$

Hence, the generator matrix of the code in systematic form is

$$\mathbf{G}_2 = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} \tag{8-1-37}$$

and the corresponding parity check matrix is

$$\mathbf{H}_2 = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix} \tag{8-1-38}$$

It is left as an exercise for the reader to demonstrate that the generator matrix $\mathbf{G}_2$ given by (8-1-35) and the systematic form given by (8-1-37) generate the same set of code words (Problem 8-2).

The method for constructing the generator matrix $\mathbf{G}$ in systematic form according to (8-1-36) also implies that a systematic code can be generated directly from the generator polynomial $g(p)$. Suppose that we multiply the message polynomial $X(p)$ by p^{n-k}. Thus, we obtain

$$p^{n-k}X(p) = x_{k-1}p^{n-1} + x_{k-2}p^{n-2} + \ldots + x_1p^{n-k+1} + x_0p^{n-k}$$

In a systematic code, this polynomial represents the first k bits in the code word $C(p)$. To this polynomial we must add a polynomial of degree less than $n - k$ representing the parity check bits. Now, if $p^{n-k}X(p)$ is divided by $g(p)$, the result is

$$\frac{p^{n-k}X(p)}{g(p)} = Q(p) + \frac{r(p)}{g(p)}$$

or, equivalently,

$$p^{n-k}X(p) = Q(p)g(p) + r(p) \tag{8-1-39}$$

where $r(p)$ has degree less than $n-k$. Clearly, $Q(p)g(p)$ is a code word of the cyclic code. Hence, by adding (modulo-2) $r(p)$ to both sides of (8-1-39), we obtain the desired systematic code.

To summarize, the systematic code may be generated by

1 multiplying the message polynomial $X(p)$ by p^{n-k};
2 dividing $p^{n-k}X(p)$ by $g(p)$ to obtain the remainder $r(p)$; and
3 adding $r(p)$ to $p^{n-k}X(p)$.

Below we demonstrate how these computations can be performed by using shift registers with feedback.

Since $p^n + 1 = g(p)h(p)$ or, equivalently, $g(p)h(p) = 0 \mod (p^n + 1)$, we say that the polynomials $g(p)$ and $h(p)$ are *orthogonal*. Furthermore, the polynomials $p^i g(p)$ and $p^j h(p)$ are also orthogonal for all i and j. However, the vectors corresponding to the polynomials $g(p)$ and $h(p)$ are orthogonal only if the ordered elements of one of these vectors are reversed. The same statement applies to the vectors corresponding to $p^i g(p)$ and $p^j h(p)$. In fact, if the parity polynomial $h(p)$ is used as a generator for the $(n, n-k)$ dual code, the set of code words obtained just comprises the same code words generated by the reciprocal polynomial except that the code vectors are reversed. This implies that the generator matrix for the dual code obtained from the reciprocal polynomial $p^k h(p^{-1})$ can also be obtained indirectly from $h(p)$. Since the parity check matrix $\mathbf{H}$ for the (n, k) cyclic code is the generator matrix for the dual code, it follows that $\mathbf{H}$ can also be obtained from $h(p)$. The following example illustrates these relationships.

Example 8-1-7

The dual code to the (7, 4) cyclic code generated by $g_1(p) = p^3 + p^2 + 1$ is the (7, 3) dual code that is generated by the reciprocal polynomial $p^4 h_1(p^{-1}) = p^4 + p^2 + p + 1$. However, we may also use $h_1(p)$ to obtain the generator matrix for the dual code. Then, the matrix corresponding to the polynomials $p^i h_1(p)$, $i = 2, 1, 0$, is

$$\mathbf{G}_{h1} = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 \end{bmatrix}$$

The generator matrix for the (7, 3) dual code, which is the parity check matrix for the (7, 4) cyclic code, consists of the rows of $\mathbf{G}_{h1}$ taken in reverse order. Thus,

$$\mathbf{H}_1 = \begin{bmatrix} 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 \end{bmatrix}$$

The reader may verify that $\mathbf{G}_1\mathbf{H}_1' = 0$.

Note that the column vectors of $\mathbf{H}_1$ consist of all seven binary vectors of length 3, except the all-zero vector. But this is just the description of the parity check matrix for a (7, 4) Hamming code. Therefore, the (7, 4) cyclic code is equivalent to the (7, 4) Hamming code discussed previously in Examples 8-1-1 and 8-1-2.

Encoders for Cyclic Codes The encoding operations for generating a cyclic code may be performed by a linear feedback shift register based on the use of either the generator polynomial or the parity polynomial. First, let us consider the use of $g(p)$.

As indicated above, the generation of a systematic cyclic code involves three steps, namely multiplying the message polynomial $X(p)$ by p^{n-k}, dividing the product by $g(p)$, and, finally, adding the remainder to $p^{n-k}X(p)$. Of these three steps, only the division is nontrivial.

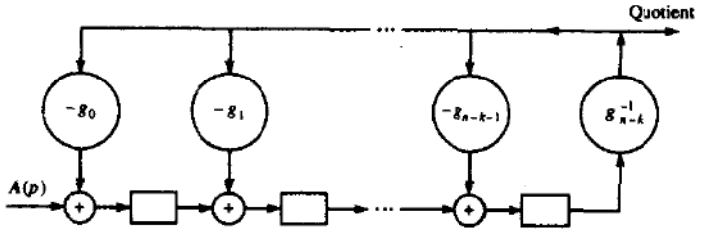
The division of the polynomial $A(p) = p^{n-k}X(p)$ of degree $n-1$ by the polynomial

$$g(p) = g_{n-k}p^{n-k} + g_{n-k-1}p^{n-k-1} + \ldots + g_1 p + g_0$$

may be accomplished by the $(n-k)$ stage feedback shift register illustrated in Fig. 8-1-2. Initially, the shift register contains all zeros. The coefficients of $A(p)$ are clocked into the shift register one (bit) coefficient at a time, beginning with the higher-order coefficients, i.e., with a_{n-1}, followed by a_{n-2}, and so on. After the kth shift, the first nonzero output of the quotient is $q_1 = g_{n-k}a_n$. Subsequent outputs are generated as illustrated in Fig. 8-1-2. For each output coefficient in the quotient, we must subtract the polynomial $g(p)$ multiplied by that coefficient, as in ordinary long division. This subtraction is performed by means of the feedback part of the shift register. Thus, the feedback shift register in Fig. 8-1-2 performs division of two polynomials.

In our case, $g_{n-k} = g_0 = 1$, and, for binary codes, the arithmetic operations are performed in modulo-2 arithmetic. Consequently, the subtraction operations reduce to modulo-2 addition. Furthermore, we are only interested in

FIGURE 8-1-2 A feedback shift register for dividing the polynomial $A(p)$ by $g(p)$.



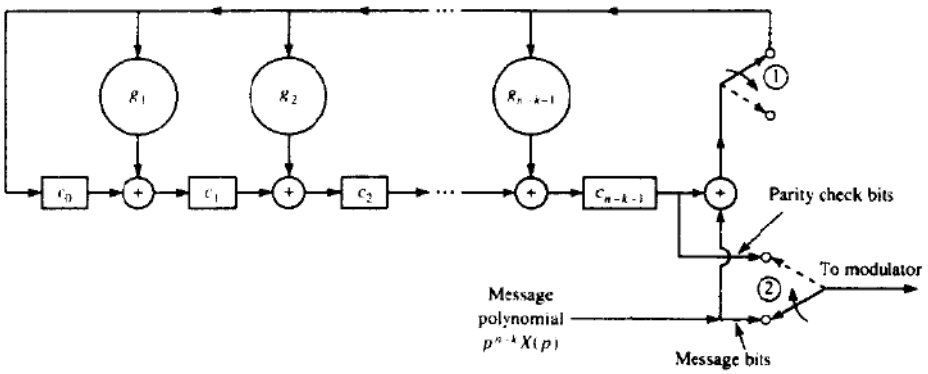


FIGURE 8-1-3 Encoding of a cyclic code by use of the generator polynomial $g(p)$.

generating the parity check bits for each code word, since the code is systematic. Consequently, the encoder for the cyclic code takes the form illustrated in Fig. 8-1-3. The first k bits at the output of the encoder are simply the k information bits. These k bits are also clocked simultaneously into the shift register, since the switch 1 is in the closed position. Note that the polynomial multiplication of p^{n-k} with $X(p)$ is not performed explicitly. After the k information bits are all clocked into the encoder, the positions of the two switches are reversed. At this time, the contents of the shift register are simply the $n-k$ parity check bits, which correspond to the coefficients of the remainder polynomial. These $n-k$ bits are clocked out one at a time and sent to the modulator.

Example 8-1-8

The shift register for encoding the (7, 4) cyclic code with generator polynomial $g(p) = p^3 + p + 1$ is illustrated in Fig. 8-1-4. Suppose the input

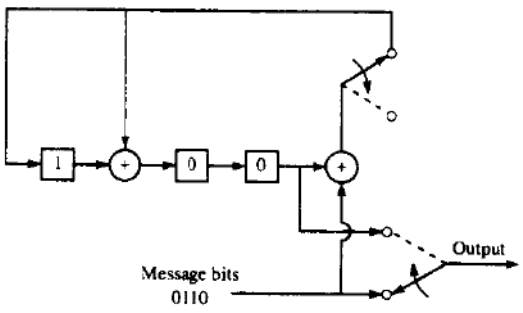


FIGURE 8-1-4 The encoder for the (7, 4) cyclic code with generator polynomial $g(p) = p^3 + p + 1$.

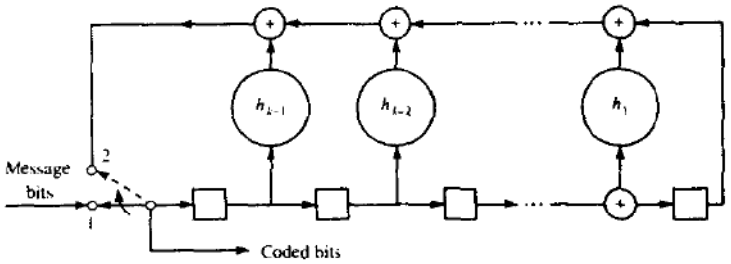


FIGURE 8-1-5 Encoder for an (n, k) cyclic code based on parity polynomial $h(p)$.

message bits are 0110. The contents of the shift register are as follows:

Input	Shift	Shift register contents
	0	0 0 0
0	1	0 0 0
1	2	1 1 0
1	3	1 0 1
0	4	1 0 0

Hence, the three parity check bits are 100, which correspond to the code bits $c_5 = 0$, $c_6 = 0$, and $c_7 = 1$.

Instead of using the generator polynomial, we may implement the encoder for the cyclic code by making use of the parity polynomial

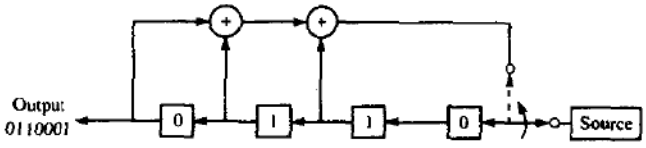
$$h(p) = p^k + h_{k-1}p^{k-1} + \ldots + h_1 p + 1$$

The encoder is shown in Fig. 8-1-5. Initially, the k information bits are shifted into the shift register and simultaneously fed to the modulator. After all k information bits are in the shift register, the switch is thrown into position 2 and the shift register is clocked $n - k$ times to generate the $n - k$ parity check bits as illustrated in Fig. 8-1-5.

Example 8-1-9

The parity polynomial for the $(7, 4)$ cyclic code generated by $g(p) = p^3 + p + 1$ is $h(p) = p^4 + p^2 + p + 1$. The encoder for this code based on the parity polynomial is illustrated in Fig. 8-1-6. If the input to the encoder is

FIGURE 8-1-6 The encoder for the $(7, 4)$ cyclic code based on the parity polynomial $h(p) = p^4 + p^2 + 1$.



the message bits 0110, the parity check bits are $c_5 = 0$, $c_6 = 0$, and $c_7 = 1$, as is easily verified.

It should be noted that the encoder based on the generator polynomial is simpler when $n - k < k$ $(k > \frac{1}{2}n)$, i.e., for high rate codes $(R_c > \frac{1}{2})$, while the encoder based on the parity polynomial is simpler when $k < n - k$ $(k < \frac{1}{2}n)$, which corresponds to low rate codes $(R_c < \frac{1}{2})$.

Cyclic Hamming Codes The class of cyclic codes include the Hamming codes, which have a block length $n = 2^m - 1$ and $n - k = m$ parity check bits, where m is any positive integer. The cyclic Hamming codes are equivalent to the Hamming codes described in Section 8-1-2.

Cyclic (23, 12) Golay Code The linear (23, 12) Golay code described in Section 8-1-2 can be generated as a cyclic code by means of the generator polynomial

$$g(p) = p^{11} + p^9 + p^7 + p^6 + p^5 + p + 1 \tag{8-1-40}$$

The code words have a minimum distance $d_{\min} = 7$.

Maximum-Length Shift-Register Codes Maximum-length shift-register codes are a class of cyclic codes with

$$(n, k) = (2^m - 1, m) \tag{8-1-41}$$

where m is a positive integer. The code words are usually generated by means of an m-stage digital shift register with feedback, based on the parity polynomial. For each code word to be transmitted, the m information bits are loaded into the shift register, and the switch is thrown from position 1 to position 2. The contents of the shift register are shifted to the left one bit at a time for a total of $2^m - 1$ shifts. This operation generates a systematic code with the desired output length $n = 2^m - 1$. For example, the code words generated by the $m = 3$ stage shift register in Fig. 8-1-7 are listed in Table 8-1-4.

Note that, with the exception of the all-zero code word, all the code words generated by the shift register are different cyclic shifts of a single code word. The reason for this structure is easily seen from the state diagram of the shift register, which is illustrated in Fig. 8-1-8 for $m = 3$. When the shift register is loaded initially and shifted $2^m - 1$ times, it will cycle through all possible $2^m - 1$ states. Hence, the shift resgister is back to its original state in $2^m - 1$ shifts.

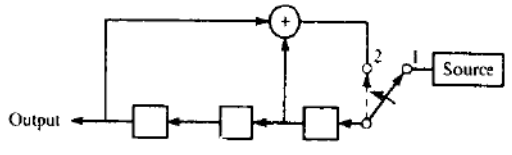


FIGURE 8-1-7 Three-stage ($m = 3$) shift register with feedback.

TABLE 8-1-4 MAXIMUM-LENGTH SHIFT-REGISTER CODE FOR $m = 3$

Information bits			Code words						
0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	1	1	1	0	1
0	1	0	0	1	0	0	1	1	1
0	1	1	0	1	1	1	0	1	0
1	0	0	1	0	0	1	1	1	0
1	0	1	1	0	1	0	0	1	1
1	1	0	1	1	0	1	0	0	1
1	1	1	1	1	1	0	1	0	0

Consequently, the output sequence is periodic with length $n = 2^m - 1$. Since there are $2^m - 1$ possible states, this length corresponds to the largest possible period. This explains why the $2^m - 1$ code words are different cyclic shifts of a single code word.

Maximum-length shift-register codes exist for any positive value of m.

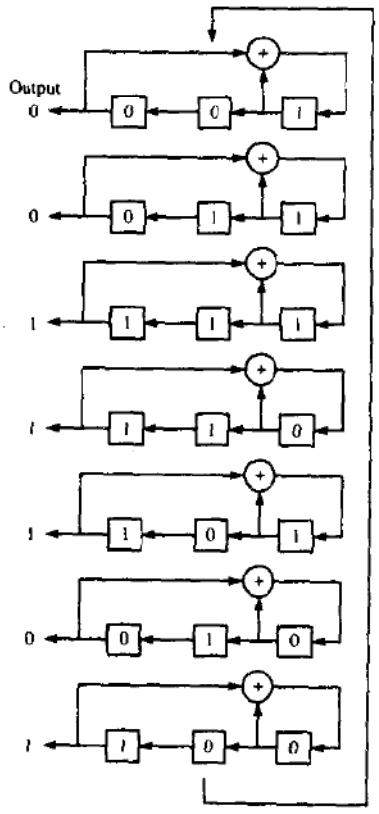


FIGURE 8-1-8 The seven states for the $m = 3$ maximum length shift register.

TABLE 8-1-5 SHIFT-REGISTER CONNECTIONS FOR GENERATING MAXIMUM-LENGTH SEQUENCES

m	Stages connected to modulo-2 adder	m	Stages connected to modulo-2 adder	m	Stages connected to modulo-2 adder
2	1, 2	13	1, 10, 11, 13	24	1, 18, 23, 24
3	1, 3	14	1, 5, 9, 14	25	1, 23
4	1, 4	15	1, 15	26	1, 21, 25, 26
5	1, 4	16	1, 5, 14, 16	27	1, 23, 26, 27
6	1, 6	17	1, 15	28	1, 26
7	1, 7	18	1, 12	29	1, 28
8	1, 5, 6, 7	19	1, 15, 18, 19	30	1, 8, 29, 30
9	1, 6	20	1, 18	31	1, 29
10	1, 8	21	1, 20	32	1, 11, 31, 32
11	1, 10	22	1, 22	33	1, 21
12	1, 7, 9, 12	23	1, 19	34	1, 8, 33, 34

Source: Forney (1970).

Table 8-1-5 lists the stages connected to the modulo-2 adder that result in a maximum-length shift register for $2 \leq m \leq 34$.

Another characteristic of the code words in a maximum-length shift-register code is that each code word, with the exception of the all-zero code word, contains 2^{m-1} ones and 2^{m-1} zeros. Hence all these code words have identical weights, namely, $w = 2^{m-1}$. Since the code is linear, this weight is also the minimum distance of the code, i.e.,

$$d_{\min} = 2^{m-1}$$

Finally, note that the (7, 3) maximum-length shift-register code shown in Table 8-1-4 is identical to the (7, 3) code given in Table 8-1-3, which is the dual of the (7, 4) Hamming code given in Table 8-1-2. This is not a coincidence. The maximum-length shift-register codes are the dual codes of the cyclic Hamming $(2^m - 1, 2^m - 1 - m)$ codes.

The shift register for generating the maximum-length code may also be used to generate a periodic binary sequence with period $n = 2^m - 1$. The binary periodic sequence exhibits a periodic autocorrelation $\phi(m)$ with values $\phi(m) = n$ for $m = 0, \pm n, \pm 2n, \ldots$, and $\phi(m) = -1$ for all other shifts as described in Section 13-2-4. This impulse-like autocorrelation implies that the power spectrum is nearly white and, hence, the sequence resembles white noise. As a consequence, maximum-length sequences are called pseudo-noise (PN) sequences and find use in the scrambling of data and in the generation of spread spectrum signals.

Bose–Chaudhuri–Hocquenghem (BCH) Codes BCH codes comprise a large class of cyclic codes that include both binary and nonbinary alphabets.

Binary BCH codes may be constructed with parameters

$$\begin{aligned} n &= 2^m - 1 \\ n - k &\leq mt \\ d_{\min} &= 2t + 1 \end{aligned} \tag{8-1-42}$$

where m ($m \geq 3$) and t are arbitrary positive integers. Hence, this class of binary codes provides the communications system designer with a large selection of block lengths and code rates. Nonbinary BCH codes include the powerful Reed–Solomon codes that are described later.

The generator polynomials for BCH codes can be constructed from factors of $p^{2^m-1} + 1$. Table 8-1-6 lists the coefficients of generator polynomials for BCH codes of block lengths $7 \leq n \leq 255$, corresponding to $3 \leq m \leq 8$. The coefficients are given in octal form, with the left-most digit corresponding to the highest-degree term of the generator polynomial. Thus, the coefficients of the generator polynomial for the (15, 5) code are 2467, which in binary form is 10 100 110 111. Consequently, the generator polynomial is $g(p) = p^{10} + p^8 + p^5 + p^4 + p^2 + p + 1$.

A more extensive list of generator polynomials for BCH codes is given by Peterson and Weldon (1972), who tabulate the polynomial factors of $p^{2^m-1} + 1$ for $m \leq 34$.

8-1-4 Optimum Soft-Decision Decoding of Linear Block Codes

In this subsection, we derive the performance of linear binary block codes on an AWGN channel when optimum (unquantized) soft-decision decoding is employed at the receiver. The bits of a code word may be transmitted by any one of the binary signaling methods described in Chapter 5. For our purposes, we consider binary (or quaternary) coherent PSK, which is the most efficient method, and binary orthogonal FSK either with coherent detection or noncoherent detection.

Let $\mathscr{E}$ denote the transmitted signal energy per code word and let $\mathscr{E}_c$ denote the signal energy required to transmit a single element (bit) in the code word. Since there are n bits per code word, $\mathscr{E} = n\mathscr{E}_c$, and since each code word conveys k bits of information, the energy per information bit is

$$\mathscr{E}_b = \frac{\mathscr{E}}{k} = \frac{n}{k}\mathscr{E}_c = \frac{\mathscr{E}_c}{R_c} \tag{8-1-43}$$

The code words are assumed to be equally likely a priori with prior probability $1/M$.

Suppose the bits of a code word are transmitted by binary PSK. Thus each code word results in one of M signaling waveforms. From Chapter 5, we know that the optimum receiver, in the sense of minimizing the average probability

TABLE 8-1-6 COEFFICIENTS OF GENERATOR POLYNOMIALS (IN OCTAL FORM) FOR BCH CODES OF LENGTHS $7 \leq n \leq 255$

n	k	t	$g(p)$
7	4	1	13
15	11	1	23
	7	2	721
	5	3	2467
31	26	1	45
	21	2	3551
	16	3	107657
	11	5	5423325
	6	7	313365047
63	57	1	103
	51	2	12471
	45	3	1701317
	39	4	166623567
	36	5	1033500423
	30	6	157464165547
	24	7	17323260404441
	18	10	1363026512351725
	16	11	6331141367235453
	10	13	472622305527250155
	7	15	5231045543503271737
127	120	1	211
	113	2	41567
	106	3	11554743
	99	4	3447023271
	92	5	624730022327
	85	6	130704476322273
	78	7	26230002166130115
	71	9	6255010713253127753
	64	10	1206534025570773100045
	57	11	335265252505705053517721
	50	13	54446512523314012421501421
	43	14	177217722136512275212205743 43
	36	15	314607466652207504476457472173 5
	29	21	403114461367670603667530141176155
	22	23	123376070404722522435445626637647043
	15	27	220570424456045547705230137622176043 53
	8	31	704726405275103065147622427156773313021 7
255	247	1	435
	239	2	267543
	231	3	156720665
	223	4	75626641375
	215	5	23157564726421
	207	6	16176560567636227
	199	7	7633031270420722341
	191	8	2663470176115333714567
	187	9	527553135400013222363 51
	179	10	226247107173404324163004 55
	171	11	154162142123423560770616306 7

TABLE 8-1-6 *(Continued)*

n	k	t	$g(p)$
	163	12	750041551007560255157472451460 1
	155	13	375751300540766501572250646467763 3
	147	14	164213017353716552530416530544101171 1
	139	15	461401732060175561570722730247453567445
	131	18	215713331471510151261250277442142024165471
	123	19	120614052242066003717210326516141226272506267
	115	21	605266655721002472636364046002763525563136472737
	107	22	22205772322066256312417300235347420176574750154441
	99	23	10656667253473174222741416201574332252411076432303431
	91	25	67502650303274441727236317247325110755507627207243445 61
	87	26	11013676341474323643523163430717204620672254527331172131 7
	79	27	66700035637657500020270344207366174621015326711766541342355
	71	29	24024710520644321515554172112331163205444250362557643221706035
	63	30	10754475055163544325315217357707003666111726455267613656702543301
	55	31	73154252035011001330152753060320543254143267550105570444260354736 17
	47	42	25335420170626465630330413774062331751233341454460450050660245525431 73
	45	43	15202056055234161131101346376423701563670024470762373033202157025051541
	37	45	51363302550670074141774472454375304207357061743234323476443547374030440 03
	29	47	30257155366730714655270640123613771153422423242011741140602547574104035650 37
	21	55	12562152570603326560017731536076121032273414056530745425211531216144665134 73725
	13	59	46417320050525645444265737142500660043306774454765614031746772135702613446050 0547
	9	63	15726025217472463201031043255355134614162367212044074545112766115547705561677516057

Source: Stenbit (1964), © 1964 IEEE.

of a code word error, for the AWGN channel, can be realized as a parallel bank of M filters matched to the M possible transmitted waveforms. The outputs of the M matched filters at the end of each signaling interval, which encompasses the transmission of n bits in the code word, are compared and the code word corresponding to the largest matched filter output is selected. Alternatively, M cross-correlators can be employed. In either case, the receiver implementation can be simplified. That is, an equivalent optimum receiver can be realized by use of a single filter (or cross-correlator) matched to the binary PSK waveform used to transmit each bit in the code word, followed by a decoder that forms the M decision variables corresponding to the M code words.

To be specific, let r_j, $j = 1, 2, \ldots, n$, represent the n sampled outputs of the matched filter for any particular code word. Since the signaling is binary coherent PSK, the output r_j may be expressed either as

$$r_j = \sqrt{\mathscr{E}_c} + n_j \tag{8-1-44}$$

when the jth bit of a code word is a 1, or as

$$r_j = -\sqrt{\mathscr{E}_c} + n_j \tag{8-1-45}$$

when the jth bit is a 0. The variables $\{n_j\}$ represent additive white gaussian noise at the sampling instants. Each n_j has zero mean and variance $\frac{1}{2}N_0$. From knowledge of the M possible transmitted code words and upon reception of $\{r_j\}$, the optimum decoder forms the M correlation metrics

$$CM_i = C(\mathbf{r}, \mathbf{C}_i) = \sum_{j=1}^{n} (2c_{ij} - 1)r_j, \qquad i = 1, 2, \ldots, M \tag{8-1-46}$$

where c_{ij} denotes the bit in the jth position of the ith code word. Thus, if $c_{ij} = 1$, the weighting factor $2c_{ij} - 1 = 1$, and if $c_{ij} = 0$, the weighting factor $2c_{ij} - 1 = -1$. In this manner, the weighting $2c_{ij} - 1$ aligns the signal components in $\{r_j\}$ such that the correlation metric corresponding to the actual transmitted code word will have a mean value $\sqrt{\mathscr{E}_c}n$, while the other $M - 1$ metrics will have smaller mean values.

Although the computations involved in forming the correlation metrics for soft-decision decoding according to (8-1-46) are relatively simple, it may still be impractical to compute (8-1-46) for all the possible code words when the number of code words is large, e.g., $M > 2^{10}$. In such a case it is still possible to implement soft-decision decoding using algorithms which employ techniques for discarding improbable code words without computing their entire correlation metrics as given by (8-1-46). Several different types of soft-decision decoding algorithms have been described in the technical literature. The interested reader is referred to the papers by Forney (1966b), Weldon (1971), Chase (1972), Wainberg and Wolf (1973), Wolf (1978), and Matis and Modestino (1982).

In determining the probability of error for a linear block code, note that

when such a code is employed on a binary-input, symmetric channel such as the AWGN channel with optimum soft-decision decoding, the error probability for the transmission of the mth code word is the same for all m. Hence, we assume for simplicity that the all-zero code word $\mathbf{C}_1$ is transmitted. For correct decoding of $\mathbf{C}_1$, the correlation metric CM_1 must exceed all the other $M-1$ correlation metrics CM_m, $m=2,\ldots,M$. All the CM are gaussian distributed. The mean value of CM_1 is $\sqrt{\mathscr{E}_c}n$, while the mean values of CM_m, $m=2,\ldots,M$ is $\sqrt{\mathscr{E}_c}n(1-2w_m/n)$. The variance of each decision variable is $\frac{1}{2}N_0$. The derivation of the exact expression for the probability of correct decoding or, equivalently, the probability of a code word error is complicated by the correlations among the M correlation metrics. The cross-correlation coefficients between $\mathbf{C}_1$ and the other $M-1$ code words are

$$\rho_m = 1 - 2w_m/n, \quad m=2,\ldots,M \tag{8-1-47}$$

where w_m denotes the weight of the mth code word.

Instead of attempting to derive the exact error probability, we resort to a union bound. The probability that $CM_m > CM_1$ is

$$P_2(m) = Q\left(\sqrt{\frac{\mathscr{E}}{N_0}(1-\rho_m)}\right) \tag{8-1-48}$$

where $\mathscr{E} = k\mathscr{E}_b$ is the transmitted energy per waveform. Substitution for ρ_m from (8-1-47) and for $\mathscr{E}$ yields

$$\begin{aligned} P_2(m) &= Q\left(\sqrt{\frac{2\mathscr{E}_b}{N_0}R_c w_m}\right) \\ &= Q(\sqrt{2\gamma_b R_c w_m}) \end{aligned} \tag{8-1-49}$$

where γ_b is the SNR per bit and R_c is the code rate. Then the average probability of a code word error is bounded from above by the sum of the binary error events given by (8-1-49). Thus,

$$\begin{aligned} P_M &\leq \sum_{m=2}^{M} P_2(m) \\ &\leq \sum_{m=2}^{M} Q(\sqrt{2\gamma_b R_c w_m}) \end{aligned} \tag{8-1-50}$$

The computation of the probability of error for soft-decision decoding according to (8-1-50) requires knowledge of the weight distribution of the code. Weight distributions of many codes are given in a number of texts on coding theory, e.g., Berlekamp (1968) and MacWilliams and Sloane (1977).

A somewhat looser bound is obtained by noting that

$$Q(\sqrt{2\gamma_b R_c w_m}) \leq Q(\sqrt{2\gamma_b R_c d_{\min}}) < \exp(-\gamma_b R_c d_{\min}) \tag{8-1-51}$$

Consequently,

$$P_M \leq (M-1)Q(\sqrt{2\gamma_b R_c d_{\min}}) < \exp(-\gamma_b R_c d_{\min} + k \ln 2) \tag{8-1-52}$$

This bound is particularly useful since it does not require knowledge of the weight distribution of the code. When the upper bound in (8-1-52) is compared with the performance of an uncoded binary PSK system, which is upper-bounded as $\frac{1}{2}\exp(-\gamma_b)$, we find that coding yields a gain of approximately $10\log(R_c d_{\min} - k\ln 2/\gamma_b)$ dB. We may call this the *coding gain*. We note that its value depends on the code parameters and also on the SNR per bit γ_b.

The expression for the probability of error for equicorrelated waveforms that can be obtained for the simplex signals described in Section 5-2 gives us yet a third approximation to the error probabilities for coded waveforms. We know that the maximum cross-correlation coefficient between a pair of coded waveforms is

$$\rho_{\max} = 1 - \frac{2}{n} d_{\min} \tag{8-1-53}$$

If we assume as a worst case that all the M code words have a cross-correlation coefficient equal to $\rho_{\max}$ then the code word error probability can easily be manipulated. Since some code words are separated by more than the minimum distance, the error probability evaluated for $\rho_r = \rho_{\max}$ is actually an upper bound. Thus,

$$P_M \leq 1 - \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-v^2/2} \left(\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{v + \sqrt{4\gamma_b R_c d_{\min}}} e^{-x^2/2}\, dx \right)^{M-1} dv \tag{8-1-54}$$

The bounds on the performance of linear block codes given above are in terms of the block error or code word error probability. The evaluation of the equivalent bit error probability P_b is much more complicated. In general, when a block error is made, some of the k information bits in the block will be correct and some will be in error. For orthogonal waveforms, the conversion factor that multiplies P_M to yield P_b is $2^{k-1}/(2^k - 1)$. This factor is unity for $k = 1$ and approaches $\frac{1}{2}$ as k increases, which is equivalent to assuming that, on the average, half of the k bits will be in error when a block error occurs. The conversion factor for coded waveforms depends in a complicated way on the distance properties of the code, but is certainly no worse than assuming that, on the average, half of the k bits will be in error when a block error occurs. Consequently, $P_b \leq \frac{1}{2} P_M$.

The bounds on performance given by (8-1-50), (8-1-52), and (8-1-54) also apply to the case in which a pair of bits of a code word are transmitted by quaternary PSK, since quaternary PSK may be viewed as being equivalent to two independent binary PSK waveforms transmitted in phase quadrature. Furthermore, the bounds in (8-1-52) and (8-1-54), which depend only on the minimum distance of the code, apply also to nonlinear binary block codes.

If binary orthogonal FSK is used to transmit each bit of a code word on the AWGN channel, the optimum receiver can be realized by means of two matched filters, one matched to the frequency corresponding to a transmission of a 0, and the other to the frequency corresponding to a transmission of a 1, followed by a decoder that forms the M correlation metrics corresponding to

the M possible code words. The detection at the receiver may be coherent or noncoherent. In either case, let r_{0j} and r_{1j} denote the input samples to the combiner. The correlation metrics formed by the decoder may be expressed as

$$CM_i = \sum_{j=1}^{n} [c_{ij} r_{1j} + (1 - c_{ij}) r_{0j}], \qquad i = 1, 2, \ldots, M \tag{8-1-55}$$

where c_{ij} represents the jth bit in the ith code word. The code word corresponding to the largest of the $\{CM_i\}$ is selected as the transmitted code word.

If the detection of the binary FSK waveforms is coherent, the random variables $\{r_{0j}\}$ and $\{r_{1j}\}$ are gaussian and, hence, the correlation metrics $\{CM_i\}$ are also gaussian. In this case, bounds on the performance of the code are easily obtained. To be specific, suppose that the all-zero code word $\mathbf{C}_1$ is transmitted. Then,

$$\left.\begin{aligned} r_{0j} &= \sqrt{\mathscr{E}_c} + n_{0j} \\ r_{1j} &= n_{1j} \end{aligned}\right\} \qquad j = 1, 2, \ldots, n \tag{8-1-56}$$

where the $\{n_{ij}\}$, $i = 0, 1$, $j = 1, 2, \ldots, n$, are mutually statistically independent gaussian random variables with zero mean and variance $\frac{1}{2}N_0$. Consequently CM_1 is gaussian with mean $\sqrt{\mathscr{E}_c}n$ and variance $\frac{1}{2}N_0$. On the other hand, the correlation metric CM_m, corresponding to the code word having weight w_m, is gaussian with mean $\sqrt{\mathscr{E}_c}n(1 - w_m/n)$ and variance $\frac{1}{2}nN_0$. Since the $\{CM_m\}$ are correlated, we again resort to a union bound. The correlation coefficients are given by

$$\rho_m = 1 - w_m/n \tag{8-1-57}$$

Hence, the probability that $CM_m > CM_1$ is

$$P_2(m) = Q(\sqrt{\gamma_b R_c w_m}) \tag{8-1-58}$$

Comparison of this result with that given in (8-1-49) for coherent PSK reveals that coherent PSK requires 3 dB less SNR to achieve the same performance. This is not surprising in view of the fact that uncoded PSK is 3 dB better than binary orthogonal FSK with coherent detection. Hence, the advantage of PSK over FSK is maintained in the coded waveforms. We conclude, then, that the bounds given in (8-1-50), (8-1-52), and (8-1-54) apply to coded waveforms transmitted by binary orthogonal coherent FSK with γ_b replaced by $\frac{1}{2}\gamma_b$.

If square-law detection of the binary orthogonal FSK signal is employed at the receiver, the performance is further degraded by the noncoherent combining loss, as shown in Chapter 12. Suppose again that the all-zero code word is transmitted. Then the correlation metrics are given by (8-1-55), where the input variables to the decoder are now

$$\left.\begin{aligned} r_{0j} &= |\sqrt{\mathscr{E}_c} + N_{0j}|^2 \\ r_{1j} &= |N_{1j}|^2 \end{aligned}\right\} \qquad j = 1, 2, \ldots, n \tag{8-1-59}$$

where $\{N_{0j}\}$ and $\{N_{1j}\}$ represent complex-valued mutually statistically independent gaussian random variables with zero mean and variance N_0. The correlation metric CM_1 is given as

$$CM_1 = \sum_{j=1}^{n} r_{0j} \tag{8-1-60}$$

while the correlation metric corresponding to the code word having weight w_m is statistically equivalent to the correlation metric of a code word in which $c_{mj} = 1$ for $1 \leq j \leq w_m$ and $c_{mj} = 0$ for $w_m + 1 \leq j \leq n$. Hence, CM_m may be expressed as

$$CM_m = \sum_{j=1}^{w_m} r_{1j} + \sum_{j=w_m+1}^{n} r_{0j} \tag{8-1-61}$$

The difference between CM_1 and CM_m is

$$CM_1 - CM_m = \sum_{j=1}^{w_m} (r_{0j} - r_{1j}) \tag{8-1-62}$$

and the probability of error is simply the probability that $CM_1 - CM_m < 0$. But this difference is a special case of the general quadratic form in complex-valued gaussian random variables considered in Chapter 12 and Appendix B. The expression for the probability of error in deciding between CM_1 and CM_m is (see Section 12-1-1)

$$P_2(m) = \frac{1}{2^{2w_m - 1}} \exp\left(-\tfrac{1}{2}\gamma_b R_c w_m\right) \sum_{i=0}^{w_m - 1} K_i \left(\tfrac{1}{2}\gamma_b R_c w_m\right)^i \tag{8-1-63}$$

where, by definition,

$$K_i = \frac{1}{i!} \sum_{r=0}^{w_m - 1 - i} \binom{2w_m - 1}{r} \tag{8-1-64}$$

The union bound obtained by summing $P_2(m)$ over $2 \leq m \leq M$ provides us with an upper bound on the probability of a code word error.

As an alternative, we may use the minimum distance instead of the weight distribution to obtain the looser upper bound

$$P_M \leq \frac{M-1}{2^{2d_{\min} - 1}} \exp\left(-\tfrac{1}{2}\gamma_b R_c d_{\min}\right) \sum_{i=0}^{d_{\min} - 1} K_i \left(\tfrac{1}{2}\gamma_b R_c d_{\min}\right)^i \tag{8-1-65}$$

A measure of the noncoherent combining loss inherent in the square-law detection and combining of the n elementary binary FSK waveforms in a code word can be obtained from Fig. 12-1-1, where $d_{\min}$ is used in place of L. The loss obtained is relative to the case in which the n elementary binary FSK waveforms are first detected coherently and combined as in (8-1-55) and then the sums are square-law-detected or envelope-detected to yield the M decision variables. The binary error probability for the latter case is

$$P_2(m) = \tfrac{1}{2} \exp\left(-\tfrac{1}{2}\gamma_b R_c w_m\right) \tag{8-1-66}$$

and, hence,

$$P_M \leq \sum_{m=2}^{M} P_2(m)$$

If $d_{\min}$ is used instead of the weight distribution, the union bound for the code word error probability in the latter case is

$$P_M \leq \tfrac{1}{2}(M-1)\exp\left(-\tfrac{1}{2}\gamma_b R_c d_{\min}\right) \tag{8-1-67}$$

The channel bandwidth required to transmit the coded waveforms can be determined as follows. If binary PSK is used to transmit each bit in a code word, the required bandwidth is approximately equal to the reciprocal of the time interval devoted to the transmission of each bit. For an information rate of R bits/s, the time available to transmit k information bits and $n-k$ redundant (parity) bits (n total bits) is $T = k/R$. Hence,

$$W = \frac{1}{T/n} = \frac{n}{k/R} = \frac{R}{R_c} \tag{8-1-68}$$

Therefore, the bandwidth expansion factor B_e for the coded waveform is

$$\begin{aligned} B_e &= \frac{W}{R} \\ &= \frac{n}{k} = \frac{1}{R_c} \end{aligned} \tag{8-1-69}$$

On the other hand, if binary FSK with noncoherent detection is employed for transmitting the bits in a code word, $W \approx 2n/T$, and, hence, the bandwidth expansion factor increases by approximately a factor of 2 relative to binary PSK. In any case, B_e increases inversely with the code rate, or, equivalently, it increases linearly with the block size n.

We are now in a position to compare the performance characteristics and bandwidth requirements of coded signaling waveforms with orthogonal signaling waveforms. A comparison of the expression for P_M given in (5-2-21) for orthogonal waveforms and in (8-1-54) for coded waveforms with coherent PSK indicates that the coded waveforms result in a loss of at most $10\log(n/2d_{\min})$ dB relative to orthogonal waveforms having the same number of waveforms. On the other hand, if we compensate for the loss in SNR due to coding by increasing the number of code words so that coded transmission requires $M_c = 2^k$ waveforms and orthogonal signaling requires $M_o = 2^{k_o}$ waveforms then [from the union bounds in (5-2-27) and (8-1-52)], the performance obtained with the two sets of signaling waveforms at high SNR is about equal if

$$k_o = 2R_c d_{\min} \tag{8-1-70}$$

Under this condition, the bandwidth expansion factor for orthogonal signaling can be expressed as

$$B_{eo} = \frac{M_o}{2\log_2 M_o} = \frac{2^{k_o}}{2k_o} = \frac{2^{2R_c d_{\min}}}{4R_c d_{\min}} \tag{8-1-71}$$

while, for coded signaling waveforms, we have $B_{ec} = 1/R_c$. The ratio of B_{eo} given in (8-1-71) to B_{ec}, which is

$$\frac{B_{eo}}{B_{ec}} = \frac{2^{2R_c d_{\min}}}{4d_{\min}} \tag{8-1-72}$$

provides a measure of the relative bandwidth between orthogonal signaling and signaling with coded coherent PSK waveforms.

For example, suppose we use a (63, 33) binary cyclic code that has a minimum distance $d_{\min} = 12$. The bandwidth ratio for orthogonal signaling relative to this code, given by (8-1-72), is 127. This is indicative of the bandwidth efficiency obtained through coding relative to orthogonal signaling.

8-1-5 Hard-Decision Decoding

The bounds given in Section 8-1-4 on the performance of coded signaling waveforms on the AWGN channel are based on the premise that the samples from the matched filter or cross correlator are not quantized. Although this processing yields the best performance, the basic limitation is the computational burden of forming M correlation metrics and comparing these to obtain the largest. The amount of computation becomes excessive when the number M of code words is large.

To reduce the computational burden, the analog samples can be quantized and the decoding operations are then performed digitally. In this subsection, we consider the extreme situation in which each sample corresponding to a single bit of a code word is quantized to two levels: zero and one. That is, a (hard) decision is made as to whether each transmitted bit in a code word is a 0 or a 1. The resulting discrete-time channel (consisting of the modulator, the AWGN channel, and the demodulator) constitutes a BSC with crossover probability p. If coherent PSK is employed in transmitting and receiving the bits in each code word then

$$\begin{aligned} p &= Q\left(\sqrt{\frac{2\mathscr{E}_c}{N_0}}\right) \\ &= Q(\sqrt{2\gamma_b R_c}) \end{aligned} \tag{8-1-73}$$

On the other hand, if FSK is used to transmit the bits in each code word then

$$p = Q(\sqrt{\gamma_b R_c}) \tag{8-1-74}$$

for coherent detection and

$$p = \tfrac{1}{2}\exp(-\tfrac{1}{2}\gamma_b R_c) \tag{8-1-75}$$

for noncoherent detection.

Minimum-Distance (Maximum-Likelihood) Decoding The n bits from the demodulator corresponding to a received code word are passed to the decoder, which compares the received code word with the M possible transmitted code words and decides in favor of the code word that is closest in Hamming distance (number of bit positions in which two code words differ) to the received code word. This minimum distance decoding rule is optimum in the sense that it results in a minimum probability of a code word error for the binary symmetric channel.

A conceptually simple, albeit computationally inefficient, method for decoding is to first add (modulo 2) the received code word vector to all the M possible transmitted code words $\mathbf{C}_i$ to obtain the error vectors $\mathbf{e}_i$. Hence, $\mathbf{e}_i$ represents the error event that must have occurred on the channel in order to transform the code word $\mathbf{C}_i$ into the particular received code word. The number of errors in transforming $\mathbf{C}_i$ into the received code word is just equal to the number of 1s in $\mathbf{e}_i$. Thus, if we simply compute the weight of each of the M error vectors $\{\mathbf{e}_i\}$ and decide in favor of the code word that results in the smallest weight error vector, we have, in effect, a realization of the minimum distance decoding rule.

A more efficient method for hard-decision decoding makes use of the parity check matrix $\mathbf{H}$. To elaborate, suppose that $\mathbf{C}_m$ is the transmitted code word and $\mathbf{Y}$ is the received code word at the output of the demodulator. In general, $\mathbf{Y}$ may be expressed as

$$\mathbf{Y} = \mathbf{C}_m + \mathbf{e}$$

where $\mathbf{e}$ denotes an arbitrary binary error vector. The product $\mathbf{YH}'$ yields

$$\begin{aligned} \mathbf{YH}' &= (\mathbf{C}_m + \mathbf{e})\mathbf{H}' \\ &= \mathbf{C}_m\mathbf{H}' + \mathbf{eH}' \\ &= \mathbf{eH}' = \mathbf{S} \end{aligned} \tag{8-1-76}$$

where the $(n-k)$-dimensional vector $\mathbf{S}$ is called the *syndrome of the error pattern*. In other words, the vector $\mathbf{S}$ has components that are zero for all parity check equations that are satisfied and nonzero for all parity check equations that are not satisfied. Thus, $\mathbf{S}$ contains the pattern of failures in the parity checks.

We emphasize that the syndrome $\mathbf{S}$ is a characteristic of the error pattern and not of the transmitted code word. Furthermore, we observe that there are 2^n possible error patterns and only 2^{n-k} syndromes. Consequently, different error patterns result in the same syndrome.

Suppose we construct a decoding table in which we list all the 2^k possible code words in the first row, beginning with the all-zero code word in the first (left-most) column. This all-zero code word also represents the all-zero error pattern. We fill in the first column by listing first all $n-1$ error patterns $\{\mathbf{e}_i\}$ of weight 1. If $n < 2^{n-k}$, we may then list all double error patterns, then all triple

error patterns, etc., until we have a total of 2^{n-k} entries in the first column. Thus, the number of rows that we can have is 2^{n-k}, which is equal to the number of syndromes. Next, we add each error pattern in the first column to the corresponding code words. Thus, we fill in the remainder of the $n \times (n-k)$ table as follows:

$$\begin{array}{llll} \mathbf{C}_1 & \mathbf{C}_2 & \mathbf{C}_3 & \cdots \quad \mathbf{C}_{2^k} \\ \mathbf{e}_2 & \mathbf{C}_2+\mathbf{e}_2 & \mathbf{C}_3+\mathbf{e}_2 & \cdots \quad \mathbf{C}_{2^k}+\mathbf{e}_2 \\ \mathbf{e}_3 & \mathbf{C}_2+\mathbf{e}_3 & \mathbf{C}_3+\mathbf{e}_3 & \cdots \quad \mathbf{C}_{2^k}+\mathbf{e}_3 \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{e}_{2^{n-k}} & \mathbf{C}_2+\mathbf{e}_{2^{n-k}} & \mathbf{C}_3+\mathbf{e}_{2^{n-k}} & \cdots \quad \mathbf{C}_{2^k}+\mathbf{e}_{2^{n-k}} \end{array}$$

This table is called a *standard array*. Each row, including the first, consists of k possible received code words that would result from the corresponding error pattern in the first column. Each row is called a *coset* and the first (left-most) code word (or error pattern) is called a *coset leader*. Therefore, a coset consists of all the possible received code words resulting from a particular error pattern (coset leader).

Example 8-1-10

Let us construct the standard array for the (5, 2), systematic code with generator matrix given by

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

Thic code has a minimum distance $d_{\min} = 3$. The standard array is given in Table 8-1-7. Note that in this code, the coset leaders consist of the all-zero error pattern, five error patterns of weight 1, and two error patterns of

TABLE 8-1-7 STANDARD ARRAY FOR THE (5, 2) CODE

Code words			
0 0 0 0 0	**0 1 0 1 1**	**1 0 1 0 1**	**1 1 1 1 0**
0 0 0 0 1	0 1 0 1 0	1 0 1 0 0	1 1 1 1 1
0 0 0 1 0	0 1 0 0 1	1 0 1 1 1	1 1 1 0 0
0 0 1 0 0	0 1 1 1 1	1 0 0 0 1	1 1 0 1 0
0 1 0 0 0	0 0 0 1 1	1 1 1 0 1	1 0 1 1 0
1 0 0 0 0	1 1 0 1 1	0 0 1 0 1	0 1 1 1 0
1 1 0 0 0	1 0 0 1 1	0 1 1 0 1	0 0 1 1 0
1 0 0 1 0	1 1 0 0 1	0 0 1 1 1	0 1 1 0 0

weight 2. Although many more double error patterns exist, there is only room for two to complete the table. These were selected such that their corresponding syndromes are distinct from those of the single error patterns.

Now, suppose that $\mathbf{e}_i$ is a coset leader and that $\mathbf{C}_m$ was the transmitted code word. Then, the error pattern $\mathbf{e}_i$ would result in the received code word

$$\mathbf{Y} = \mathbf{C}_m + \mathbf{e}_i$$

The syndrome is

$$\mathbf{S} = (\mathbf{C}_m + \mathbf{e}_i)\mathbf{H}' = \mathbf{C}_m\mathbf{H}' + \mathbf{e}_i\mathbf{H}' = \mathbf{e}_i\mathbf{H}'$$

Clearly, all received code words in the same coset have the same syndrome, since the latter depends only on the error pattern. Furthermore, each coset has a different syndrome. Having established this characteristic of the standard array, we may simply construct a syndrome decoding table in which we list the 2^{n-k} syndromes and the corresponding 2^{n-k} coset leaders that represent the minimum weight error patterns. Then, given a received code vector $\mathbf{Y}$, we compute the syndrome

$$\mathbf{S} = \mathbf{Y}\mathbf{H}'$$

For the computed $\mathbf{S}$, we find the corresponding (most likely) error vector, say $\hat{\mathbf{e}}_m$. This error vector is added to $\mathbf{Y}$ to yield the decoded word

$$\hat{\mathbf{C}}_m = \mathbf{Y} \oplus \hat{\mathbf{e}}_m$$

Example 8-1-11

Consider the (5, 2) code with the standard array given in Table 8-1-7. The syndromes versus the most likely error patterns are given in Table 8-1-8. Now suppose the actual error vector on the channel is

$$\mathbf{e} = [1 \quad 0 \quad 1 \quad 0 \quad 0]$$

TABLE 8-1-8 SYNDROME TABLE FOR THE (5, 2) CODE

Syndrome	Error pattern
0 0 0	0 0 0 0 0
0 0 1	0 0 0 0 1
0 1 0	0 0 0 1 0
1 0 0	0 0 1 0 0
0 1 1	0 1 0 0 0
1 0 1	1 0 0 0 0
1 1 0	1 1 0 0 0
1 1 1	1 0 0 1 0

The syndrome computed for the error is $\mathbf{S} = [0\ 0\ 1]$. Hence, the error determined from the table is $\hat{\mathbf{e}} = [0\ 0\ 0\ 0\ 1]$. When $\hat{\mathbf{e}}$ is added to $\mathbf{Y}$, the result is a decoding error. In other words the $(5, 2)$ code corrects all single errors and only two double errors, namely $[1\ 1\ 0\ 0\ 0]$ and $[1\ 0\ 0\ 1\ 0]$.

Syndrome Decoding of Cyclic Codes As described above, hard-decision decoding of a linear block code may be accomplished by first computing the syndrome $\mathbf{S} = \mathbf{YH}'$, then using a table lookup to find the most probable error pattern $\mathbf{e}$ corresponding to the computed syndrome $\mathbf{S}$, and, finally, adding the error pattern $\mathbf{e}$ to the received vector $\mathbf{Y}$ to obtain the most probable code word $\hat{\mathbf{C}}_m$. When the code is cyclic, the syndrome computation may be performed by a shift register similar in form to that used for encoding.

To elaborate, let us consider a systematic cyclic code and let us represent the received code vector $\mathbf{Y}$ by the polynomial $Y(p)$. In general, $\mathbf{Y} = \mathbf{C} + \mathbf{e}$, where $\mathbf{C}$ is the transmitted code word and $\mathbf{e}$ is the error vector. Hence, we have

$$\begin{aligned} Y(p) &= C(p) + e(p) \\ &= X(p)g(p) + e(p) \end{aligned} \tag{8-1-77}$$

Now, suppose we divide $Y(p)$ by the generator polynomial $g(p)$. This division will yield

$$\frac{Y(p)}{g(p)} = Q(p) + \frac{R(p)}{g(p)}$$

or, equivalently,

$$Y(p) = Q(p)g(p) + R(p) \tag{8-1-78}$$

The remainder $R(p)$ is a polynomial of degree less than or equal to $n - k - 1$. If we combine (8-1-77) with (8-1-78), we obtain

$$e(p) = [X(p) + Q(p)]g(p) + R(p) \tag{8-1-79}$$

This relationship illustrates that the remainder $R(p)$ obtained from dividing $Y(p)$ by $g(p)$ depends only on the error polynomial $e(p)$, and, hence, $R(p)$ is simply the syndrome associated with the error pattern $\mathbf{e}$. Therefore,

$$Y(p) = Q(p)g(p) + S(p) \tag{8-1-80}$$

where $S(p)$ is the syndrome polynomial of degree less than or equal to $n - k - 1$. If $g(p)$ divides $Y(p)$ exactly then $S(p) = 0$ and the received decoded word is $\hat{\mathbf{C}}_m = \mathbf{Y}$.

The division of $Y(p)$ by the generator polynomial $g(p)$ may be carried out by means of a shift register which performs division as described previously. First the received vector $\mathbf{Y}$ is shifted into an $(n - k)$-stage shift register as

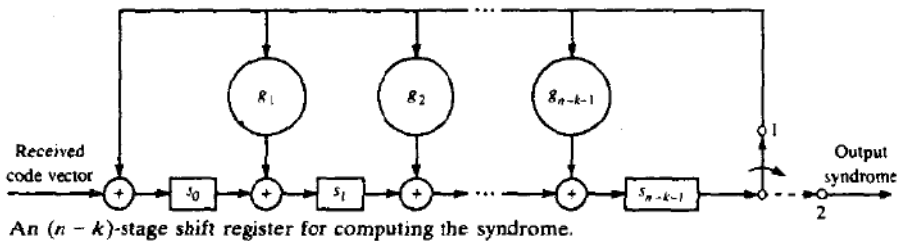


FIGURE 8-1-9 An $(n-k)$-stage shift register for computing the syndrome.

illustrated in Fig. 8-1-9. Initially, all the shift-register contents are zero and the switch is closed in position 1. After the entire n-bit received vector has been shifted into the register, the contents of the $n-k$ stages constitute the syndrome with the order of the bits numbered as shown in Fig. 8-1-9. These bits may be clocked out by throwing the switch into position 2. Given the syndrome from the $(n-k)$-stage shift register, a table lookup may be performed to identify the most probable error vector.

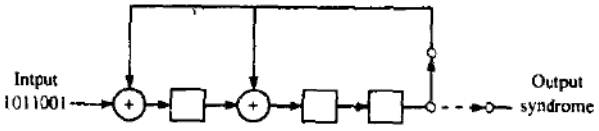
Example 8-1-12

Let us consider the syndrome computation for the (7, 4) cyclic Hamming code generated by the polynomial $g(p) = p^3 + p + 1$. Suppose that the received vector is $\mathbf{Y} = [1\ 0\ 0\ 1\ 1\ 0\ 1]$. This is fed into the three-stage register shown in Fig. 8-1-10. After seven shifts the contents of the shift register are 110, which corresponds to the syndrome $\mathbf{S} = [0\ 1\ 1]$. The most probable error vector corresponding to this syndrome is $\mathbf{e} = [0\ 0\ 0\ 1\ 0\ 0\ 0]$ and, hence,

$$\hat{\mathbf{C}}_m = \mathbf{Y} + \mathbf{e} = [1\ \ 0\ \ 0\ \ 0\ \ 1\ \ 0\ \ 1]$$

The information bits are 1 0 0 0.

FIGURE 8-1-10 Syndrome computation for the (7, 4) cyclic code with generator polynomial $g(p) = p^3 + p + 1$ and received vector $\mathbf{Y} = [1\ 0\ 0\ 1\ 1\ 0\ 1]$.



Shift	Register contents
0	000
1	100
2	010
3	001
4	010
5	101
5	100
7	110

The table lookup decoding method using the syndrome is practical only when $n - k$ is small, e.g., $n - k < 10$. This method is impractical for many interesting and powerful codes. For example, if $n - k = 20$, the table has 2^{20} (approximately 1 million) entries. Such a large amount of storage and the time required to locate an entry in such a large table renders the table lookup decoding method impractical for long codes having large numbers of check bits.

More efficient and practical hard-decision decoding algorithms have been devised for the class of cyclic codes and, more specifically, the BCH codes. A description of these algorithms requires further development of computational methods with finite fields, which is beyond the scope of our treatment of coding theory. It suffices to indicate that efficient decoding algorithms exist which make it possible to implement long BCH codes with high redundancy in practical digital communications systems. The interested reader is referred to the texts of Peterson and Weldon (1972). Lin and Costello (1983), Blahut (1983), and Berlekamp (1968), and to the paper by Forney (1965).

Error Detection and Error Correction Capability It is clear from the discussion above that when the syndrome consists of all zeros, the received code word is one of the 2^k possible transmitted code words. Since the minimum separation between a pair of code words is $d_{\min}$, it is possible for an error pattern of weight $d_{\min}$ to transform one of these 2^k code words in the code into another code word. When this happens we have an *undetected error*. On the other hand, if the actual number of errors is less than $d_{\min}$, the syndrome will have a nonzero weight. When this occurs, we have detected the presence of one or more errors on the channel. Clearly, the (n, k) block code is capable of *detecting* $d_{\min} - 1$ errors. Error detection may be used in conjunction with an automatic repeat-request (ARQ) scheme for retransmission of the code word.

The *error correction capability* of a code also depends on the minimum distance. However, the number of correctable error patterns is limited by the number of possible syndromes or coset leaders in the standard array. To determine the error correction capability of an (n, k) code, it is convenient to view the 2^k code words as points in an n-dimensional space. If each code word is viewed as the center of a sphere of radius (Hamming distance) t, the largest value that t may have without intersection (or tangency) of any pair of the 2^k spheres is $t = \lfloor \frac{1}{2}(d_{\min} - 1) \rfloor$, where $\lfloor x \rfloor$ denotes the largest integer contained in x. Within each sphere lie all the possible received code words of distance less than or equal to t from the valid code word. Consequently, any received code vector that falls within a sphere is decoded into the valid code word at the center of the sphere. This implies that an (n, k) code with minimum distance $d_{\min}$ is capable of correcting $t = \lfloor \frac{1}{2}(d_{\min} - 1) \rfloor$ errors. Figure 8-1-11 is a two-dimensional representation of the code words and the spheres.

As described above, a code may be used to detect $d_{\min} - 1$ errors or to correct $t = \lfloor \frac{1}{2}(d_{\min} - 1) \rfloor$ errors. Clearly, to correct t error implies that we have

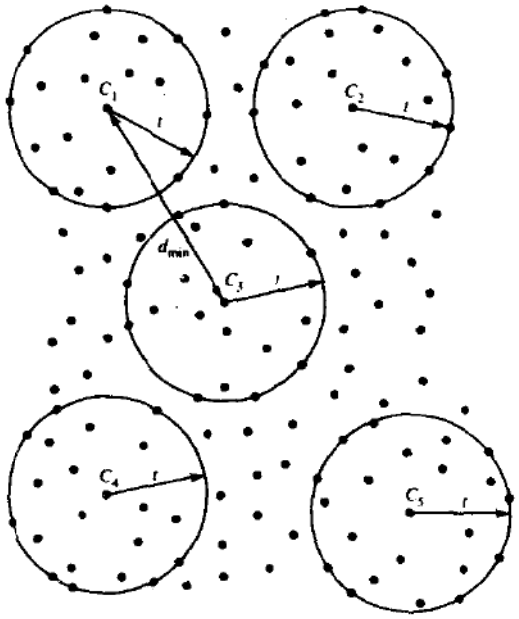


FIGURE 8-1-11 A representation of code words as centers of spheres of radius $t = \lfloor \frac{1}{2}(d_{min} - 1) \rfloor$.

detected t errors. However, it is also possible to detect more than t errors if we compromise in the error correction capability of the code. For example, a code with $d_{min} = 7$ can correct $t = 3$ errors. If we wish to detect four errors, we can do so by reducing the radius of the sphere around each code word from 3 to 2. Thus, patterns with four errors are detectable but only patterns of two errors are correctable. In other words, when only two errors occur, these are corrected, and when three or four errors occur, the receiver may ask for a retransmission. If more than four errors occur, they will go undetected if the code word falls within a sphere of radius 2. Similarly, for $d_{min} = 7$, five errors can be detected and one error corrected. In general, a code with minimum distance d_{min} can detect e_d errors and correct e_c errors, where

$$e_d + e_c \leq d_{min} - 1$$

and

$$e_c \leq e_d$$

Probability of Error Based on Error Correction We conclude this section with the derivation of the probability of error for hard-decision decoding of linear binary block codes based on error correction only.

From the above discussion, it is clear that the optimum decoder for a binary

symmetric channel will decode correctly if (but not necessarily only if) the number of errors in a code word is less than half the minimum distance $d_{\min}$ of the code. That is, any number of errors up to

$$t = \lfloor \tfrac{1}{2}(d_{\min} - 1) \rfloor$$

are always correctable. Since the binary symmetric channel is memoryless, the bit errors occur independently. Hence, the probability of m errors in a block of n bits is

$$P(m, n) = \binom{n}{m} p^m (1-p)^{n-m} \tag{8-1-81}$$

and, therefore, the probability of a code word error is upper-bounded by the expression

$$P_M \leq \sum_{m=t+1}^{n} P(m, n) \tag{8-1-82}$$

Equality holds in (8-1-82) if the linear block code is a perfect code. In order to describe the basic characteristics of a perfect code, suppose we place a sphere of radius t around each of the possible transmitted code words. Each sphere around a code word contains the set of all code words of Hamming distance less than or equal to t from the code word. Now, the number of code words in a sphere of radius $t = \lfloor \frac{1}{2}(d_{\min} - 1) \rfloor$ is

$$1 + \binom{n}{1} + \binom{n}{2} + \ldots + \binom{n}{t} = \sum_{i=0}^{t} \binom{n}{i}$$

Since there are $M = 2^k$ possible transmitted code words, there are 2^k nonoverlapping spheres each having a radius t. The total number of code words enclosed in the 2^k spheres cannot exceed the 2^n possible received code words. Thus, a t-error correcting code must satisfy the inequality

$$2^k \sum_{i=0}^{t} \binom{n}{i} \leq 2^n$$

or, equivalently,

$$2^{n-k} \geq \sum_{i=0}^{t} \binom{n}{i} \tag{8-1-83}$$

A perfect code has the property that all spheres of Hamming distance $t = \lfloor \frac{1}{2}(d_{\min} - 1) \rfloor$ around the $M = 2^k$ possible transmitted code words are disjoint and every received code word falls in one of the spheres. Thus, every received code word is at most, at distance t from one of the possible transmitted code words and (8-1-83) holds with equality. For such a code, all error

patterns of weight less than or equal to t are corrected by the optimum (minimum distance) decoder. On the other hand, any error pattern of weight $t+1$ or greater cannot be corrected. Consequently, the expression for the error probability given in (8-1-82) holds with equality.The Golay (23, 12) code, having $d_{\min} = 7$ and $t = 3$, is a perfect code. The Hamming codes, which have the parameters $n = 2^{n-k} - 1$, $d_{min} = 3$, and $t = 1$, are also perfect codes. These two nontrivial codes and the trivial code consisting of two code words of odd length n and $d_{\min} = n$ are the only perfect binary block codes. These codes are optimum on the BSC in the sense that they result in a minimum error probability among all codes having the same block length and the same number of information bits.

The optimality property defined above also holds for quasiperfect codes. A quasiperfect code is characterized by the property that all spheres of Hamming radius t around the M possible transmitted code words are disjoint and every received code word is at most at distance $t+1$ from one of the possible transmitted code words. For such a code, all error patterns of weight less than or equal to t and some error patterns of weight $t+1$ are correctable, but any error pattern of weight $t+2$ or greater leads to incorrect decoding of the code word. Clearly, (8-1-82) is an upper bound on the error probability and

$$P_M \geq \sum_{m=t+2}^{n} P(m, n) \tag{8-1-84}$$

is a lower bound.

A more precise measure of the performance for quasiperfect codes can be obtained by making use of the inequality in (8-1-83). That is, the total number of code words outside the 2^k spheres of radius t is

$$N_{t+1} = 2^n - 2^k \sum_{i=0}^{t} \binom{n}{i}$$

If these code words are equally subdivided into 2^k sets and each set is associated with one of the 2^k spheres then each sphere is enlarged by the addition of

$$\beta_{t+1} = 2^{n-k} - \sum_{i=0}^{t} \binom{n}{i} \tag{8-1-85}$$

code words having distance $t+1$ from the transmitted code word. Consequently, of the $\binom{n}{t+1}$ error patterns of distance $t+1$ from each code word, we can correct β_{t+1} error patterns. Thus, the error probability for decoding the quasiperfect code may be expressed as

$$P_M = \sum_{m=t+2}^{n} P(m, n) + \left[\binom{n}{t+1} - \beta_{t+1}\right] p^{t+1}(1-p)^{n-t-1} \tag{8-1-86}$$

There are many known quasiperfect codes, although they do not exist for

all choices of n and k. Since such codes are optimum for the binary symmetric channel, any (n, k) linear block code must have an error probability that is at least as large as (8-1-86). Consequently, (8-1-86) is a lower bound on the probability of error for any (n, k) linear block code, where t is the largest integer such that $\beta_{t-1} \geq 0$.

Another pair of upper and lower bounds is obtained by considering two code words that differ by the minimum distance. First, we note that P_M cannot be less than the probability of erroneously decoding the transmitted code word as its nearest neighbor, which is at distance $d_{\min}$ from the transmitted code word. That is,

$$P_M \geq \sum_{m=[d_{\min}/2]+1}^{d_{\min}} \binom{d_{\min}}{m} p^m (1-p)^{d_{\min}-m} \tag{8-1-87}$$

On the other hand, P_M cannot be greater than $M-1$ times the probability of erroneously decoding the transmitted code word as its nearest neighbor, which is at distance $d_{\min}$ from the transmitted code word. That is a union bound, which is expressed as

$$P_M \leq (M-1) \sum_{m=[d_{\min}/2]+1}^{d_{\min}} \binom{d_{\min}}{m} p^m (1-p)^{d_{\min}-m} \tag{8-1-88}$$

When M is large, the lower bound in (8-1-87) and the upper bound in (8-1-88) are very loose.

A tight upper bound on P_M can be obtained by applying the Chernoff bound presented earlier in Section 2-1-6. We assume again that the all-zero code was transmitted. In comparing the received code word to the all-zero code word and to a code word of weight w_m, the probability of a decoding error, obtained from the Chernoff bound (Problem 8-22), is upper-bounded by the expression

$$P_2(w_m) \leq [4p(1-p)]^{w_m/2} \tag{8-1-89}$$

The union of these binary decisions yields the upper bound

$$P_M \leq \sum_{m=2}^{M} [4p(1-p)]^{w_m/2} \tag{8-1-90}$$

A simpler version of (8-1-90) is obtained if we employ $d_{\min}$ in place of the weight distribution. That is,

$$P_M \leq (M-1)[4p(1-p)]^{d_{\min}/2} \tag{8-1-91}$$

Of course (8-1-90) is a tighter upper bound than (8-1-91).

In Section 8-1-6, we compare the various bounds given above for a specific code, namely, the Golay (23, 12) code. In addition, we compare the error rate performance of hard-decision and soft-decision decoding.

8-1-6 Comparison of Performance between Hard-Decision and Soft-Decision Decoding

It is both interesting and instructive to compare the bounds on the error rate performance of linear block codes for soft-decision decoding and hard-decision decoding on an AWGN channel. For illustrative purposes, we shall use the Golay (23, 12) code, which has the relatively simple weight distribution given in Table 8-1-1. As stated previously, this code has a minimum distance $d_{\min} = 7$.

First we compute and compare the bounds on the error probability for hard-decision decoding. Since the Golay (23, 12) code is a perfect code, the exact error probability for hard-decision decoding is

$$P_M = \sum_{m=4}^{23} \binom{23}{m} p^m (1-p)^{23-m}$$

$$= 1 - \sum_{m=0}^{3} \binom{23}{m} p^m (1-p)^{23-m} \tag{8-1-92}$$

where p is the probability of a binary digit error for the binary symmetric channel. Binary (or four-phase) coherent PSK is assumed to be the modulation/demodulation technique for the transmission and reception of the binary digits contained in each code word. Thus, the appropriate expression for p is given by (8-1-73). In addition to the exact error probability given by (8-1-92), we have the lower bound given by (8-1-87) and the three upper bounds given by (8-1-88), (8-1-90), and (8-1-91).

Numerical results obtained from these bounds are compared with the exact error probability in Fig. 8-1-12. We observe that the lower bound is very loose.

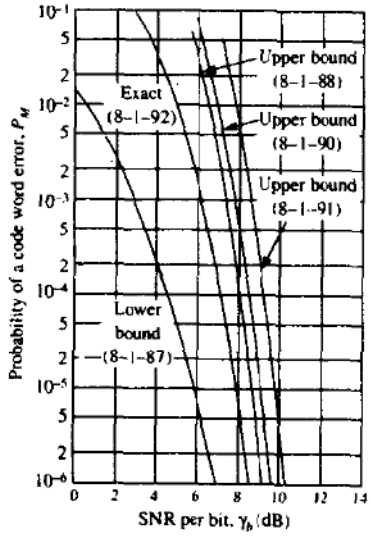


FIGURE 8-1-12 Comparison of bounds with exact error probability for hard-decision decoding of Golay (23, 12) code.

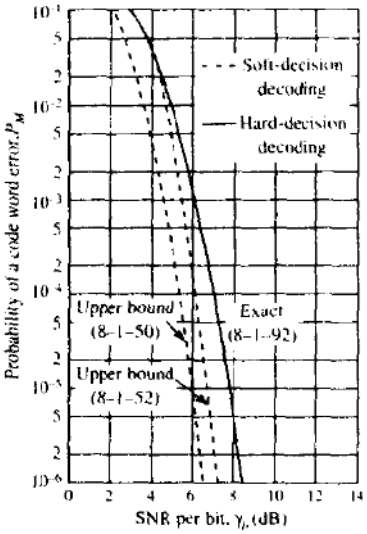


FIGURE 8-1-13 Comparison of soft-decision decoding with hard-decision decoding for the Golay (23, 12) code.

At $P_M = 10^{-5}$, the lower bound is off by approximately 2 dB from the exact error probability. At $P_M = 10^{-2}$, the difference increases to approximately 4 dB. Of the three upper bounds, the one given by (8-1-88) is the tightest; it differs by less than 1 dB from the exact error probability at $P_M = 10^{-5}$. The Chernoff bound in (8-1-90), which employs the weight distribution, is also relatively tight. Finally, the Chernoff bound that employs only the minimum distance of the code is the poorest of the three. At $P_M = 10^{-5}$, it differs from the exact error probability by approximately 2 dB. All three upper bounds are very loose for error rates above $P_M = 10^{-2}$.

It is also interesting to compare the performance between soft- and hard-decision decoding. For this comparison, we use the upper bounds on the error probability for soft-decision decoding given by (8-1-52) and the exact error probability for hard-decision decoding given by (8-1-92). Figure 8-1-13 illustrates these performance characteristics. We observe that the two bounds for soft-decision decoding differ by approximately 0.5 dB at $P_M = 10^{-6}$ and by approximately 1 dB at $P_M = 10^{-2}$. We also observe that the difference in performance between hard- and soft-decision decoding is approximately 2 dB in the range $10^{-2} < P_M < 10^{-6}$. In the range $P_M > 10^{-2}$, the curve of the error probability for hard-decision decoding crosses the curves for the bounds. This behavior indicates that the bounds for soft-decision decoding are loose when $P_M > 10^{-2}$.

The 2 dB difference between hard- and soft-decision decoding is a characteristic that applies not only to the Golay code, but is a fundamental result that applies in general to coded digital communications over the AWGN channel. This result is derived below by computing the capacity of the AWGN channel with hard- and soft-decision decoding.

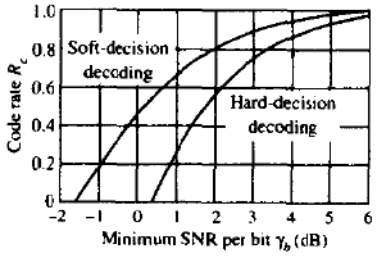


FIGURE 8-1-14 Code rate as a function of the minimum SNR per bit for soft- and hard-decision decoding.

The channel capacity of the BSC in bits per code symbol, derived in Section 7-1-2, is

$$C = 1 + p \log_2 p + (1 - p) \log_2 (1 - p) \tag{8-1-93}$$

where the probability of a bit error for binary, coherent PSK on an AWGN channel is given by (8-1-73). Suppose we use (8-1-73) for p, let $C = R_c$ in (8-1-93), and then determine the value of γ_b that satisfies this equation. The result is shown in Fig. 8-1-14 as a graph of R_c versus γ_b. For example, suppose that we are interested in using a code with rate $R_c = \frac{1}{2}$. For this code rate, note that the minimum SNR per bit required to achieve capacity with hard-decision decoding is approximately 1.6 dB.

What is the limit on the minimum SNR as the code rate approaches zero? For small values of R_c, the probability p can be approximated as

$$p \approx \tfrac{1}{2} - \sqrt{\gamma_b R_c / \pi} \tag{8-1-94}$$

When the expression for p is substituted into (8-1-93) and the logarithms in (8-1-93) are approximated by

$$\log_2 (1 + x) \approx (x - \tfrac{1}{2}x^2)/\ln 2$$

the channel capacity formula reduces to

$$C = \frac{2}{\pi \ln 2} \gamma_b R_c \tag{8-1-95}$$

Now we set $C = R_c$. Thus, in the limit as R_c approaches zero, we obtain the result

$$\gamma_b = \tfrac{1}{2}\pi \ln 2 \quad (0.37 \text{ dB}) \tag{8-1-96}$$

The capacity of the binary-input AWGN channel with soft-decision decoding can be computed in a similar manner. The expression for the capacity in bits per code symbol, derived in Section 7-1-2, is

$$C = \tfrac{1}{2} \sum_{k=0}^{1} \int_{-\infty}^{\infty} p(y \mid k) \log_2 \frac{p(y \mid k)}{p(y)} dy \tag{8-1-97}$$

where $p(y \mid k)$, $k = 0, 1$, denote the probability density functions of the demodulator output conditioned on the transmitted bit being a 0 and a 1, respectively. For the AWGN channel, we have

$$p(y \mid k) = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-(y-m_k)^2/2\sigma^2}, \qquad k = 0, 1 \tag{8-1-98}$$

where $m_0 = -\sqrt{\mathscr{E}_c}$, $m_1 = \sqrt{\mathscr{E}_c}$, $\sigma^2 = \frac{1}{2}N_0$, and $\mathscr{E}_c = R_c \mathscr{E}_b$. The unconditional probability density $p(y)$ is simply one-half of the sum of $p(y \mid 1)$ and $p(y \mid 0)$. As R_c approaches zero, the expression (8-1-97) for the channel capacity can be approximated as

$$C \approx \gamma_b R_c / \ln 2 \tag{8-1-99}$$

Again, we set $C = R_c$. Thus, as $R_c \to 0$, the minimum SNR per bit to achieve capacity is

$$\gamma_b = \ln 2 \qquad (-1.6\,\text{dB}) \tag{8-1-100}$$

By using (8-1-98) in (8-1-97) and setting $C = R_c$, a numerical solution can be obtained for code rates in the range $0 \leq R_c \leq 1$. The result of this solution is also shown in Fig. 8-1-14.

From the above, we observe that in the limit as R_c approaches zero, the difference in SNR γ_b between hard- and soft-decision decoding is $\frac{1}{2}\pi$, which is approximately 2 dB. On the other hand, as R_c increases toward unity, the difference in γ_b between these two decoding techniques decreases. For example, at $R_c = 0.8$, the difference is about 1.5 dB.

The curves in Fig. 8-1-14 provide more information than just the difference in performance between soft- and hard-decision decoding. These curves also specify the minimum SNR per bit that is required for a given code rate. For example, a code rate of $R_c = 0.8$ can provide arbitrarily small error probability at an SNR per bit of 2 dB, when soft-decision decoding is used. By comparison, an uncoded binary PSK requires 9.6 dB to achieve an error probability of 10^{-5}. Hence, a 7.6 dB gain is possible by employing a rate $R_c = \frac{4}{5}$ code. Unfortunately, to achieve such a large coding gain usually implies the use of an extremely long block length code, which leads to a very complex receiver. Nevertheless, the curves in Fig. 8-1-14 provide a benchmark for comparing the coding gains achieved by practically implementable codes with the ultimate limits for either soft- or hard-decision decoding.

Instead of comparing the difference between hard- and soft-decision decoding based on the channel capacity relations, we may perform similar comparisons based on the random coding rate parameters. In Chapter 7, we demonstrated that the ensemble average probability of error for randomly selected binary code words is upper-bounded as

$$\bar{P}_e < 2^{-n(R_0 - R_c)} \tag{8-1-101}$$

where $R_c = k/n$ is the code rate and the cutoff rate R_0 represents the upper

bound on R_c such that $\bar{P}_e \to 0$ as $n \to \infty$. For unquantized (soft-decision) decoding, R_0 is given as

$$R_0 = \log_2 \frac{2}{1 + e^{-\mathscr{E}_c/N_0}} \tag{8-1-102}$$

where $\mathscr{E}_c/N_0 = R_c \gamma_b$ is the SNR per dimension. This result was derived in Section 7-2.

On the other hand, if the output of the demodulator is quantized to Q levels prior to decoding, the Chernoff bound may be used to upper-bound the ensemble average binary error probability $\overline{P_2(\mathbf{s}_i, \mathbf{s}_m)}$ defined in Section 7-2. The result of this derivation is the same upper bound on $\bar{P}_e$ given in (8-1-101) but with R_0 replaced by R_Q, where

$$R_Q = \max_{\{p_j\}} \left\{ -\log_2 \sum_{i=0}^{Q-1} \left[\sum_{j=0}^{1} p_j \sqrt{P(i \mid j)} \right]^2 \right\} \tag{8-1-103}$$

In (8-1-103), $\{p_j\}$ are the prior probabilities of the two signals at the input to the channel and $\{P(i \mid j)\}$ denote the transition probabilities of the channel. For example, in the case of a binary symmetric channel, we have $p_1 = p_0 = \frac{1}{2}$, $P(0 \mid 0) = P(1 \mid 1) = 1 - p$, and $P(0 \mid 1) = P(1 \mid 0) = p$. Hence,

$$R_Q = \log_2 \frac{2}{1 + \sqrt{4p(1-p)}} \qquad Q = 2 \tag{8-1-104}$$

where

$$p = Q(\sqrt{2\gamma_b R_c}) \tag{8-1-105}$$

A plot of R_Q versus $10 \log (\mathscr{E}_c/N_0)$ is illustrated in Fig. 8-1-15 for $Q = 2$ and $Q = \infty$ (soft-decision decoding). Note that the difference in decoder performance between unquantized soft-decision decoding and hard-decision decoding is approximately 2 dB. In fact, it is easily demonstrated again that as $\mathscr{E}_c/N_0 \to 0$, the loss in performance due to hard-decision decoding is

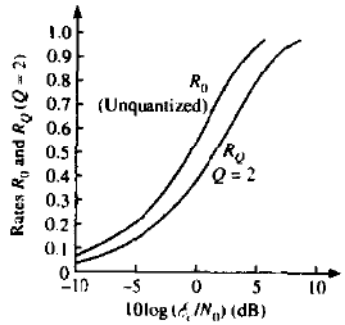


FIGURE 8-1-15 Comparison of R_0 (soft-decision decoding) with R_Q (hard-decision decoding) as a function of the SNR per dimension.

$10 \log_{10} \frac{1}{2}\pi \approx 2$ dB, which is the same decibel difference that was obtained in our comparison of the channel capacity relations. We mention that about 1 dB of this loss can be recovered by quantizing the output of the demodulator to three levels instead of two (see Problem 7-11). Additional improvements are possible by quantizing the output into more than three levels, as shown in Section 7-3.

8-1-7 Bounds on Minimum Distance of Linear Block Codes

The expressions for the probability of error derived in this chapter for soft-decision and hard-decision decoding of linear binary block codes clearly indicate the importance that the minimum distance parameter plays in the performance of the code. If we consider soft-decision decoding, for example, the upper bound on the error probability given by (8-1-52) indicates that, for a given code rate $R_c = k/n$, the probability of error in an AWGN channel decreases exponentially with $d_{\min}$. When this bound is used in conjunction with the lower bound on $d_{\min}$ given below, we obtain an upper bound on P_M that can be achieved by many known codes. Similarly, we may use the upper bound given by (8-1-82) for the probability of error for hard-decision decoding in conjunction with the lower bound on $d_{\min}$ to obtain an upper bound on the error probability for linear binary block codes on the binary symmetric channel.

On the other hand, an upper bound on $d_{\min}$ can be used to determine a lower bound on the probability of error achieved by the best code. For example, suppose that hard-decision decoding is employed. In this case, we have the two lower bounds on P_M given by (8-1-86) and (8-1-87), with the former being the tighter. When either one of these two bounds is used in conjunction with an upper bound on $d_{\min}$ the result is a lower bound on P_M for the best (n, k) code. Thus, upper and lower bounds on $d_{\min}$ are important in assessing the capabilities of codes.

A simple upper bound on the minimum distance of an (n, k) binary or non-binary linear block code was given in (8-1-14) as $d_{\min} \leq n - k + 1$. It is convenient to normalize this expression by the block size n. That is,

$$\frac{d_{\min}}{n} \leq (1 - R_c) + \frac{1}{n} \tag{8-1-106}$$

where R_c is the code rate. For large n, the factor $1/n$ can be neglected.

If a code has the largest possible distance, i.e., $d_{\min} = n - k + 1$, it is called a *maximum-distance-separable code*. Except for the trivial repetition-type codes, there are no binary maximum-separable codes. In fact, the upper bound in (8-1-106) is extremely loose for binary codes. On the other hand, nonbinary codes with $d_{\min} = n - k + 1$ do exist. For example, the Reed–Solomon codes, which comprise a subclass of BCH codes, are maximum-distance-separable.

In addition to the upper bound given above, there are several relatively

tight bounds on the minimum distance of linear block codes. We shall briefly describe four important bounds, three of which are upper bounds and the other a lower bound. The derivations of these bounds are lengthy and are not of particular interest in our subsequent discussion. The interested reader may refer to Chapter 4 of the book by Peterson and Weldon (1972) for those derivations.

One upper bound on the minimum distance can be obtained from the inequality in (8-1-83). By taking the logarithm of both sides of (8-1-83) and dividing by n, we obtain

$$1 - R_c \geq \frac{1}{n} \log_2 \sum_{i=0}^{t} \binom{n}{i} \tag{8-1-107}$$

Since the error-correcting capability of the code, measured by t, is related to the minimum distance, the above relation is an upper bound on the minimum distance. It is called the *Hamming upper bound.*

The asymptotic form of (8-1-107) is obtained by letting $n \to \infty$. Now, for any n, let t_0 be the largest integer t for which (8-1-107) holds. Then, it can be shown (Peterson and Weldon, 1972) that as $n \to \infty$, the ratio t/n for any (n, k) block code cannot exceed t_0/n, where t_0/n satisfies the equation

$$1 - R_c = H(t_0/n) \tag{8-1-108}$$

and $H(x)$ is the binary entropy function defined by (3-2-10).

The generalization of the Hamming bound to nonbinary codes is simply

$$1 - R_c \geq \frac{1}{n} \log_q \left[\sum_{i=0}^{t} \binom{n}{i} (q-1)^i \right] \tag{8-1-109}$$

Another upper bound, developed by Plotkin (1960), may be stated as follows. The number of check digits required to achieve a minimum distance $d_{\min}$ in an (n, k) linear block code satisfies the inequality

$$n - k \geq \frac{q d_{\min} - 1}{q - 1} - 1 - \log_q d_{\min} \tag{8-1-110}$$

where q is the alphabet size. When the code is binary, (8-1-110) may be expressed as

$$\frac{d_{\min}}{n} \left(1 - \frac{1}{2d_{\min}} \log_2 d_{\min} \right) \leq \frac{1}{2} \left(1 - R_c + \frac{2}{n} \right)$$

In the limit as $n \to \infty$ with $d_{\min}/n \leq \frac{1}{2}$, (8-1-110) reduces to

$$d_{\min}/n \leq \tfrac{1}{2}(1 - R_c) \tag{8-1-111}$$

Finally, there is another tight upper bound on the minimum distance obtained by Elias (Berlekamp, 1968). It may be expressed in its asymptotic form as

$$d_{\min}/n \leq 2A(1-A) \tag{8-1-112}$$

where the parameter A is related to the code rate through the equation

$$R_c = 1 + A\log_2 A + (1-A)\log_2(1-A), \quad 0 \leq A \leq \tfrac{1}{2} \tag{8-1-113}$$

Lower bounds on the minimum distance of (n, k) block codes also exist. In particular, binary block codes exist that have a normalized minimum distance that asymptotically satisfies the inequality

$$d_{\min}/n \geq \alpha \tag{8-1-114}$$

where α is related to the code rate through the equation

$$\begin{aligned} R_c &= 1 - H(\alpha) \\ &= 1 + \alpha\log_2\alpha + (1-\alpha)\log_2(1-\alpha), \quad 0 \leq \alpha \leq \tfrac{1}{2} \end{aligned} \tag{8-1-115}$$

This lower bound is a special case of a lower bound developed by Gilbert (1952) and Varsharmov (1957), which applies to nonbinary and binary block codes.

The asymptotic bounds given above are plotted in Fig. 8-1-16 for binary codes. Also plotted in the figure for purposes of comparison are curves of the minimum distance as a function of code rate for BCH codes of block lengths $n = 31$ and 63. We observe that for $n = 31$ and 63, the normalized minimum distance falls well above the Varsharmov–Gilbert lower bound. As the block length n increases, the efficiency of the BCH codes diminishes. For example, when $n = 1023$, the curve for the normalized minimum distance falls close to

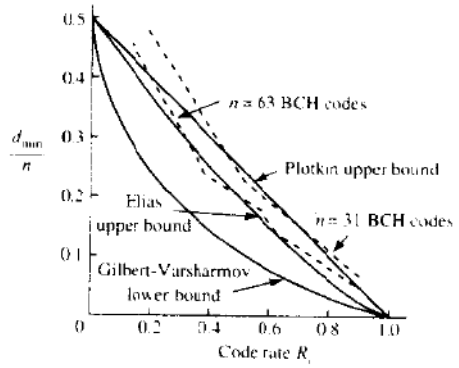


FIGURE 8-1-16 Upper and lower bounds on normalized minimum distance as a function of code rate.

the Varsharmov–Gilbert bound. As n increases beyond $n = 1023$, the normalized minimum distance of the BCH codes continues to decrease and falls below the Varsharmov–Gilbert bound. That is, $d_{\min}/n$ approaches zero as n tends to infinity. Consequently the BCH codes, which are the most important class of cyclic codes, are not very efficient at large block lengths.

8-1-8 Nonbinary Block Codes and Concatenated Block Codes

A nonbinary block code consists of a set of fixed-length code words in which the elements of the code words are selected from an alphabet of q symbols, denoted by $\{0, 1, 2, \ldots, q-1\}$. Usually, $q = 2^k$, so that k information bits are mapped into one of the q symbols. The length of the nonbinary code word is denoted by N and the number of information symbols encoded into a block of N symbols is denoted by K. The minimum distance of the nonbinary code is denoted by $D_{\min}$. A systematic (N, K) block code consists of K information symbols and $N - K$ parity check symbols.

Among the various types of nonbinary linear block codes, the Reed–Solomon codes are some of the most important for practical applications. As indicated previously, they comprise a subset of the BCH codes, which in turn are a class of cyclic codes. These codes are described by the parameters

$$\begin{aligned} N &= q - 1 = 2^k - 1 \\ K &= 1, 2, 3, \ldots, N-1 \\ D_{\min} &= N - K + 1 \\ R_c &= K/N \end{aligned} \tag{8-1-116}$$

Such a code is guaranteed to correct up to

$$\begin{aligned} t &= \lfloor \tfrac{1}{2}(D_{min} - 1) \rfloor \\ &= \lfloor \tfrac{1}{2}(N - K) \rfloor \end{aligned} \tag{8-1-117}$$

symbol errors. Of course, these codes may be extended or shortened in the manner described previously for binary block codes.

The weight distribution $\{A_i\}$ of the class of Reed–Solomon codes is known. The coefficients in the weight enumerating polynomial are given as

$$A_i = \binom{N}{i}(q-1)\sum_{j=0}^{i-D}(-1)^j\binom{i-1}{j}q^{i-j-D}, \quad i \geq D_{\min} \tag{8-1-118}$$

where $D \equiv D_{min}$ and $q = 2^k$.

One reason for the importance of the Reed–Solomon codes is their good

distance properties. A second reason for their importance is the existence of efficient hard-decision decoding algorithms, which make it possible to implement relatively long codes in many practical applications where coding is desirable.

A nonbinary code is particularly matched to an M-ary modulation technique for transmitting the 2^k possible symbols. Specifically, M-ary orthogonal signaling, e.g., M-ary FSK, is frequently used. Each of the 2^k symbols in the q-ary alphabet is mapped to one of the $M = 2^k$ orthogonal signals. Thus, the transmission of a code word is accomplished by transmitting N orthogonal signals, where each signal is selected from the set of $M = 2^k$ possible signals.

The optimum demodulator for such a signal corrupted by AWGN consists of M matched filters (or cross-correlators) whose outputs are passed to the decoder, either in the form of soft decisions or in the form of hard decisions. If hard decisions are made by the demodulator, the symbol error probability P_M and the code parameters are sufficient to characterize the performance of the decoder. In fact, the modulator, the AWGN channel, and the demodulator form an equivalent discrete (M-ary) input, discrete (M-ary) output, symmetric memoryless channel characterized by the transition probabilities $P_c = 1 - P_M$ and $P_M/(M-1)$. This channel model, which is illustrated in Fig. 8-1-17, is a generalization of the BSC.

The performance of the hard-decision decoder may be characterized by the following upper bound on the code word error probability:

$$P_e \leq \sum_{i=t+1}^{N} \binom{N}{i} P_M^i (1 - P_M)^{N-i} \tag{8-1-119}$$

where t is the number of errors guaranteed to be corrected by the code.

When a code word error is made, the corresponding symbol error probability is

$$P_{es} = \frac{1}{N} \sum_{i=t+1}^{N} i \binom{N}{i} P_M^i (1 - P_M)^{N-i} \tag{8-1-120}$$

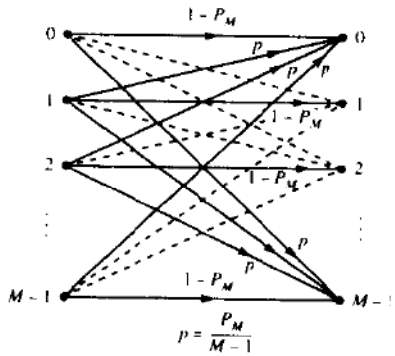


FIGURE 8-1-17 M-ary input, M-ary output, symmetric memoryless channel.

Furthermore, if the symbols are converted to binary digits, the bit error probability corresponding to (8-1-120) is

$$P_{eb} = \frac{2^{k-1}}{2^k - 1} P_{es} \tag{8-1-121}$$

Example 8-1-13

Let us evaluate the performance of an $N = 2^5 - 1 = 31$ Reed–Solomon code with $D_{\min} = 3$, 5, 9, and 17. The corresponding values of K are 29, 27, 23, and 15. The modulation is $M = 32$ orthogonal FSK with noncoherent detection at the receiver.

The probability of a symbol error is given by (5-4-46), and may be expressed as

$$P_M = \frac{1}{M} e^{-\gamma} \sum_{n=2}^{M} (-1)^n \binom{M}{n} e^{\gamma/n} \tag{8-1-122}$$

where γ is the SNR per code symbol. By using (8-1-122) in (8-1-120) and combining the result with (8-1-121), we obtain the bit error probability. The results of these computations are plotted in Fig. 8-1-18. Note that the more powerful codes (large $D_{\min}$) give poorer performance at low SNR per bit than the weaker codes. On the other hand, at high SNR, the more powerful codes give better performance. Hence, there are crossovers among the various codes, as illustrated for example in Fig. 8-1-18 for the $t = 1$ and $t = 8$ codes. Crossovers also occur among the $t = 1$, 2, and 4 codes at smaller values of SNR per bit. Similarly, the curves for $t = 4$ and 8 and for $t = 8$ and 2 cross in the region of high SNR. This is the characteristic behavior for noncoherent detection of the coded waveforms.

If the demodulator does not make a hard decision on each symbol, but.

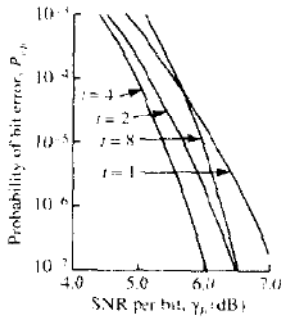


FIGURE 8-1-18 Performance of several $N = 31$, t-error correcting Reed–Solomon codes with 32-ary FSK modulation on an AWGN channel (noncoherent demodulation).

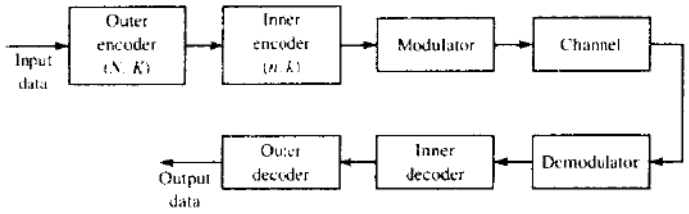


FIGURE 8-1-19 Block diagram of a communications system employing a concatenated code.

instead, passes the unquantized matched filter outputs to the decoder, soft-decision decoding can be performed. This decoding involves the formation of $q^K = 2^{kK}$ correlation metrics, where each metric corresponds to one of the q^K code words and consists of a sum of N matched filter outputs corresponding to the N code symbols. The matched filter outputs may be added coherently, or they may be envelope-detected and then added, or they may be square-law detected and then added. If coherent detection is used and the channel noise is AWGN, the computation of the probability of error is a straightforward extension of the binary case considered in Section 8-1-4. On the other hand, when envelope detection or square-law detection and noncoherent combining are used to form the decision variables, the computation of the decoder performance is considerably more complicated.

Concatenated Block Codes A concatenated code consists of two separate codes which are combined to form a larger code. Usually one code is selected to be nonbinary and the other is binary. The two codes are concatenated as illustrated in Fig. 8-1-19. The nonbinary (N, K) code forms the outer code and the binary code forms the inner code. Code words are formed by subdividing a block of kK information bits into K groups, called *symbols*, where each symbol consists of k bits. The K k-bit symbols are encoded into N k-bit symbols by the outer encoder, as is usually done with a nonbinary code. The inner encoder takes each k-bit symbol and encodes it into a binary block code of length n. Thus we obtain a concatenated block code having a block length of Nn bits and containing kK information bits. That is, we have created an equivalent (Nn, Kk) long binary code. The bits in each code word are transmitted over the channel by means of PSK or, perhaps, by FSK.

We also indicate that the minimum distance of the concatenated code is $d_{\min} D_{\min}$, where $D_{\min}$ is the minimum distance of the outer code and $d_{\min}$ is the minimum distance of the inner code. Furthermore, the rate of the concatenated code is Kk/Nn, which is equal to the product of the two code rates.

A hard-decision decoder for a concatenated code is conveniently separated into an inner decoder and an outer decoder. The inner decoder takes the hard decisions on each group of n bits, corresponding to a code word of the inner code, and makes a decision on the k information bits based on maximum-likelihood (minimum-distance) decoding. These k bits represent one symbol of

the outer code. When a block of N k-bit symbols are received from the inner decoder, the outer decoder makes a hard decision on the K k-bit symbols based on maximum-likelihood decoding.

Soft-decision decoding is also a possible alternative with a concatenated code. Usually, the soft-decision decoding is performed on the inner code, if it is selected to have relatively few code words, i.e., if 2^k is not too large. The outer code is usually decoded by means of hard-decision decoding, especially if the block length is long and there are many code words. On the other hand, there may be a significant gain in performance when soft-decision decoding is used on both the outer and inner codes, to justify the additional decoding complexity. This is the case in digital communications over fading channels, as we shall demonstrate in Chapter 14.

We conclude this subsection with the following example.

Example 8-1-14

Suppose that the (7, 4) Hamming code described in Examples 8-1-1 and 8-1-2 is used as the inner code in a concatenated code in which the outer code is a Reed–Solomon code. Since $k = 4$, we select the length of the Reed–Solomon code to be $N = 2^4 - 1 = 15$. The number of information symbols K per outer code word may be selected over the range $1 \leq K \leq 14$ in order to achieve a desired code rate.

8-1-9 Interleaving of Coded Data for Channels with Burst Errors

Most of the well-known codes that have been devised for increasing the reliability in the transmission of information are effective when the errors caused by the channel are statistically independent. This is the case for the AWGN channel. However, there are channels that exhibit bursty error characteristics. One example is the class of channels characterized by multipath and fading, which is described in detail in Chapter 14. Signal fading due to time-variant multipath propagation often causes the signal to fall below the noise level, thus resulting in a large number of errors. A second example is the class of magnetic recording channels (tape or disk) in which defects in the recording media result in clusters of errors. Such error clusters are not usually corrected by codes that are optimally designed for statistically independent errors.

Considerable work has been done on the construction of codes that are capable of correcting burst errors. Probably the best known burst error correcting codes are the subclass of cyclic codes called Fire codes, named after P. Fire (1959), who discovered them. Another class of cyclic codes for burst error correction were subsequently discovered by Burton (1969).

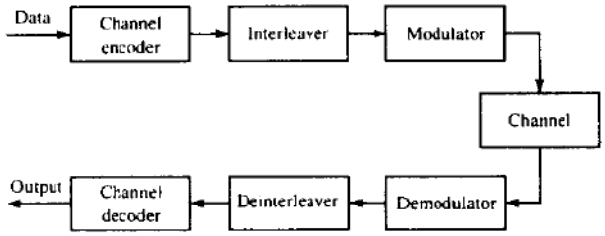


FIGURE 8-1-20 Block diagram of system employing interleaving for burst-error channel.

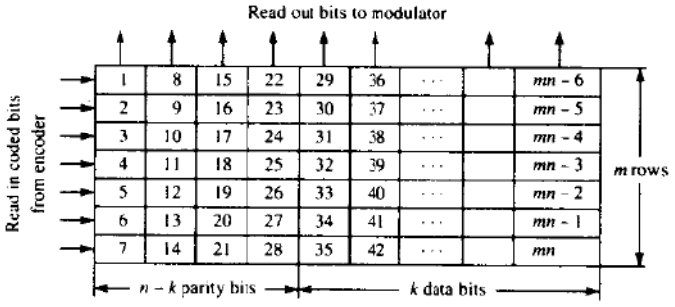
A *burst* of errors of length b is defined as a sequence of b-bit errors, the first and last of which are 1's. The *burst error correction capability* of a code is defined as one less than the length of the shortest uncorrectable burst. It is relatively easy to show that a systematic (n, k) code, which has $n - k$ parity check bits, can correct bursts of length $b \leq \lfloor \frac{1}{2}(n-k) \rfloor$.

An effective method for dealing with burst error channels is to interleave the coded data in such a way that the bursty channel is transformed into a channel having independent errors. Thus, a code designed for independent channel errors (short bursts) is used.

A block diagram of a system that employs interleaving is shown in Fig. 8-1-20. The encoded data are reordered by the interleaver and transmitted over the channel. At the receiver, after (either hard- or soft-decision) demodulation, the deinterleaver puts the data in proper sequence and passes it to the decoder. As a result of the interleaving/deinterleaving, error bursts are spread out in time so that errors within a code word appear to be independent.

The interleaver can take one of two forms: a block structure or a convolutional structure. A block *interleaver* formats the encoded data in a rectangular array of m rows and n columns. Usually, each row of the array constitutes a code word of length n. An *interleaver of degree m* consists of m rows (m code words) as illustrated in Fig. 8-1-21. The bits are read out

FIGURE 8-1-21 A block interleaver for coded data.



column-wise and transmitted over the channel. At the receiver, the deinterleaver stores the data in the same rectangular array format, but it is read out row-wise, one code word at a time. As a result of this reordering of the data during transmission, a burst of errors of length $l = mb$ is broken up into m bursts of length b. Thus, an (n, k) code that can handle burst errors of length $b \leq \lfloor \frac{1}{2}(n-k) \rfloor$ can be combined with an interleaver of degree m to create an interleaved (mn, mk) block code that can handle bursts of length mb.

A *convolutional interleaver* can be used in place of a block interleaver in much the same way. Convolutional interleavers are better matched for use with the class of convolutional codes that is described in the following section. Convolutional interleaver structures have been described by Ramsey (1970) and Forney (1971).

8-2 CONVOLUTIONAL CODES

A convolutional code is generated by passing the information sequence to be transmitted through a linear finite-state shift register. In general, the shift register consists of K (k-bit) stages and n linear algebraic function generators, as shown in Fig. 8-2-1. The input data to the encoder, which is assumed to be binary, is shifted into and along the shift register k bits at a time. The number of output bits for each k-bit input sequence is n bits. Consequently, the code rate is defined as $R_c = k/n$, consistent with the definition of the code rate for a block code. The parameter K is called the *constraint length* of the convolutional code.†

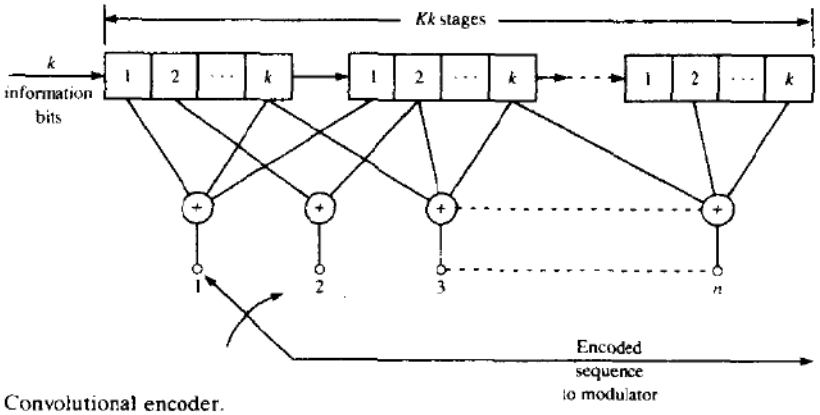


FIGURE 8-2-1 Convolutional encoder.

† In many cases, the constraint length of the code is given in bits rather than k-bit bytes. Hence the shift register may be called a *L-stage shift register*, where $L = Kk$. Furthermore, L may not be a multiple of k, in general.

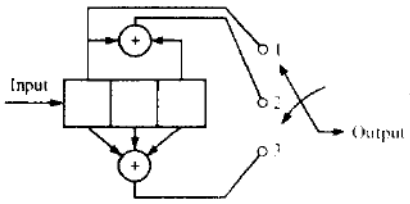


FIGURE 8-2-2 $K = 3$, $k = 1$, $n = 3$ convolutional encoder.

One method for describing a convolutional code is to give its generator matrix, just as we did for block codes. In general, the generator matrix for a convolutional code is semi-infinite since the input sequence is semi-infinite in length. As an alternative to specifying the generator matrix, we shall use a functionally equivalent representation in which we specify a set of n vectors, one vector for each of the n modulo-2 adders. Each vector has Kk dimensions and contains the connections of the encoder to that modulo-2 adder. A 1 in the ith position of the vector indicates that the corresponding stage in the shift register is connected to the modulo-2 adder and a 0 in a given position indicates that no connection exists between that stage and the modulo-2 adder.

To be specific, let us consider the binary convolutional encoder with constraint length $K = 3$, $k = 1$, and $n = 3$, which is shown in Fig. 8-2-2. Initially, the shift register is assumed to be in the all-zero state. Suppose the first input bit is a 1. Then the output sequence of 3 bits is 111. Suppose the second bit is a 0. The output sequence will then be 001. If the third bit is a 1, the output will be 100, and so on. Now, suppose we number the outputs of the function generators that generate each three-bit output sequence as 1, 2, and 3, from top to bottom, and similarly number each corresponding function generator. Then, since only the first stage is connected to the first function generator (no modulo-2 adder is needed), the generator is

$$\mathbf{g}_1 = [100]$$

The second function generator is connected to stages 1 and 3. Hence

$$\mathbf{g}_2 = [101]$$

Finally,

$$\mathbf{g}_3 = [111]$$

The generators for this code are more conveniently given in octal form as $(4, 5, 7)$. We conclude that, when $k = 1$, we require n generators, each of dimension K to specify the encoder.

For a rate k/n binary convolutional code with $k > 1$ and constraint length K, the n generators are Kk-dimensional vectors, as stated above. The following example illustrates the case in which $k = 2$ and $n = 3$.

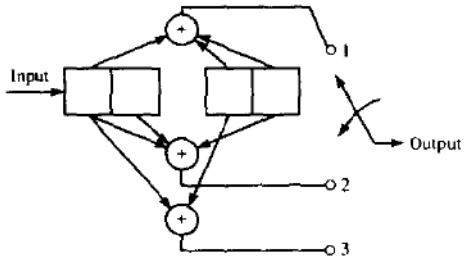


FIGURE 8-2-3 $K = 2$, $k = 2$, $n = 3$ convolutional encoder.

Example 8-2-1

Consider the rate 2/3 convolutional encoder illustrated in Fig. 8-2-3. In this encoder, two bits at a time are shifted into it and three output bits are generated. The generators are

$$\mathbf{g}_1 = [1011], \quad \mathbf{g}_2 = [1101], \quad \mathbf{g}_3 = [1010]$$

In octal form, these generators are (13, 15, 12).

There are three alternative methods that are often used to describe a convolutional code. These are the tree diagram, the trellis diagram, and the state diagram. For example, the tree diagram for the convolutional encoder shown in Fig. 8-2-2 is illustrated in Fig. 8-2-4. Assuming that the encoder is in the all-zero state initially, the diagram shows that, if the first input bit is a 0, the output sequence is 000 and, if the first bit is a 1; the output sequence is 111. Now, if the first input bit is a 1 and the second bit is a 0, the second set of three output bits is 001. Continuing through the tree, we see that if the third bit is a

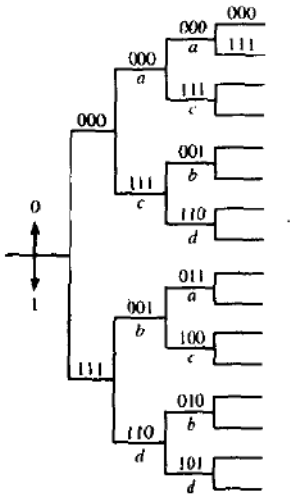
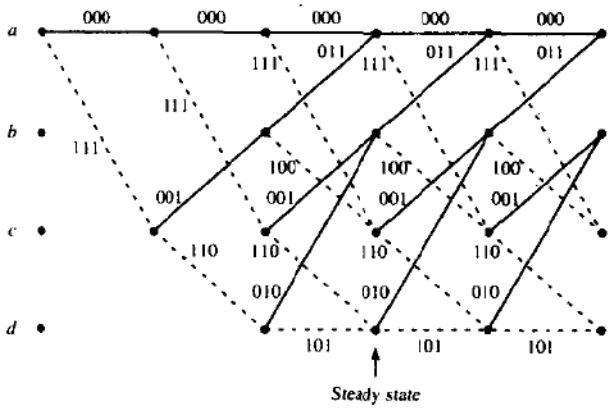


FIGURE 8-2-4 Tree diagram for rate 1/3, $K = 3$ convolutional code.

0 then the output is 011, while if the third bit is a 1 then the output is 100. Given that a particular sequence has taken us to a particular node in the tree, the branching rule is to follow the upper branch if the next input bit is a 0 and the lower branch if the bit is a 1. Thus, we trace a particular path through the tree that is determined by the input sequence.

Close observation of the tree that is generated by the convolutional encoder shown in Fig. 8-2-2 reveals that the structure repeats itself after the third stage. This behavior is consistent with the fact that the constraint length $K = 3$. That is, the three-bit output sequence at each stage is determined by the input bit and the two previous input bits, i.e., the two bits contained in the first two stages of the shift register. The bit in the last stage of the shift register is shifted out at the right and does not affect the output. Thus we may say that the three-bit output sequence for each input bit is determined by the input bit and the four possible states of the shift register, denoted as $a = 00$, $b = 01$, $c = 10$, $d = 11$. If we label each node in the tree to correspond to the four possible states in the shift register, we find that at the third stage there are two nodes with the label a, two with the label b, two with the label c, and two with the label d. Now we observe that all branches emanating from two nodes having the same label (same state) are identical in the sense that they generate identical output sequences. This means that the two nodes having the same label can be merged. If we do this to the tree shown in Fig. 8-2-4, we obtain another diagram, which is more compact, namely, a *trellis*. For example, the trellis diagram for the convolutional encoder of Fig. 8-2-2 is shown in Fig. 8-2-5. In drawing this diagram, we use the convention that a solid line denotes the output generated by the input bit 0 and a dotted line the output generated by the input bit 1. In the example being considered, we observe that, after the initial transient, the trellis contains four nodes at each stage, corresponding to the four states of the shift register, a, b, c, and d. After the second stage, each node in the trellis has two incoming paths and two outgoing paths. Of the two

FIGURE 8-2-5 Trellis diagram for rate 1/3, $K = 3$ convolutional code.



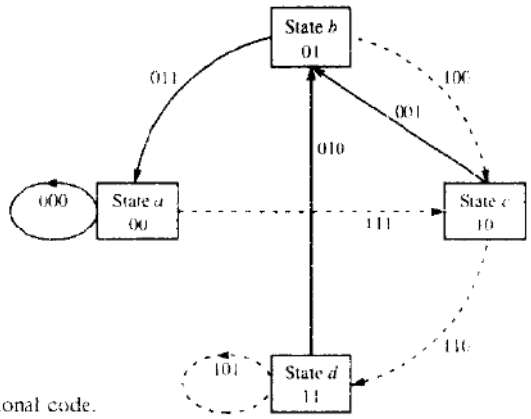


FIGURE 8-2-6 State diagram for rate 1/3, $K = 3$ convolutional code.

outgoing paths, one corresponds to the input bit 0 and the other to the path followed if the input bit is a 1.

Since the output of the encoder is determined by the input and the state of the encoder, an even more compact diagram than the trellis is the state diagram. The state diagram is simply a graph of the possible states of the encoder and the possible transitions from one state to another. For example the state diagram for the encoder shown in Fig. 8-2-2 is illustrated in Fig. 8-2-6. This diagram shows that the possible transitions are

$$a \xrightarrow{0} a,\ a \xrightarrow{1} c,\ b \xrightarrow{0} a,\ b \xrightarrow{1} c,\ c \xrightarrow{0} b,\ c \xrightarrow{1} d,\ d \xrightarrow{0} b,\ d \xrightarrow{1} d,$$

where $\alpha \xrightarrow{1} \beta$ denotes the transition from state α to β when the input bit is a 1. The three bits shown next to each branch in the state diagram represent the output bits. A dotted line in the graph indicates that the input bit is a 1, while the solid line indicates that the input bit is a 0.

Example 8-2-2

Let us consider the $k = 2$, rate 2/3 convolutional code described in Example 8-2-1 and shown in Fig. 8-2-3. The first two input bits may be 00, 01, 10, or 11. The corresponding output bits are 000, 010, 111, 101. When the next pair of input bits enter the encoder, the first pair is shifted to the second stage. The corresponding output bits depend on the pair of bits shifted into the second stage and the new pair of input bits. Hence, the tree diagram for this code, shown in Fig. 8-2-7, has four branches per node, corresponding to the four possible pairs of input symbols. Since the constraint length of the code is $K = 2$, the tree begins to repeat after the second stage. As illustrated in Fig. 8-2-7, all the branches emanating from nodes labeled a (state a) yield identical outputs. By merging the nodes having identical labels, we obtain the trellis, which is shown in Fig. 8-2-8. Finally, the state diagram for this code is shown in Fig. 8-2-9.

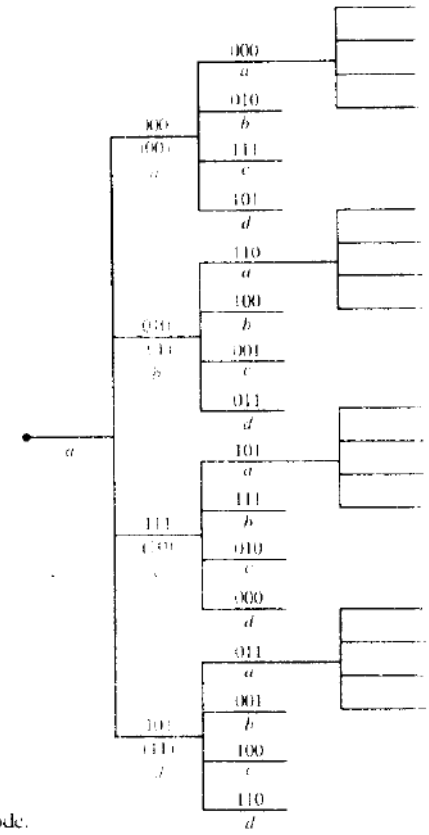


FIGURE 8-2-7 Tree diagram for $K = 2$, $k = 2$, $n = 3$ convolutional code.

To generalize, we state that a rate k/n, constraint length K, convolutional code is characterized by 2^k branches emanating from each node of the tree diagram. The trellis and the state diagrams each have $2^{k(K-1)}$ possible states. There are 2^k branches entering each state and 2^k branches leaving each state (in the trellis and tree, this is true after the initial transient).

The three types of diagrams described above are also used to represent nonbinary convolutional codes. When the number of symbols in the code alphabet is $q = 2^k$, $k > 1$, the resulting nonbinary code may also be represented as an equivalent binary code. The following example considers a convolutional code of this type.

Example 8-2-3

Let us consider the convolutional code generated by the encoder shown in Fig. 8-2-10. This code may be described as a binary convolutional code with parameters $K = 2$, $k = 2$, $n = 4$, $R_c = 1/2$, and having the generators

$$\mathbf{g}_1 = [1010], \quad \mathbf{g}_2 = [0101], \quad \mathbf{g}_3 = [1110], \quad \mathbf{g}_4 = [1001]$$

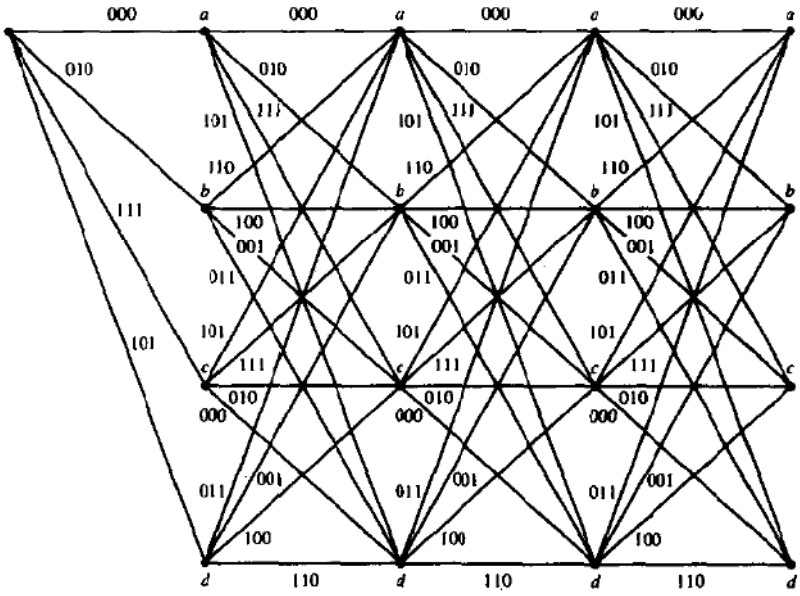
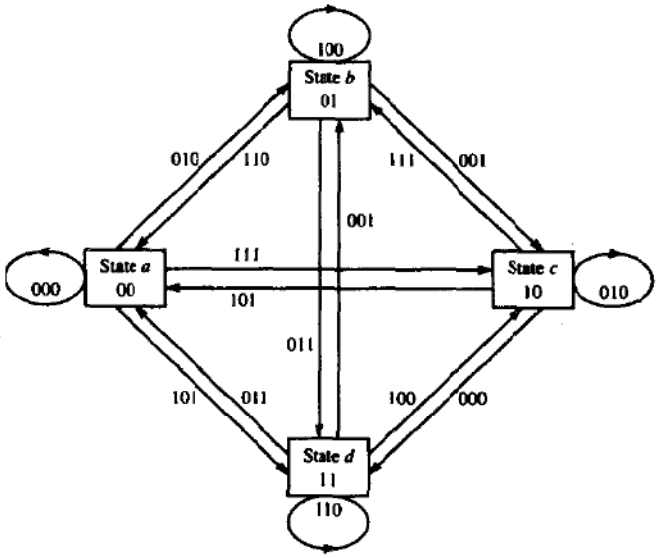


FIGURE 8-2-8 Trellis diagram for $K = 2$, $k = 2$, $n = 3$ convolutional code.

FIGURE 8-2-9 State diagram for $K = 2$, $k = 2$, $n = 3$ convolutional code.



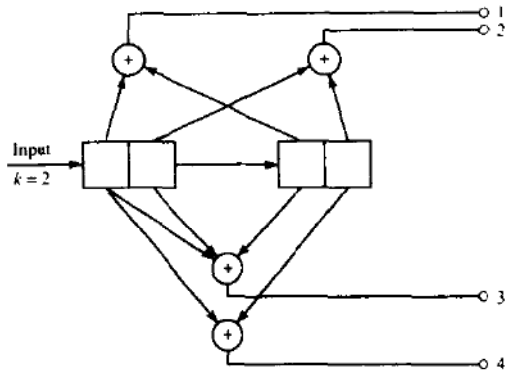


FIGURE 8-2-10 $K = 2$, $k = 2$, $n = 4$ convolutional encoder.

Except for the difference in rate, this code is similar in form to the rate 2/3, $k = 2$ convolutional code considered in Example 8-2-1.

Alternatively, the code generated by the encoder in Fig. 8-2-10 may be described as a nonbinary ($q = 4$) code with one quaternary symbol as an input and two quaternary symbols as an output. In fact, if the output of the encoder is treated by the modulator and demodulator as q-ary ($q = 4$) symbols that are transmitted over the channel by means of some M-ary ($M = 4$) modulation technique, the code is appropriately viewed as nonbinary.

In any case, the tree, the trellis, and the state diagrams are independent of how we view the code. That is, this particular code is characterized by a tree with four branches emanating from each node, or a trellis with four possible states and four branches entering and leaving each state or, equivalently, by a state diagram having the same parameters as the trellis.

8-2-1 The Transfer Function of a Convolutional Code

The distance properties and the error rate performance of a convolutional code can be obtained from its state diagram. Since a convolutional code is linear, the set of Hamming distances of the code sequences generated up to some stage in the tree, from the all-zero code sequence, is the same as the set of distances of the code sequences with respect to any other code sequence. Consequently, we assume without loss of generality that the all-zero code sequence is the input to the encoder.

The state diagram shown in Fig. 8-2-6 will be used to demonstrate the method for obtaining the distance properties of a convolutional code. First, we label the branches of the state diagram as either $D^0 = 1$, D^1, D^2, or D^3, where the exponent of D denotes the Hamming distance of the sequence of output bits corresponding to each branch from the sequence of output bits corresponding to the all-zero branch. The self-loop at node a can be eliminated, since it contributes nothing to the distance properties of a code sequence relative to

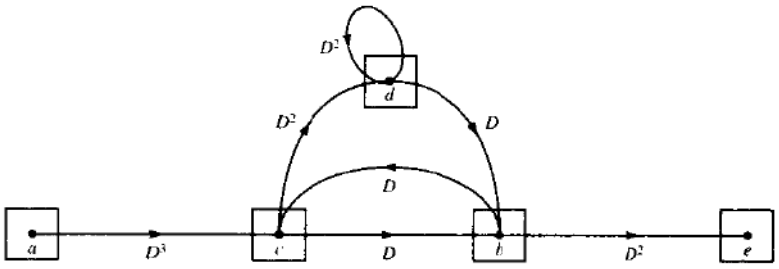


FIGURE 8-2-11 State diagram for rate 1/3, $K = 3$ convolutional code.

the all-zero code sequence. Furthermore, node a is split into two nodes, one of which represents the input and the other the output of the state diagram. Figure 8-2-11 illustrates the resulting diagram. We use this diagram, which now consists of five nodes because node a was split into two, to write the four state equations

$$\begin{aligned} X_c &= D^3 X_a + D X_b \\ X_b &= D X_c + D X_d \\ X_d &= D^2 X_c + D^2 X_d \\ X_e &= D^2 X_b \end{aligned} \tag{8-2-1}$$

The transfer function for the code is defined as $T(D) = X_e/X_a$. By solving the state equations given above, we obtain

$$\begin{aligned} T(D) &= \frac{D^6}{1 - 2D^2} \\ &= D^6 + 2D^8 + 4D^{10} + 8D^{12} + \cdots \\ &= \sum_{d=6}^{\infty} a_d D^d \end{aligned} \tag{8-2-2}$$

where, by definition,

$$a_d = \begin{cases} 2^{(d-6)/2} & (\text{even } d) \\ 0 & (\text{odd } d) \end{cases} \tag{8-2-3}$$

The transfer function for this code indicates that there is a single path of Hamming distance $d = 6$ from the all-zero path that merges with the all-zero path at a given node. From the state diagram shown in Fig. 8-2-6 or the trellis diagram shown in Fig. 8-2-5, it is observed that the $d = 6$ path is *acbe*. There is no other path from node a to node e having a distance $d = 6$. The second term in (8-2-2) indicates that there are two paths from node a to node e having a

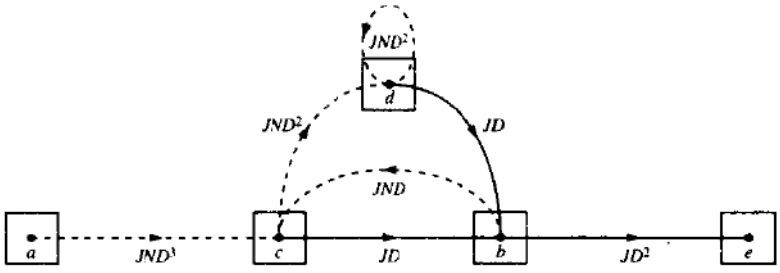


FIGURE 8-2-12 State diagram for rate 1/3, $K = 3$ convolutional code.

distance $d = 8$. Again, from the state diagram or the trellis, we observe that these paths are *acdbe* and *acbcbe*. The third term in (8-2-2) indicates that there are four paths of distance $d = 10$, and so forth. Thus the transfer function gives us the distance properties of the convolutional code. The minimum distance of the code is called the *minimum free distance* and denoted by d_{free}. In our example, $d_{\text{free}} = 6$.

The transfer function can be used to provide more detailed information than just the distance of the various paths. Suppose we introduce a factor N into all branch transitions caused by the input bit 1. Thus, as each branch is traversed, the cumulative exponent on N increases by one only if that branch transition is due to an input bit 1. Furthermore, we introduce a factor of J into each branch of the state diagram so that the exponent of J will serve as a counting variable to indicate the number of branches in any given path from node a to node e. For the rate 1/3 convolutional code in our example, the state diagram that incorporates the additional factors of J and N is shown in Fig. 8-2-12.

The state equations for the state diagram shown in Fig. 8-2-12 are

$$\begin{aligned} X_c &= JND^3X_a + JNDX_b \\ X_b &= JDX_c + JDX_d \\ X_d &= JND^2X_c + JND^2X_d \\ X_e &= JD^2X_b \end{aligned} \tag{8-2-4}$$

Upon solving these equations for the ratio X_e/X_a, we obtain the transfer function

$$\begin{aligned} T(D, N, J) &= \frac{J^3ND^6}{1 - JND^2(1 + J)} \\ &= J^3ND^6 + J^4N^2D^8 + J^5N^2D^8 + J^5N^3D^{10} \\ &\quad + 2J^6N^3D^{10} + J^7N^3D^{10} + \dots \end{aligned} \tag{8-2-5}$$

This form for the transfer functions gives the properties of all the paths in

the convolutional code. That is, the first term in the expansion of $T(D, N, J)$ indicates that the distance $d = 6$ path is of length 3 and of the three information bits, one is a 1. The second and third terms in the expansion of $T(D, N, J)$ indicate that of the two $d = 8$ terms, one is of length 4 and the second has length 5. Two of the four information bits in the path having length 4 and two of the five information bits in the path having length 5 are 1s. Thus, the exponent of the factor J indicates the length of the path that merges with the all-zero path for the first time, the exponent of the factor N indicates the number of 1s in the information sequence for that path, and the exponent of D indicates the distance of the sequence of encoded bits for that path from the all-zero sequence.

The factor J is particularly important if we are transmitting a sequence of finite duration, say m bits. In such a case, the convolutional code is truncated after m nodes or m branches. This implies that the transfer function for the truncated code is obtained by truncating $T(D, N, J)$ at the term J^m. On the other hand, if we are transmitting an extremely long sequence, i.e., essentially an infinite-length sequence, we may wish to suppress the dependence of $T(D, N, J)$ on the parameter J. This is easily accomplished by setting $J = 1$. Hence, for the example given above, we have

$$\begin{aligned} T(D, N, 1) = T(D, N) &= \frac{ND^6}{1 - 2ND^2} \\ &= ND^6 + 2N^2D^8 + 4N^3D^{10} + \cdots \\ &= \sum_{d=6}^{\infty} a_d N^{(d-4)/2} D^d \end{aligned} \tag{8-2-6}$$

where the coefficients $\{a_d\}$ are defined by (8-2-3).

The procedure outlined above for determining the transfer function of a binary convolutional code is easily extended to nonbinary codes. In the following example, we determine the transfer function of the nonbinary convolutional code previously introduced in Example 8-2-3.

Example 8-2-4

The convolutional code shown in Fig. 8-2-10 has the parameters $K = 2$, $k = 2$, $n = 4$. In this example, we have a choice of how we label distances and count errors, depending on whether we treat the code as binary or nonbinary. Suppose we treat the code as nonbinary. Thus, the input to the encoder and the output are treated as quaternary symbols. In particular, if we treat the input and output as quaternary symbols 00, 01, 10, and 11, the distance measured in symbols between the sequences 0111 and 0000 is 2. Furthermore, suppose that an input symbol 00 is decoded as the symbol 11; then we have made one symbol error. This convention applied to the

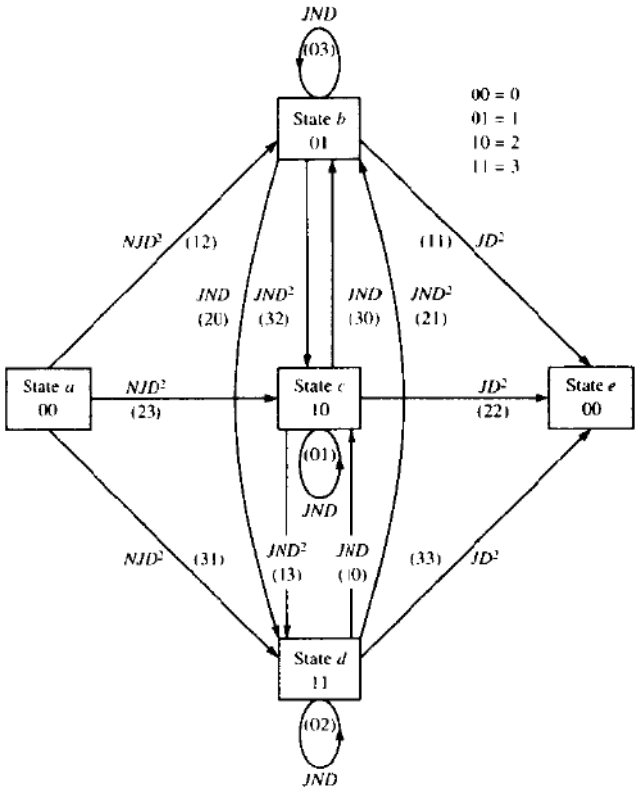


FIGURE 8-2-13 State diagram for $K = 2$, $k = 2$, rate 1/2 nonbinary code.

convolutional code shown in Fig. 8-2-10 results in the state diagram illustrated in Fig. 8-2-13, from which we obtain the state equations

$$\begin{aligned} X_b &= NJD^2X_a + NJDX_b + NJDX_c + NJD^2X_d \\ X_c &= NJD^2X_a + NJD^2X_b + NJDX_c + NJDX_d \\ X_d &= NJD^2X_a + NJDX_b + NJD^2X_c + NJDX_d \\ X_e &= JD^2(X_b + X_c + X_d) \end{aligned} \tag{8-2-7}$$

Solution of these equations leads to the transfer function

$$T(D, N, J) = \frac{3NJ^2D^4}{1 - 2NJD - NJD^2} \tag{8-2-8}$$

This expression for the transfer function is particularly appropriate when the quaternary symbols at the output of the encoder are mapped into a

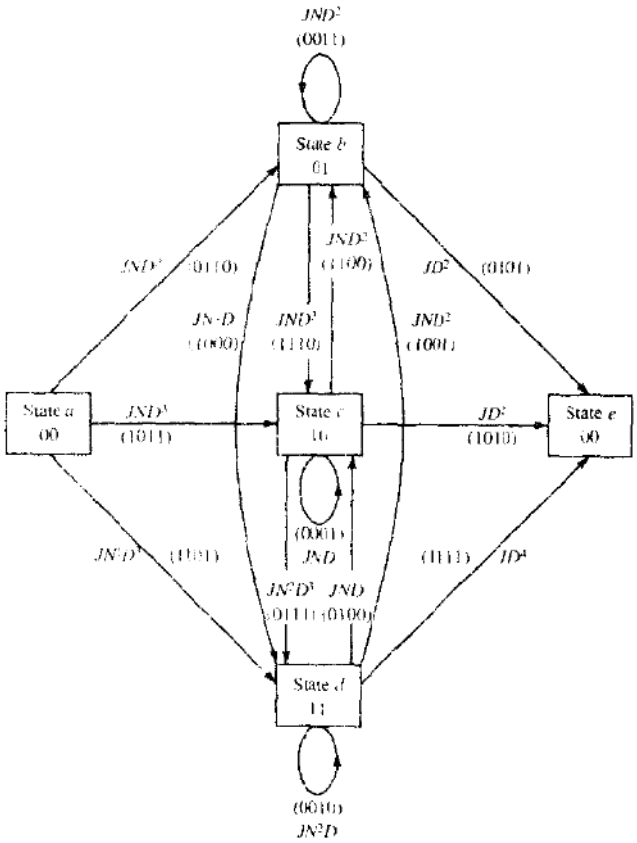


FIGURE 8-2-14 State diagram for $K = 2$, $k = 2$, rate 1/2 convolutional code with output treated as a binary sequence.

corresponding set of quaternary waveforms $s_m(t)$, $m = 1, 2, 3, 4$, e.g., four orthogonal waveforms. Thus, there is a one-to-one correspondence between code symbols and signal waveforms.

Alternatively, for example, the output of the encoder may be transmitted as a sequence of binary digits by means of binary PSK. In such a case, it is appropriate to measure distance in terms of bits. When this convention is employed, the state diagram is labeled as shown in Fig. 8-2-14. Solution of the state equations obtained from this state diagram yields a transfer function that is different from the one given in (8-2-8).

Some convolutional codes exhibit a characteristic behavior that is called *catastrophic error propagation*. When a code that has this characteristic is used on a binary symmetric channel, it is possible for a finite number of channel errors to cause an infinite number of decoding errors. Such a code can be

identified from its state diagram. It will contain a zero-distance path (a path with multiplier $D^0 = 1$) from some nonzero state back to the same state. This means that one can loop around this zero-distance path an infinite number of times without increasing the distance relative to the all-zero path. But, if this self-loop corresponds to the transmission of a 1, the decoder will make an infinite number of errors. Since such codes are easily recognized, they are easily avoided in practice.

8-2-2 Optimum Decoding of Convolutional Codes—The Viterbi Algorithm

In the decoding of a block code for a memoryless channel, we computed the distances (Hamming distance for hard-decision decoding and euclidean distance for soft-decision decoding) between the received code word and the 2^k possible transmitted code words. Then we selected the code word that was closest in distance to the received code word. This decision rule, which requires the computation of 2^k *metrics*, is optimum in the sense that it results in a minimum probability of error for the binary symmetric channel with $p < \frac{1}{2}$ and the additive white gaussian noise channel.

Unlike a block code, which has a fixed length n, a convolutional encoder is basically a finite-state machine. Hence the optimum decoder is a maximum-likelihood sequence estimator (MLSE) of the type described in Section 5-1-4 for signals with memory, such as NRZI and CPM. Therefore, optimum decoding of a convolutional code involves a search through the trellis for the most probable sequence. Depending on whether the detector following the demodulator performs hard or soft decisions, the corresponding metric in the trellis search may be either a Hamming metric or a euclidean metric, respectively. We elaborate below, using the trellis in Fig. 8-2-5 for the convolutional code shown in Fig. 8-2-2.

Consider the two paths in the trellis that begin at the initial state a and remerge at state a after three state transitions (three branches), corresponding to the two information sequences 000 and 100 and the transmitted sequences 000 000 000 and 111 001 011, respectively. We denote the transmitted bits by $\{c_{jm}, j = 1, 2, 3; m = 1, 2, 3\}$, where the index j indicates the jth branch and the index m the mth bit in that branch. Correspondingly, we define $\{r_{jm}, j = 1, 2, 3; m = 1, 2, 3\}$ as the output of the demodulator. If the detector performs hard-decision decoding, its output for each transmitted bit is either 0 or 1. On the other hand, if soft-decision decoding is employed and the coded sequence is transmitted by binary coherent PSK, the input to the decoder is

$$r_{jm} = \sqrt{\mathcal{E}_c}(2c_{jm} - 1) + n_{jm} \tag{8-2-9}$$

where n_{jm} represents the additive noise and $\mathcal{E}_c$ is the transmitted signal energy for each code bit.

A metric is defined for the jth branch of the ith path through the trellis as the logarithm of the joint probability of the sequence $\{r_{jm}, m = 1, 2, 3\}$

conditioned on the transmitted sequence $\{c_{jm}^{(i)},\ m = 1, 2, 3\}$ for the ith path. That is,

$$\mu_j^{(i)} = \log P(\mathbf{Y}_j \mid \mathbf{C}_j^{(i)}), \quad j = 1, 2, 3, \ldots \tag{8-2-10}$$

Furthermore, a metric for the ith path consisting of B branches through the trellis is defined as

$$PM^{(i)} = \sum_{j=1}^{B} \mu_j^{(i)} \tag{8-2-11}$$

The criterion for deciding between two paths through the trellis is to select the one having the larger metric. This rule maximizes the probability of a correct decision or, equivalently, it minimizes the probability of error for the sequence of information bits. For example, suppose that hard-decision decoding is performed by the demodulator, yielding the received sequence {101 000 100}. Let $i = 0$ denote the three-branch all-zero path and $i = 1$ the second three-branch path that begins in the initial state a and remerges with the all-zero path at state a after three transitions. The metrics for these two paths are

$$\begin{aligned} PM^{(0)} &= 6 \log (1 - p) + 3 \log p \\ PM^{(1)} &= 4 \log (1 - p) + 5 \log p \end{aligned} \tag{8-2-12}$$

where p is the probability of a bit error. Assuming that $p < \frac{1}{2}$, we find that the metric $PM^{(0)}$ is larger than the metric $PM^{(1)}$. This result is consistent with the observation that the all-zero path is at Hamming distance $d = 3$ from the received sequence, while the $i = 1$ path is at Hamming distance $d = 5$ from the received path. Thus, the Hamming distance is an equivalent metric for hard-decision decoding.

Similarly, suppose that soft-decision decoding is employed and the channel adds white gaussian noise to the signal. Then the demodulator output is described statistically by the probability density function

$$p(r_{jm} \mid c_{jm}^{(i)}) = \frac{1}{\sqrt{2\pi}\,\sigma} \exp\left\{ -\frac{[r_{jm} - \sqrt{\mathscr{E}_c}(2c_{jm}^{(i)} - 1)]^2}{2\sigma^2} \right\} \tag{8-2-13}$$

where $\sigma^2 = \frac{1}{2}N_0$ is the variance of the additive gaussian noise. If we neglect the terms that are common to all branch metrics, the branch metric for the jth branch of the ith path may be expressed as

$$\mu_j^{(i)} = \sum_{m=1}^{n} r_{jm}(2c_{jm}^{(i)} - 1) \tag{8-2-14}$$

where, in our example, $n = 3$. Thus the correlation metrics for the two paths under consideration are

$$\begin{aligned} CM^{(0)} &= \sum_{j=1}^{3} \sum_{m=1}^{3} r_{jm}(2c_{jm}^{(0)} - 1) \\ CM^{(1)} &= \sum_{j=1}^{3} \sum_{m=1}^{3} r_{jm}(2c_{jm}^{(1)} - 1) \end{aligned} \tag{8-2-15}$$

Having defined the branch metrics and path metrics computed by the decoder, we now consider the use of the Viterbi algorithm for optimum decoding of the convolutionally encoded information sequence. We consider the two paths described above, which merge at state a after three transitions. Note that any particular path through the trellis that stems from this node will add identical terms to the path metrics $CM^{(0)}$ and $CM^{(1)}$. Consequently, if $CM^{(0)} > CM^{(1)}$ at the merged node a after three transitions $CM^{(0)}$ will continue to be larger than $CM^{(1)}$ for any path that stems from node a. This means that the path corresponding to $CM^{(1)}$ can be discarded from further consideration. The path corresponding to the metric $CM^{(0)}$ is the *survivor*. Similarly, one of the two paths that merge at state b can be elminated on the basis of the two corresponding metrics. This procedure is repeated at state c and state d. As a result, after the first three transitions, there are four surviving paths, one terminating at each state, and a corresponding metric for each survivor. This procedure is repeated at each stage of the trellis as new signals are received in subsequent time intervals.

In general, when a binary convolutional code with $k = 1$ and constraint length K is decoded by means of the Viterbi algorithm, there are 2^{K-1} states. Hence, there are 2^{K-1} surviving paths at each stage and 2^{K-1} metrics, one for each surviving path. Furthermore, a binary convolutional code in which k bits at a time are shifted into an encoder that consists of K (k-bit) shift-register stages generates a trellis that has $2^{k(K-1)}$ states. Consequently, the decoding of such a code by means of the Viterbi algorithm requires keeping track of $2^{k(K-1)}$ surviving paths and $2^{k(K-1)}$ metrics. At each stage of the trellis, there are 2^k paths that merge at each node. Since each path that converges at a common node requires the computation of a metric, there are 2^k metrics computed for each node. Of the 2^k paths that merge at each node, only one survives, and this is the most-probable (minimum-distance) path. Thus the number of computations in decoding performed at each stage increases exponentially with k and K. The exponential increase in computational burden limits the use of the Viterbi algorithm to relatively small values of K and k.

The decoding delay in decoding a long information sequence that has been convolutionally encoded is usually too long for most practical applications. Moreover, the memory required to store the entire length of surviving sequences is large and expensive. As indicated in Section 5-1-4, a solution to this problem is to modify the Viterbi algorithm in a way which results in a fixed decoding delay without significantly affecting the optimal performance of the algorithm. Recall that the modification is to retain at any given time t only the most recent δ decoded information bits (symbols) in each surviving sequence. As each new information bit (symbol) is received, a final decision is made on the bit (symbol) received δ branches back in the trellis, by comparing the metrics in the surviving sequences and deciding in favor of the bit in the sequence having the largest metric. If δ is chosen sufficiently large, all surviving sequences will contain the identical decoded bit (symbol) δ branches back in time. That is, with high probability, all surviving sequences at time t stem from

the same node at $t - \delta$. It has been found experimentally (computer simulation) that a delay $\delta \geq 5K$ results in a negligible degradation in the performance relative to the optimum Viterbi algorithm.

8-2-3 Probability of Error for Soft-Decision Decoding

The topic of this subsection is the error rate performance of the Viterbi algorithm on an additive white gaussian noise channel with soft-decision decoding.

In deriving the probability of error for convolutional codes, the linearity property for this class of codes is employed to simplify the derivation. That is, we assume that the all-zero sequence is transmitted and we determine the probability of error in deciding in favor of another sequence. The coded binary digits for the jth branch of the convolutional code, denoted as $\{c_{jm}, m = 1, 2, \ldots, n\}$ and defined in Section 8-2-2, are assumed to be transmitted by binary PSK (or four-phase PSK) and detected coherently at the demodulator. The output of the demodulator, which is the input to the Viterbi decoder, is the sequence $\{r_{jm}, m = 1, 2, \ldots, n; j = 1, 2, \ldots\}$ where r_{jm} is defined in (8-2-9).

The Viterbi soft-decision decoder forms the branch metrics defined by (8-2-14) and from these computes the path metrics

$$CM^{(i)} = \sum_{j=1}^{B} \mu_j^{(i)} = \sum_{j=1}^{B} \sum_{m=1}^{n} r_{jm}(2c_{jm}^{(i)} - 1) \tag{8-2-16}$$

where i denotes any one of the competing paths at each node and B is the number of branches (information symbols) in a path. For example, the all-zero path, denoted as $i = 0$, has a path metric

$$\begin{aligned} CM^{(0)} &= \sum_{j=1}^{B} \sum_{m=1}^{n} (-\sqrt{\mathscr{E}_c} + n_{jm})(-1) \\ &= \sqrt{\mathscr{E}_c}\, Bn + \sum_{j=1}^{B} \sum_{m=1}^{n} n_{jm} \end{aligned} \tag{8-2-17}$$

Since the convolutional code does not necessarily have a fixed length, we derive its performance from the probability of error for sequences that merge with the all-zero sequence for the first time at a given node in the trellis. In particular, we define the first-event error probability as the probability that another path that merges with the all-zero path at node B has a metric that exceeds the metric of the all-zero path for the first time. Suppose the incorrect path, call it $i = 1$, that merges with the all-zero path differs from the all-zero path in d bits, i.e., there are d 1s in the path $i = 1$ and the rest are 0s. The probability of error in the pairwise comparison of the metrics $CM^{(0)}$ and $CM^{(1)}$ is

$$\begin{aligned} P_2(d) &= P(CM^{(1)} \geq CM^{(0)}) = P(CM^{(1)} - CM^{(0)} \geq 0) \\ P_2(d) &= P\left[2 \sum_{j=1}^{B} \sum_{m=1}^{n} r_{jm}(c_{jm}^{(1)} - c_{jm}^{(0)}) \geq 0\right] \end{aligned} \tag{8-2-18}$$

Since the coded bits in the two paths are identical except in the d positions, (8-2-18) can be written in the simpler form

$$P_2(d) = P\left(\sum_{l=1}^{d} r_l' \geq 0\right) \tag{8-2-19}$$

where the index l runs over the set of d bits in which the two paths differ and the set $\{r_l'\}$ represents the input to the decoder for these d bits.

The $\{r_l'\}$ are independent and identically distributed gaussian random variables with mean $-\sqrt{\mathscr{E}_c}$ and variance $\frac{1}{2}N_0$. Consequently the probability of error in the pairwise comparison of these two paths that differ in d bits is

$$\begin{aligned} P_2(d) &= Q\left(\sqrt{\frac{2\mathscr{E}_c}{N_0}d}\right) \\ &= Q(\sqrt{2\gamma_b R_c d}) \end{aligned} \tag{8-2-20}$$

where $\gamma_b = \mathscr{E}_b/N_0$ is the received SNR per bit and R_c is the code rate.

Although we have derived the first-event error probability for a path of distance d from the all-zero path, there are many possible paths with different distances that merge with the all-zero path at a given node B. In fact, the transfer function $T(D)$ provides a complete description of all the possible paths that merge with the all-zero path at node B and their distances. Thus we can sum the error probability in (8-2-20) over all possible path distances. Upon performing this summation, we obtain an upper bound on the first-event error probability in the form

$$\begin{aligned} P_e &\leq \sum_{d=d_{\text{free}}}^{\infty} a_d P_2(d) \\ &\leq \sum_{d=d_{\text{free}}}^{\infty} a_d Q(\sqrt{2\gamma_b R_c d}) \end{aligned} \tag{8-2-21}$$

where a_d denotes the number of paths of distance d from the all-zero path that merge with the all-zero path for the first time.

There are two reasons why (8-2-21) is an upper bound on the first-event error probability. One is that the events that result in the error probabilities $\{P_2(d)\}$ are not disjoint. This can be seen from observation of the trellis. Second, by summing over all possible $d \geq d_{\text{free}}$, we have implicitly assumed that the convolutional code has infinite length. If the code is truncated periodically after B nodes, the upper bound in (8-2-21) can be improved by summing the error events for $d_{\text{free}} \leq d \leq B$. This refinement has some merit in determining the performance of short convolutional codes, but the effect on performance is negligible when B is large.

The upper bound in (8-2-21) can be expressed in a slightly different form if the Q function is upper-bounded by an exponential. That is,

$$Q(\sqrt{2\gamma_b R_c d}) \leq e^{-\gamma_b R_c d} = D^d\big|_{D=e^{-\gamma_b R_c}} \tag{8-2-22}$$

If we use (8-2-22) in (8-2-21), the upper bound on the first-event error probability can be expressed as

$$P_e < T(D)\big|_{D=e^{-\gamma_b R_c}} \tag{8-2-23}$$

Although the first-event error probability provides a measure of the performance of a convolutional code, a more useful measure of performance is the bit error probability. This probability can be upper-bounded by the procedure used in bounding the first-event error probability. Specifically, we know that when an incorrect path is selected, the information bits in which the selected path differs from the correct path will be decoded incorrectly. We also know that the exponents in the factor N contained in the transfer function $T(D, N)$ indicate the number of information bit errors (number of 1s) in selecting an incorrect path that merges with the all-zero path at some node B. If we multiply the pairwise error probability $P_2(d)$ by the number of incorrectly decoded information bits for the incorrect path at the node where they merge, we obtain the bit error rate for that path. The average bit error probability is upper-bounded by multiplying each pairwise error probability $P_2(d)$ by the corresponding number of incorrectly decoded information bits, for each possible incorrect path that merges with the correct path at the Bth node, and summing over all d.

The appropriate multiplication factors corresponding to the number of information bit errors for each incorrectly selected path may be obtained by differentiating $T(D, N)$ with respect to N. In general, $T(D, N)$ can be expressed as

$$T(D, N) = \sum_{d=d_{\text{free}}}^{\infty} a_d D^d N^{f(d)} \tag{8-2-24}$$

where $f(d)$ denotes the exponent of N as a function of d. Taking the derivative of $T(D, N)$ with respect to N and setting $N = 1$, we obtain

$$\begin{aligned} \left.\frac{dT(D, N)}{dN}\right|_{N=1} &= \sum_{d=d_{\text{free}}}^{\infty} a_d f(d) D^d \\ &= \sum_{d=d_{\text{free}}}^{\infty} \beta_d D^d \end{aligned} \tag{8-2-25}$$

where $\beta_d = a_d f(d)$. Thus the bit error probability for $k = 1$ is upper-bounded by

$$\begin{aligned} P_b &< \sum_{d=d_{\text{free}}}^{\infty} \beta_d P_2(d) \\ &< \sum_{d=d_{\text{free}}}^{\infty} \beta_d Q(\sqrt{2\gamma_b R_c d}) \end{aligned} \tag{8-2-26}$$

If the Q function is upper-bounded by an exponential as indicated in (8-2-22) then (8-2-26) can be expressed in the simple form

$$P_b < \sum_{d=d_{\text{free}}}^{\infty} \beta_d D^d \Big|_{D=e^{-\gamma_b R_c}} < \frac{dT(D, N)}{dN}\Big|_{N=1, D=e^{-\gamma_b R_c}} \tag{8-2-27}$$

If $k > 1$, the equivalent bit error probability is obtained by dividing (8-2-26) and (8-2-27) by k.

The expressions for the probability of error given above are based on the assumption that the code bits are transmitted by binary coherent PSK. The results also hold for four-phase coherent PSK, since this modulation/demodulation technique is equivalent to two independent (phase-quadrature) binary PSK systems. Other modulation and demodulation techniques, such as coherent and noncoherent binary FSK, can be accommodated by recomputing the pairwise error probability $P_2(d)$. That is, a change in the modulation and demodulation technique used to transmit the coded information sequence affects only the computation of $P_2(d)$. Otherwise, the derivation for P_b remains the same.

Although the above derivation of the error probability for Viterbi decoding of a convolutional code applies to binary convolutional codes, it is relatively easy to generalize it to nonbinary convolutional codes in which each nonbinary symbol is mapped into a distinct waveform. In particular, the coefficients $\{\beta_d\}$ in the expansion of the derivative of $T(D, N)$, given in (8-2-25), represent the number of symbol errors in two paths separated in distance (measured in terms of symbols) by d symbols. Again, we denote the probability of error in a pairwise comparison of two paths that are separated in distance by d as $P_2(d)$. Then the symbol error probability, for a k-bit symbol, is upper-bounded by

$$P_M \leq \sum_{d=d_{\text{free}}}^{\infty} \beta_d P_2(d)$$

The symbol error probability can be converted into an equivalent bit error probability. For example, if 2^k orthogonal waveforms are used to transmit the k-bit symbols, the equivalent bit error probability is P_M multiplied by a factor $2^{k-1}/(2^k - 1)$, as shown in Chapter 5.

8-2-4 Probability of Error for Hard-Decision Decoding

We now consider the performance achieved by the Viterbi decoding algorithm on a binary symmetric channel. For hard-decision decoding of the convolutional code, the metrics in the Viterbi algorithm are the Hamming distances between the received sequence and the $2^{k(K-1)}$ surviving sequences at each node of the trellis.

As in our treatment of soft-decision decoding, we begin by determining the

first-event error probability. The all-zero path is assumed to be transmitted. Suppose that the path being compared with the all-zero path at some node B has distance d from the all-zero path. If d is odd, the all-zero path will be correctly selected if the number of errors in the received sequence is less than $\frac{1}{2}(d+1)$; otherwise, the incorrect path will be selected. Consequently, the probability of selecting the incorrect path is

$$P_2(d) = \sum_{k=(d+1)/2}^{d} \binom{d}{k} p^k (1-p)^{d-k} \tag{8-2-28}$$

where p is the probability of a bit error for the binary symmetric channel. If d is even, the incorrect path is selected when the number of errors exceeds $\frac{1}{2}d$. If the number of errors equals $\frac{1}{2}d$, there is a tie between the metrics in the two paths, which may be resolved by randomly selecting one of the paths; thus, an error occurs half the time. Consequently, the probability of selecting the incorrect path is

$$P_2(d) = \sum_{k=d/2+1}^{d} \binom{d}{k} p^k (1-p)^{d-k} + \frac{1}{2}\binom{d}{\frac{1}{2}d} p^{d/2}(1-p)^{d/2} \tag{8-2-29}$$

As indicated in Section 8-2-3, there are many possible paths with different distances that merge with the all-zero path at a given node. Therefore, there is no simple exact expression for the first-event error probability. However, we can overbound this error probability by the sum of the pairwise error probabilities $P_2(d)$ over all possible paths that merge with the all-zero path at the given node. Thus, we obtain the union bound

$$P_e < \sum_{d=d_{\text{free}}}^{\infty} a_d P_2(d) \tag{8-2-30}$$

where the coefficients $\{a_d\}$ represent the number of paths corresponding to the set of distances $\{d\}$. These coefficients are the coefficients in the expansion of the transfer function $T(D)$ or $T(D, N)$.

Instead of using the expressions for $P_2(d)$ given in (8-2-28) and (8-2-29), we can use the upper bound

$$P_2(d) < [4p(1-p)]^{d/2} \tag{8-2-31}$$

which was given in Section 8-1-5. Use of this bound in (8-2-30) yields a looser upper bound on the first-event error probability, in the form

$$\begin{aligned} P_e &< \sum_{d=d_{\text{free}}}^{\infty} a_d [4p(1-p)]^{d/2} \\ &< T(D)\big|_{D=\sqrt{4p(1-p)}} \end{aligned} \tag{8-2-32}$$

Let us now determine the probability of a bit error. As in the case of soft-decision decoding, we make use of the fact that the exponents in the factors of N that appear in the transfer function $T(D, N)$ indicate the number of nonzero information bits that are in error when an incorrect path is selected over the all-zero path. By differentiating $T(D, N)$ with respect to N and setting $N = 1$, the exponents of N become multiplication factors of the corresponding error-event probabilities $P_2(d)$. Thus, we obtain the expression for the upper bound on the bit error probability, in the form

$$P_b < \sum_{d=d_{\text{free}}}^{\infty} \beta_d P_2(d) \tag{8-2-33}$$

where the $\{\beta_d\}$ are the coefficients in the expansion of the derivative of $T(D, N)$, evaluated at $N = 1$. For $P_2(d)$, we may use either the expressions given in (8-2-28) and (8-2-29) or the upper bound in (8-2-31). If the latter is used, the upper bound on P_b can be expressed as

$$P_b < \left.\frac{dT(D, N)}{dN}\right|_{N=1, D=\sqrt{4p(1-p)}} \tag{8-2-34}$$

When $k > 1$, the results given in (8-2-33) and (8-2-34) for P_b should be divided by k.

A comparison of the error probability for the rate 1/3, $K = 3$ convolutional code with soft-decision decoding and hard-decision decoding is made in Fig. 8-2-15. Note that the Chernoff upper bound given by (8-2-34) is less than 1 dB above the tighter upper bound given by (8-2-33) in conjunction with (8-2-28) and (8-2-29). The advantage of the Chernoff bound is its computational

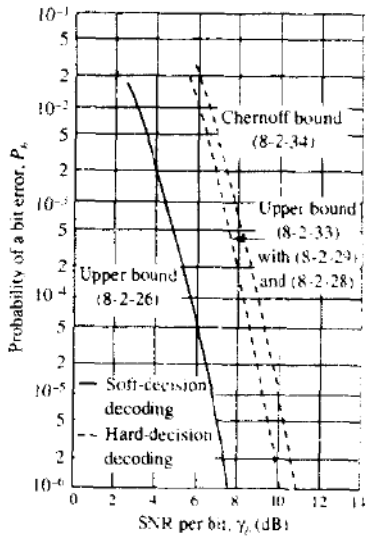


FIGURE 8-2-15 Comparison of soft-decision and hard-decision decoding for $K = 3$, $k = 1$, $n = 3$ convolutional code

simplicity. In comparing the performance between soft-decision and hard-decision decoding, note that the difference obtained from the upper bounds is approximately 2.5 dB for $10^{-6} \leq P_b \leq 10^{-2}$.

Finally, we should mention that the ensemble average error rate performance of a convolutional code on a discrete memoryless channel, just as in the case of a block code, can be expressed in terms of the cutoff rate parameter R_0 as (for the derivation, see Viterbi and Omura, 1979).

$$\bar{P}_b < \frac{(q-1)q^{-KR_c/R_c}}{[1-q^{-(R_0-R_c)/R_c}]^2}, \qquad R_c \leq R_0$$

where q is the number of channel input symbols, K is the constraint length of the code, R_c is the code rate, and R_0 is the cutoff rate defined in Sections 7-2 and 8-1. Therefore, conclusions reached by computing R_0 for various channel conditions apply to both block codes and convolutional codes.

8-2-5 Distance Properties of Binary Convolutional Codes

In this subsection, we shall tabulate the minimum free distance and the generators for several binary, short-constraint-length convolutional codes for several code rates. These binary codes are optimal in the sense that, for a given rate and a given constraint length, they have the largest possible d_{free}. The generators and the corresponding values of d_{free} tabulated below have been obtained by Odenwalder (1970), Larsen (1973), Paaske (1974), and Daut *et al.* (1982) using computer search methods.

Heller (1968) has derived a relatively simple upper bound on the minimum free distance of a rate $1/n$ convolutional code. It is given by

$$d_{\text{free}} \leq \min_{l \geq 1} \left\lfloor \frac{2^{l-1}}{2^l - 1} (K + l - 1)n \right\rfloor \tag{8-2-35}$$

where $\lfloor x \rfloor$ denotes the largest integer contained in x. For purposes of comparison, this upper bound is also given in the tables for the rate $1/n$ codes. For rate k/n convolutional codes, Daut *et al.* (1982) has given a modification to Heller's bound. The values obtained from this upper bound for k/n codes are also tabulated.

Tables 8-2-1 to 8-2-7 list the parameter of rate $1/n$ convolutional codes for $n = 2, 3, \ldots, 8$. Tables 8-2-8 to 8-2-11 list the parameters of several rate k/n convolutional codes for $k \leq 4$ and $n \leq 8$.

8-2-6 Nonbinary Dual-k Codes and Concatenated Codes

Our treatment of convolutional codes thus far has been focused primarily on binary codes. Binary codes are particularly suitable for channels in which binary or quaternary PSK modulation and coherent demodulation is possible.

TABLE 8-2-1 RATE 1/2 MAXIMUM FREE DISTANCE CODE

Constraint length K	Generators in octal		d_{free}	Upper bound on d_{free}
3	5	7	5	5
4	15	17	6	6
5	23	35	7	8
6	53	75	8	8
7	133	171	10	10
8	247	371	10	11
9	561	753	12	12
10	1,167	1,545	12	13
11	2,335	3,661	14	14
12	4,335	5,723	15	15
13	10,533	17,661	16	16
14	21,675	27,123	16	17

Source: Odenwalder (1970) and Larsen (1973).

However, there are many applications in which PSK modulation and coherent demodulation is not suitable or possible. In such cases, other modulation techniques, e.g., M-ary FSK, are employed in conjunction with noncoherent demodulation. Nonbinary codes are particularly matched to M-ary signals that are demodulated noncoherently.

In this subsection, we describe a class of nonbinary convolutional codes, called *dual-k codes*, that are easily decoded by means of the Viterbi algorithm using either soft-decision or hard-decision decoding. They are also suitable either as an outer code or as an inner code in a concatenated code, as will also be described below.

TABLE 8-2-2 RATE 1/3 MAXIMUM FREE DISTANCE CODES

Constraint length K	Generators in octal			d_{free}	Upper bound on d_{free}
3	5	7	7	8	8
4	13	15	17	10	10
5	25	33	37	12	12
6	47	53	75	13	13
7	133	145	175	15	15
8	225	331	367	16	16
9	557	663	711	18	18
10	1,117	1,365	1,633	20	20
11	2,353	2,671	3,175	22	22
12	4,767	5,723	6,265	24	24
13	10,533	10,675	17,661	24	24
14	21,645	35,661	37,133	26	26

Sources: Odenwalder (1970) and Larsen (1973).

TABLE 8-2-3 RATE 1/4 MAXIMUM FREE DISTANCE CODES

Constraint length K	Generators in octal				d_{free}	Upper bound on d_{free}
3	5	7	7	7	10	10
4	13	15	15	17	13	15
5	25	27	33	37	16	16
6	53	67	71	75	18	18
7	135	135	147	163	20	20
8	235	275	313	357	22	22
9	463	535	733	745	24	24
10	1,117	1,365	1,633	1,653	27	27
11	2,387	2,353	2,671	3,175	29	29
12	4,767	5,723	6,265	7,455	32	32
13	11,145	12,477	15,537	16,727	33	33
14	21,113	23,175	35,527	35,537	36	36

Source: Larsen (1973).

TABLE 8-2-4 RATE 1/5 MAXIMUM FREE DISTANCE CODES

Constraint length K	Generators in octal					d_{free}	Upper bound on d_{free}
3	7	7	7	5	5	13	13
4	17	17	13	15	15	16	16
5	37	27	33	25	35	20	20
6	75	71	73	65	57	22	22
7	175	131	135	135	147	25	25
8	257	233	323	271	357	28	28

Source: Daut *et al.* (1982).

TABLE 8-2-5 RATE 1/6 MAXIMUM FREE DISTANCE CODES

Constraint length K	Generators in octal			d_{free}	Upper bound on d_{free}
3	7	7	7	16	16
	7	5	5		
4	17	17	13	20	20
	13	15	15		
5	37	35	27	24	24
	33	25	35		
6	73	75	55	27	27
	65	47	57		
7	173	151	135	30	30
	135	163	137		
8	253	375	331	34	34
	235	313	357		

Source: Daut *et al.* (1982).

TABLE 8-2-6 RATE 1/7 MAXIMUM FREE DISTANCE CODES

Constraint length K	Generators in octal				d_{free}	Upper bound on d_{free}
3	7	7	7	7	18	18
	5	5	5			
4	17	17	13	13	23	23
	13	15	15			
5	35	27	25	27	28	28
	33	35	37			
6	53	75	65	75	32	32
	47	67	57			
7	165	145	173	135	36	36
	135	147	137			
8	275	253	375	331	40	40
	235	313	357			

Source: Daut *et al.* (1982).

TABLE 8-2-7 RATE 1/8 MAXIMUM FREE DISTANCE CODES

Constraint length K	Generators in octal				d_{free}	Upper bound on d_{free}
3	7	7	5	5	21	21
	5	7	7	7		
4	17	17	13	13	26	26
	13	15	15	17		
5	37	33	25	25	32	32
	35	33	27	37		
6	57	73	51	65	36	36
	75	47	67	57		
7	153	111	165	173	40	40
	135	135	147	137		
8	275	275	253	371	45	45
	331	235	313	357		

Source: Daut *et al.* (1982).

TABLE 8-2-8 RATE 2/3 MAXIMUM FREE DISTANCE CODES

Constraint length K	Generators in octal			d_{free}	Upper bound on d_{free}
2	17	06	15	3	4
3	27	75	72	5	6
4	236	155	337	7	7

Source: Duat *et al.* (1982).

TABLE 8-2-9 RATE $k/5$ MAXIMUM FREE DISTANCE CODES

Rate	Constraint length K	Generators in octal					d_{free}	Upper bound on d_{free}
2/5	2	17	07	11	12	04	6	6
	3	27	71	52	65	57	10	10
	4	247	366	171	266	373	12	12
3/5	2	35	23	75	61	47	5	5
4/5	2	237	274	156	255	337	3	4

Source: Daut *et al.* (1982).

TABLE 8-2-10 RATE $k/7$ MAXIMUM FREE DISTANCE CODES

Rate	Constraint length K	Generators in octal					d_{free}	Upper bound on d_{free}
2/7	2	05	06	12	15		9	9
		15	13	17				
	3	33	55	72	47		14	14
		25	53	75				
	4	312	125	247	366		18	18
		171	266	373				
3/7	2	45	21	36	62		8	8
		57	43	71				
4/7	2	130	067	237	274		6	7
		156	255	337				

Source: Daut *et al.* (1982).

TABLE 8-2-11 RATES 3/4 AND 3/8 MAXIMUM FREE DISTANCE CODES

Rate	Constraint length K	Generators in octal				d_{free}	Upper bound on d_{free}
3/4	2	13	25	61	47	4	4
3/8	2	15	42	23	61	8	8
		51	36	75	47		

Source: Daut *et al.* (1982).

A dual-k rate 1/2 convolutional encoder may be represented as shown in Fig. 8-2-16. It consists of two ($K = 2$) k-bit shift-register stages and $n = 2k$ function generators. Its output is two k-bit symbols. We note that the code considered in Example 8-2-3 is a dual-2 convolutional code.

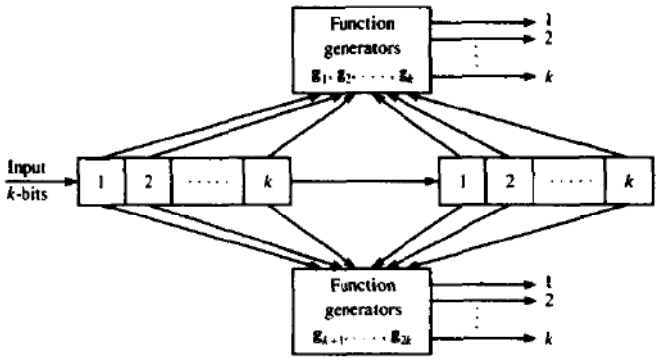


FIGURE 8-2-16 Encoder for rate 1/2 dual-k codes.

The $2k$ function generators for the dual-k codes have been given by Viterbi and Jacobs (1975). These may be expressed in the form

$$\begin{bmatrix} \leftarrow \mathbf{g}_1 \rightarrow \\ \leftarrow \mathbf{g}_2 \rightarrow \\ \vdots \\ \leftarrow \mathbf{g}_k \rightarrow \end{bmatrix} = \left[\begin{array}{ccccccc:ccc} 1 & 0 & 0 & \cdots & 0 & 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 & 0 & 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & & & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 0 & 0 & \cdots & 0 & 1 \end{array}\right] = [\mathbf{I}_k \quad \mathbf{I}_k]$$

$$\begin{bmatrix} \leftarrow \mathbf{g}_{k+1} \rightarrow \\ \leftarrow \mathbf{g}_{k+2} \rightarrow \\ \vdots \\ \leftarrow \mathbf{g}_{2k} \rightarrow \end{bmatrix} = \left[\begin{array}{cccccccc:cccccc} 1 & 1 & 0 & 0 & \cdots & & 0 & 1 & 0 & 0 & \cdots & & 0 \\ 0 & 0 & 1 & 0 & \cdots & & 0 & 0 & 1 & 0 & \cdots & & 0 \\ 0 & 0 & 0 & 1 & 0 & \cdots & 0 & 0 & 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 0 & & \cdots & 0 & 1 & & & & & & \\ 1 & 0 & 0 & & \cdots & 0 & 0 & 0 & 0 & \cdots & & 0 & 1 \end{array}\right]$$

$$= \left[\begin{array}{cccccccc:c} 1 & 1 & 0 & 0 & \cdots & & 0 & \\ 0 & 0 & 1 & 0 & \cdots & & 0 & \\ 0 & 0 & 0 & 1 & 0 & \cdots & 0 & \\ \hdashline 0 & 0 & 0 & & \cdots & 0 & 1 & \\ 1 & 0 & 0 & & \cdots & 0 & 0 & \end{array}\right] \begin{array}{c} \\ \\ \\ \\ \end{array} \left[\begin{array}{c} \\ \\ \ddots \\ \\ \end{array}\right] \begin{array}{c} \\ \\ \mathbf{I}_k \\ \\ \end{array}$$

(8-2-36)

where $\mathbf{I}_k$ denotes the $k \times k$ identity matrix.

The general form for the transfer function of a rate 1/2 dual-k code has been derived by Odenwalder (1976). It is expressed as

$$T(D, N, J) = \frac{(2^k - 1)D^4J^2N}{1 - NJ[2D + (2^k - 3)D^2]}$$

$$= \sum_{i=4}^{\infty} a_i D^i N^{f(i)} J^{h(i)} \tag{8-2-37}$$

where D represents the Hamming distance for the q-ary ($q = 2^k$) symbols, the $f(i)$ exponent on N represents the number of information symbol errors that are produced in selecting a branch in the tree or trellis other than a corresponding branch on the all-zero path, and the $h(i)$ exponent on J is equal to the number of branches in a given path. Note that the minimum free distance is $d_{\text{free}} = 4$ symbols ($4k$ bits).

Lower-rate dual-k convolutional codes can be generated in a number of ways, the simplest of which is to repeat each symbol generated by the rate 1/2 code r times, where $r = 1, 2, \ldots, m$ ($r = 1$ corresponds to each symbol appearing once). If each symbol in any particular branch of the tree or trellis or state diagram is repeated r times, the effect is to increase the distance parameter from D to D^r. Consequently the transfer function for a rate $1/2r$ dual-k code is

$$T(D, N, J) = \frac{(2^k - 1)D^{4r}J^2N}{1 - NJ[2D^r + (2^k - 3)D^{2r}]} \tag{8-2-38}$$

In the transmission of long information sequences, the path length parameter J in the transfer function may be suppressed by setting $J = 1$. The resulting transfer function $T(D, N)$ may be differentiated with respect to N, and N is set to unity. This yields

$$\left.\frac{dT(D, N)}{dN}\right|_{N=1} = \frac{(2^k - 1)D^{4r}}{[1 - 2D^r - (2^k - 3)D^{2r}]^2} = \sum_{i=4r}^{\infty} \beta_i D^i \tag{8-2-39}$$

where β_i represents the number of symbol errors associated with a path having distance D^i from the all-zero path, as described previously in Section 8-2-3. The expression in (8-2-39) may be used to evaluate the error probability for dual-k codes under various channel conditions.

Performance of Dual-k Codes with M-ary Modulation Suppose that a dual-k code is used in conjunction with M-ary orthogonal signaling at the modulator, where $M = 2^k$. Each symbol from the encoder is mapped into one of the M possible orthogonal waveforms. The channel is assumed to add white gaussian noise. The demodulator consists of M matched filters.

If the decoder performs hard-decision decoding, the performance of the code is determined by the symbol error probability P_M. This error probability has been computed in Chapter 5 for both coherent and noncoherent detection. From P_M, we can determine $P_2(d)$ according to (8-2-28) or (8-2-29), which is the probability of error in a pairwise comparison of the all-zero path with a path that differs in d symbols. The probability of a bit error is upper-bounded as

$$P_b < \frac{2^{k-1}}{2^k - 1} \sum_{d=4r}^{\infty} \beta_d P_2(d) \tag{8-2-40}$$

The factor $2^{k-1}/(2^k - 1)$ is used to convert the symbol error probability to the bit error probability.

Instead of hard-decision decoding, suppose that the decoder performs soft-decision decoding using the output of a demodulator that employs a square-law detector. The expression for the bit error probability given by (8-2-40) still applies, but now $P_2(d)$ is given by (see Section 12-1-1)

$$P_2(d) = \frac{1}{2^{2d-1}} \exp\left(-\tfrac{1}{2}\gamma_b R_c d\right) \sum_{i=0}^{d-1} K_i\left(\tfrac{1}{2}\gamma_b R_c d\right) \tag{8-2-41}$$

where

$$K_i = \frac{1}{i!} \sum_{l=0}^{d-1-i} \binom{2d-1}{l}$$

and $R_c = 1/2r$ is the code rate. This expression follows from the result (8-1-63).

Concatenated Codes In Section 8-1-8, we considered the concatenation of two block codes to form a long block code. Now that we have described convolutional codes, we broaden our viewpoint and consider the concatenation of a block code with a convolutional code or the concatenation of two convolutional codes.

As described previously, the outer code is usually chosen to be nonbinary, with each symbol selected from an alphabet of $q = 2^k$ symbols. This code may be a block code, such as a Reed–Solomon code, or a convolutional code, such as a dual-k code. The inner code may be either binary or nonbinary, and either a block or a convolutional code. For example, a Reed–Solomon code may be selected as the outer code and a dual-k code may be selected as the inner code. In such a concatenation scheme, the number of symbols in the outer (Reed–Solomon) code q equals 2^k, so that each symbol of the outer code maps into a k-bit symbol of the inner dual-k code. M-ary orthogonal signals may be used to transmit the symbols.

The decoding of such concatenated codes may also take a variety of different forms. If the inner code is a convolutional code having a short constraint length, the Viterbi algorithm provides an efficient means for decoding, using either soft-decision or hard-decision decoding.

If the inner code is a block code, and the decoder for this code performs soft-decision decoding, the outer decoder may also perform soft-decision decoding using as inputs the metrics corresponding to each word of the inner code. On the other hand, the inner decoder may make a hard decision after receipt of the code word and feed the hard decisions to the outer decoder. Then the outer decoder must perform hard-decision decoding.

The following example describes a concatenation code in which the outer code is a convolutional code and the inner code is a block code.

Example 8-2-5

Suppose we construct a concatenated code by selecting a dual-k code as the outer code and a Hadamard code as the inner code. To be specific, we select a rate 1/2 dual-5 code and a Hadamard (16, 5) inner code. The dual-5 rate

1/2 code has a minimum free distance $D_{free} = 4$ and the Hadamard code has a minimum distance $d_{min} = 8$. Hence, the concatenated code has an effective minimum distance of 32. Since there are 32 code words in the Hadamard code and 32 possible symbols in the outer code, in effect, each symbol from the outer code is mapped into one of the 32 Hadamard code words.

The probability of a symbol error in decoding the inner code may be determined from the results of the performance of block codes given in Sections 8-1-4 and 8-1-5 for soft-decision and hard-decision decoding, respectively. First, suppose that hard-decision decoding is performed in the inner decoder with the probability of a code word (symbol of outer code) error denoted as P_{32}, since $M = 32$. Then the performance of the outer code and, hence, the performance of the concatenated code is obtained by using this error probability in conjunction with the transfer function for the dual-5 code given by (8-2-37).

On the other hand, if soft-decision decoding is used on both the outer and the inner codes, the soft-decision metric from each received Hadamard code word is passed to the Viterbi algorithm, which computes the accumulated metrics for the competing paths through the trellis. We shall give numerical results on the performance of concatenated codes of this type in our discussion of coding for Rayleigh fading channels.

8-2-7 Other Decoding Algorithms for Convolutional Codes

The Viterbi algorithm described in Section 8-2-2 is the optimum decoding algorithm (in the sense of maximum-likelihood decoding of the entire sequence) for convolutional codes. However, it requires the computation of 2^{kK} metrics at each node of the trellis and the storage of $2^{k(K-1)}$ metrics and $2^{k(K-1)}$ surviving sequences, each of which may be about $5kK$ bits long. The computational burden and the storage required to implement the Viterbi algorithm make it impractical for convolutional codes with large constraint length.

Prior to the discovery of the optimum algorithm by Viterbi, a number of other algorithms had been proposed for decoding convolutional codes. The earliest was the sequential decoding algorithm originally proposed by Wozencraft (1957, 1961), and subsequently modified by Fano (1963).

The Fano sequential decoding algorithm searches for the most probable path through the tree or trellis by examining one path at a time. The increment added to the metric along each branch is proportional to the probability of the received signal for that branch, just as in Viterbi decoding, with the exception that an additional negative constant is added to each branch metric. The value of this constant is selected such that the metric for the correct path will increase on the average, while the metric for any incorrect path will decrease on the average. By comparing the metric of a candidate path with a moving (increasing) threshold, Fano's algorithm detects and discards incorrect paths.

To be more specific, let us consider a memoryless channel. The metric for

the ith path through the tree or trellis from the first branch to branch B may be expressed as

$$CM^{(i)} = \sum_{j=1}^{B} \sum_{m=1}^{n} \mu_{jm}^{(i)} \tag{8-2-42}$$

where

$$\mu_{jm}^{(i)} = \log_2 \frac{p(r_{jm} \mid c_{jm}^{(i)})}{p(r_{jm})} - \mathcal{K} \tag{8-2-43}$$

In (8-2-43), r_{jm} is the demodulator output sequence, $p(r_{jm} \mid c_{jm}^{(i)})$ denotes the pdf of r_{jm} conditional on the code bit $c_{jm}^{(i)}$ for the mth bit of the jth branch of the ith path, and $\mathcal{K}$ is a positive constant. $\mathcal{K}$ is selected as indicated above so that the incorrect paths will have a decreasing metric while the correct path will have an increasing metric on the average. Note that the term $p(r_{jm})$ in the denominator is independent of the code sequence, and, hence, may be subsumed in the constant factor.

The metric given by (8-2-43) is generally applicable for either hard- or soft-decision decoding. However, it can be considerably simplified when hard-decision decoding is employed. Specifically, if we have a BSC with transition (error) probability p, the metric for each received bit, consistent with the form in (8-2-43) is given by

$$\mu_{jm}^{(i)} = \begin{cases} \log_2 [2(1-p)] - R_c & \text{if } \tilde{r}_{jm} = c_{jm}^{(i)} \\ \log_2 2p - R_c & \text{if } \tilde{r}_{jm} \neq c_{jm}^{(i)} \end{cases} \tag{8-2-44}$$

where $\tilde{r}_{jm}$ is the hard-decision output from the demodulator and $c_{jm}^{(i)}$ is the mth code bit in the jth branch of the ith path in the tree and R_c is the code rate. Note that this metric requires some (approximate) knowledge of the error probability.

Example 8-2-6

Suppose we have a rate $R_c = 1/3$ binary convolutional code for transmitting information over a BSC with $p = 0.1$. By evaluating (8-2-44) we find that

$$\mu_{jm}^{(i)} = \begin{cases} 0.52 & \text{if } \tilde{r}_{jm} = c_{jm}^{(i)} \\ -2.65 & \text{if } \tilde{r}_{jm} \neq c_{jm}^{(i)} \end{cases} \tag{8-2-45}$$

To simplify the computations, the metric in (8-2-45) may be normalized. It is well approximated as

$$\mu_{jm}^{(i)} = \begin{cases} 1 & \text{if } \tilde{r}_{jm} = c_{jm}^{(i)} \\ -5 & \text{if } \tilde{r}_{jm} \neq c_{jm}^{(i)} \end{cases} \tag{8-2-46}$$

Since the code rate is 1/3, there are three output bits from the encoder for each input bit. Hence, the branch metric consistent with (8-2-46) is

$$\mu_j^{(i)} = 3 - 6d$$

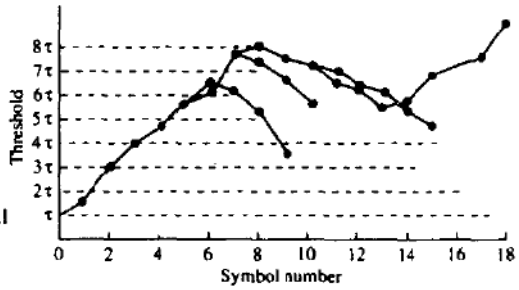


FIGURE 8-2-17 An example of the path search in sequential decoding. [*From Jordan (1966), © 1966 IEEE.*]

or, equivalently,

$$\mu_j^{(i)} = 1 - 2d \tag{8-2-47}$$

where d is the Hamming distance of the three received bits from the three branch bits. Thus, the metric $\mu_j^{(i)}$ is simply related to the Hamming distance of the received bits to the code bits in the jth branch of the ith path.

Initially, the decoder may be forced to start on the correct path by the transmission of a few known bits of data. Then it proceeds forward from node to node, taking the most probable (largest metric) branch at each node and increasing the threshold such that the threshold is never more than some preselected value, say τ, below the metric. Now suppose that the additive noise (for soft-decision decoding) or demodulation errors resulting from noise on the channel (for hard-decision decoding) cause the decoder to take an incorrect path because it appears more probable than the correct path. This is illustrated in Fig. 8-2-17. Since the metrics of an incorrect path decrease on the average, the metric will fall below the current threshold, say τ_0. When this occurs, the decoder backs up and takes alternative paths through the tree or trellis, in order of decreasing branch metrics, in an attempt to find another path that exceeds the threshold τ_0. If it is successful in finding an alternative path, it continues along that path, always selecting the most probable branch at each node. On the other hand, if no path exists that exceeds the threshold τ_0, the threshold is reduced by an amount τ and the original path is retraced. If the original path does not stay above the new threshold, the decoder resumes its backward search for other paths. This procedure is repeated, with the threshold reduced by τ for each repetition, until the decoder finds a path that remains above the adjusted threshold. A simplified flow diagram of Fano's algorithm is shown in Fig. 8-2-18.

The sequential decoding algorithm requires a buffer memory in the decoder to store incoming demodulated data during periods when the decoder is searching for alternate paths. When a search terminates, the decoder must be capable of processing demodulated bits sufficiently fast to empty the buffer prior to commencing a new search. Occasionally, during extremely long searches, the buffer may overflow. This causes loss of data, a condition that

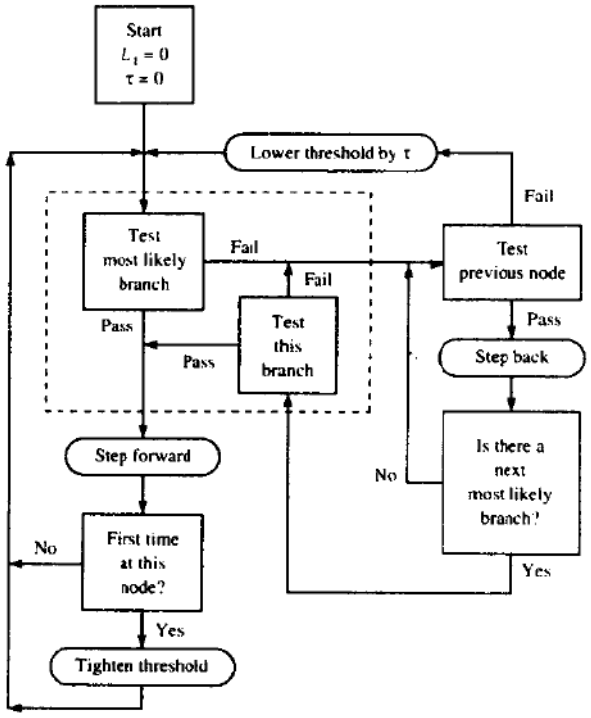
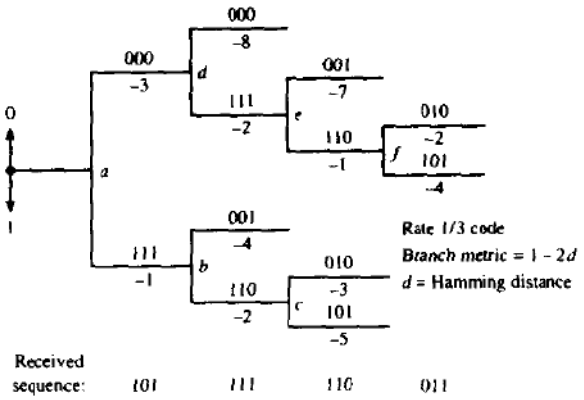


FIGURE 8-2-18 A simplified flow diagram of Fano's algorithm. [*From Jordan (1966), © 1966 IEEE.*]

can be remedied by retransmission of the lost information. In this regard, we should mention that the cutoff rate R_0 has special meaning in sequential decoding. It is the rate above which the average number of decoding operations per decoded digit becomes infinite, and it is termed the *computational cutoff rate* R_{comp}. In practice, sequential decoders usually operate at rates near R_0.

The Fano sequential decoding algorithm has been successfully implemented in several communication systems. Its error rate performance is comparable to that of Viterbi decoding. However, in comparison with Viterbi decoding, sequential decoding has a significantly larger decoding delay. On the positive side, sequential decoding requires less storage than Viterbi decoding and, hence, it appears attractive for convolutional codes with a large constraint length. The issues of computational complexity and storage requirements for sequential decoding are interesting and have been thoroughly investigated. For an analysis of these topics and other characteristics of the Fano algorithm, the interested reader may refer to Gallager (1968), Wozencraft and Jacobs (1965), Savage (1966), and Forney (1974).

Another type of sequential decoding algorithm, called a *stack algorithm*, has been proposed independently by Jelinek (1969) and Zigangirov (1966). In contrast to the Viterbi algorithm, which keeps track of $2^{(K-1)k}$ paths and



Stack with accumulated path metrics

Step *a*	Step *b*	Step *c*	Step *d*	Step *e*	Step *f*
-1	-2	-3	-2	-1	-2
-3	-3	-3	-3	-3	-3
	-4	-4	-4	-4	-4
		-5	-5	-5	-4
			-8	-7	-5
				-8	-7
					-8

FIGURE 8-2-19 A example of the stack algorithm for decoding a rate 1/3 convolutional code.

corresponding metrics, the stack sequential decoding algorithm deals with fewer paths and their corresponding metrics. In a stack algorithm, the more probable paths are ordered according to their metrics, with the path at the top of the stack having the largest metric. At each step of the algorithm, only the path at the top of the stack is extended by one branch. This yields 2^k successors and their corresponding metrics. These 2^k successors along with the other paths are then reordered according to the values of the metrics and all paths with metrics that fall below some preselected amount from the metric of the top path may be discarded. Then the process of extending the path with the largest metric is repeated. Figure 8-2-19 illustrates the first few steps in a stack algorithm.

It is apparent that when none of the 2^k extensions of the path with the largest metric remains at the top of the stack, the next step in the search involves the extension of another path that has climbed to the top of the stack. It follows that the algorithm does not necessarily advance by one branch through the trellis in every iteration. Consequently, some amount of storage must be provided for newly received signals and previously received signals in order to allow the algorithm to extend the search along one of the shorter paths, when such a path reaches the top of the stack.

In a comparison of the stack algorithm with the Viterbi algorithm, the stack algorithm requires fewer metric computations, but this computational saving is offset to a large extent by the computations involved in reordering the stack after every iteration. In comparison with the Fano algorithm, the stack algorithm is computationally simpler, since there is no retracing over the same path as is done in the Fano algorithm. On the other hand, the stack algorithm requires more storage than the Fano algorithm.

A third alternative to the optimum Viterbi decoder is a method called *feedback decoding* (Heller, 1975), which has been applied to decoding for a BSC (hard-decision decoding). In feedback decoding, the decoder makes a hard decision on the information bit at stage j based on metrics computed from stage j to stage $j+m$, where m is a preselected positive integer. Thus, the decision on the information bit is either 0 or 1 depending on whether the minimum Hamming distance path that begins at stage j and ends at stage $j+m$ contains a 0 or 1 in the branch emanating from stage j. Once a decision is made on the information bit at stage j, only that part of the tree that stems from the bit selected at stage j is kept (half the paths emanating from node j) and the remaining part is discarded. This is the feedback feature of the decoder.

The next step is to extend the part of the tree that has survived to stage $j+1+m$ and consider the paths from stage $j+1$ to $j+1+m$ in deciding on the bit at stage $j+1$. Thus, this procedure is repeated at every stage. The parameter m is simply the number of stages in the tree that the decoder looks ahead before making a hard decision. Since a large value of m results in a large amount of storage, it is desirable to select m as small as possible. On the other hand, m must be sufficiently large to avoid a severe degradation in performance. To balance these two conflicting requirements, m is usually selected in the range $K \leq m \leq 2K$, where K is the constraint length. Note that this decoding delay is significantly smaller than the decoding delay in a Viterbi decoder, which is usually about $5K$.

Example 8-2-7

Let us consider the use of a feedback decoder for the rate 1/3 convolutional code shown in Fig. 8-2-2. Figure 8-2-20 illustrates the tree diagram and the operation of the feedback decoder for $m = 2$. That is, in decoding the bit at branch j, the decoder considers the paths at branches j, $j+1$, and $j+2$. Beginning with the first branch, the decoder computes eight metrics (Hamming distances), and decides that the bit for the first branch is 0 if the minimum distance path is contained in the upper part of the tree, and 1 if the minimum distance path is contained in the lower part of the tree. In this example, the received sequence for the first three branches is assumed to be 101111110, so that the minimum distance path is in the upper part of the tree. Hence, the first output bit is 0.

The next step is to extend the upper part of the tree (the part of the tree that has survived) by one branch, and to compute the eight metrics for

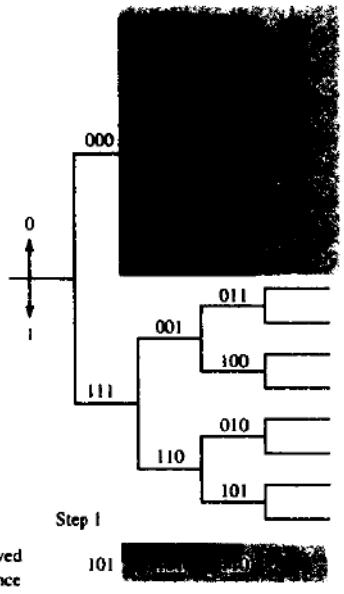


FIGURE 8-2-20 An example of feedback decoding for a rate 1/3 convolutional code.

Step 1 : Upper-tree metrics : 7,6,5,2* ; lower-tree metrics : 5,4,3,4 → 0
Step 2 : Upper-tree metrics : 7,6,5,6 ; lower-tree metrics : 3,6,1* ,2 → 1

branches 2, 3, and 4. For the assumed received sequence 111110011, the minimum-distance path is contained in the lower part of the section of the tree that survived from the first step. Hence, the second output bit is 1. The third step is to extend this lower part of the tree and to repeat the procedure described for the first two steps.

Instead of computing metrics as described above, a feedback decoder for the BSC may be efficiently implemented by computing the syndrome from the received sequence and using a table lookup method for correcting errors. This method is similar to the one described for decoding block codes. For some convolutional codes, the feedback decoder simplifies to a form called a *majority logic decoder* or a *threshold decoder* (Massey, 1963; Heller, 1975).

8-2-8 Practical Considerations in the Application of Convolutional Codes

Convolutional codes are widely used in many practical applications of communications system design. Viterbi decoding is predominantly used for short constraint lengths ($K \leq 10$), while sequential decoding is used for long constraint length codes, where the complexity of Viterbi decoding becomes prohibitive. The choice of constraint length is dictated by the desired coding gain.

From the error probability results for soft-decision decoding given by

TABLE 8-2-12 UPPER BOUNDS ON CODING GAIN FOR SOFT-DECISION DECODING OF SOME CONVOLUTION CODES

Rate 1/2 codes			Rate 1/3 codes		
Constraint length K	d_{free}	Upper bound (db)	Constraint length K	d_{free}	Upper bound (dB)
3	5	3.98	3	8	4.26
4	6	4.77	4	10	5.23
5	7	5.44	5	12	6.02
6	8	6.02	6	13	6.37
7	10	6.99	7	15	6.99
8	10	6.99	8	16	7.27
9	12	7.78	9	18	7.78
10	12	7.78	10	20	8.24

(8-2-26) it is apparent that the coding gain achieved by a convolutional code over an uncoded binary PSK or QPSK system is

$$\text{coding gain} \leq 10 \log_{10} (R_c d_{free})$$

We also know that the minimum free distance d_{free} can be increased either by decreasing the code rate or by increasing the constraint length, or both. Table 8-2-12 provides a list of upper bounds on the coding gain for several convolutional codes. For purposes of comparison, Table 8-2-13 lists the actual coding gains and the upper bounds for several short constraint length convolutional codes with Viterbi decoding. It should be noted that the coding gain increases toward the asymptotic limit as the SNR per bit increases.

These results are based on soft-decision Viterbi decoding. If hard-decision decoding is used, the coding gains are reduced by approximately 2 dB for the AWGN channel.

Larger coding gains than those listed in the above tables are achieved by

TABLE 8-2-13 CODING GAIN (dB) FOR SOFT-DECISION VITERBI DECODING

	$\mathcal{E}_b/N_0$ uncoded	$R_c = 1/3$		$R_c = 1/2$			$R_c = 2/3$		$R_c = 3/4$	
P_b	(dB)	$K = 7$	$K = 8$	$K = 5$	$K = 6$	$K = 7$	$K = 6$	$K = 8$	$K = 6$	$K = 9$
10^{-3}	6.8	4.2	4.4	3.3	3.5	3.8	2.9	3.1	2.6	2.6
10^{-5}	9.6	5.7	5.9	4.3	4.6	5.1	4.2	4.6	3.6	4.2
10^{-7}	11.3	6.2	6.5	4.9	5.3	5.8	4.7	5.2	3.9	4.8

Source: Jacobs (1974); © IEEE.

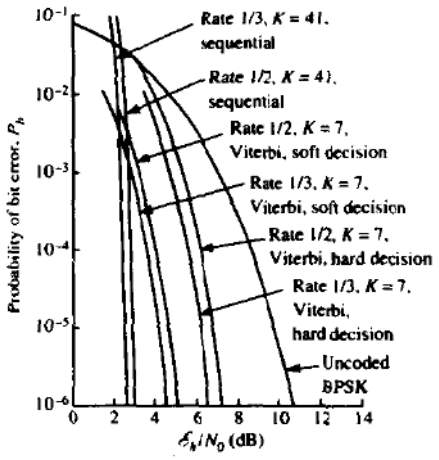


FIGURE 8-2-21 Performance of rate 1/2 and rate 1/3 Viterbi and sequential decoding. [*From Omura and Levitt (1982). © 1982 IEEE.*]

employing long constraint length convolutional codes, e.g., $K = 50$, and decoding such codes by sequential decoding. Invariably, sequential decoders are implemented for hard-decision decoding to reduce complexity. Figure 8-2-21 illustrates the error rate performance of several constraint-length $K = 7$ convolutional codes for rates 1/2 and 1/3 and for sequential decoding (with hard decisions) of a rate 1/2 and a rate 1/3 constraint-length $K = 41$ convolutional codes. Note that the $K = 41$ codes achieve an error rate of 10^{-6} at 2.5 and 3 dB, which are within 4–4.5 dB of the channel capacity limit, i.e., in vicinity of the cutoff rate limit. However, the rate 1/2 and rate 1/3, $K = 7$ codes with soft-decision Viterbi decoding operate at about 5 and 4.4 dB at 10^{-6}, respectively. These short-constraint-length codes achieve a coding gain of about 6 dB at 10^{-6}, while the long constraint codes gain about 7.5–8 dB.

Two important issues in the implementation of Viterbi decoding are

1 the effect of path memory truncation, which is a desirable feature that ensures a fixed decoding delay, and

2 the degree of quantization of the input signal to the Viterbi decoder.

As a rule of thumb, we stated that path memory truncation to about five constraint lengths has been found to result in negligible performance loss. Figure 8-2-22 illustrates the performance obtained by simulation for rate 1/2, constraint-lengths $K = 3$, 5, and 7 codes with memory path length of 32 bits. In addition to path memory truncation, the computations were performed with eight-level (three bits) quantized input signals from the demodulator. The broken curves are performance results obtained from the upper bound in the bit error rate given by (8-2-26). Note that the simulation results are close to the theoretical upper bounds, which indicate that the degradation due to path memory truncation and quantization of the input signal has a minor effect on performance (0.20–0.30 dB).

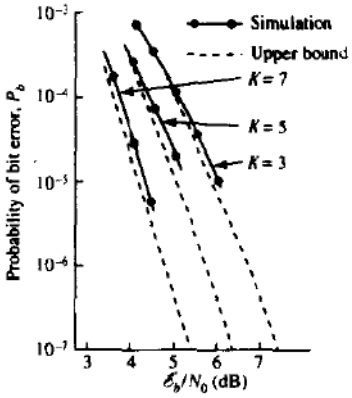


FIGURE 8-2-22 Bit error probability for rate 1/2 Viterbi decoding with eight-level quantized inputs to the decoder and 32-bit path memory. [*From Heller and Jacobs (1971).* © *1971 IEEE.*]

Figure 8-2-23 illustrates the bit error rate performance obtained via simulation for hard-decision decoding of convolutional codes with $K = 3$–8. Note that with the $K = 8$ code, an error rate of 10^{-5} requires about 6 dB, which represents a coding gain of nearly 4 dB relative to uncoded QPSK.

The effect of input signal quantization is further illustrated in Fig. 8-2-24 for a rate 1/2, $K = 5$ code. Note that three-bit quantization (eight levels) is about 2 dB better than hard-decision decoding, which is the ultimate limit between soft-decision decoding and hard-decision decoding on the AWGN channel. The combined effect of signal quantization and path memory trunction for the rate 1/2, $K = 5$ code with 8-, 16-, and 32-bit path memories and either one- or three-bit quantization is shown in Fig. 8-2-25. It is apparent from these results that a path memory as short as three constraint lengths does not seriously degrade performance.

When the signal from the demodulator is quantized to more than two levels, another problem that must be considered is the spacing between quantization levels. Figure 8-2-26 illustrates the simulation results for an eight-level uniform quantizer as a function of the quantizer threshold spacing. We observe that

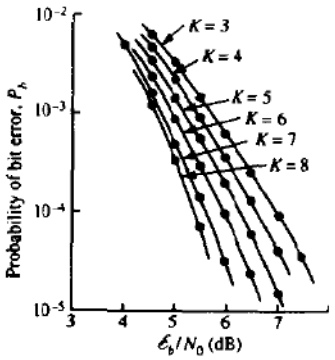


FIGURE 8-2-23 Performance of rate 1/2 codes with hard-decision Viterbi decoding and 32-bit path memory truncation. [*From Heller and Jacobs (1971).* © *1971 IEEE.*]

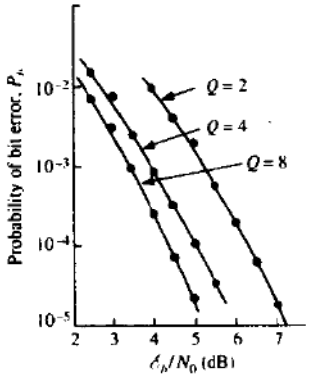


FIGURE 8-2-24 Performance of rate 1/2, $K = 5$ code with eight-, four-, and two-level quantization at the input to the Viterbi decoder. Path truncation length = 32 bits. [*From Heller and Jacobs (1971). © 1971 IEEE.*]

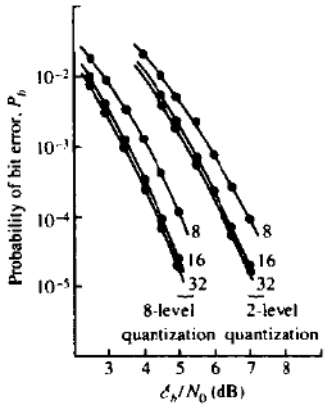


FIGURE 8-2-25 Performance of rate 1/2, $K = 5$ code with 32-, 16-, and 8-bit path memory truncation and eight- and two-level quantization. [*From Heller and Jacobs (1971). © 1971 IEEE.*]

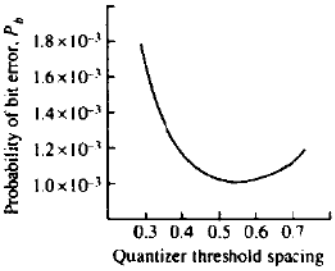


FIGURE 8-2-26 Error rate performance of rate 1/2, $K = 5$ Viterbi decoder for $\mathscr{E}_b/N_0 = 3.5$ dB and eight-level quantization as a function of quantizer threshold level spacing for equally spaced thresholds [*From Heller and Jacobs (1971). © 1971 IEEE.*]

there is an optimum spacing between thresholds (approximately equal to 0.5). However, the optimum is sufficiently broad (0.4–0.7) so that, once it is set, there is little degradation resulting from variations in the AGC level of the order of ±20%.

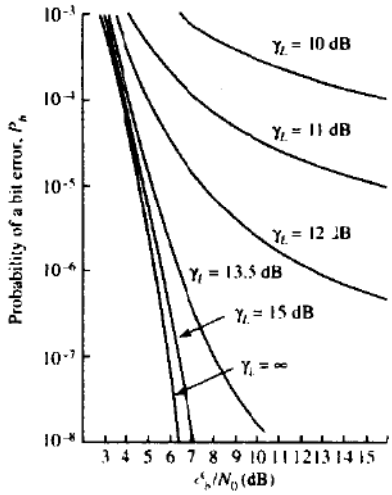


FIGURE 8-2-27 Performance of a rate 1/2, $K = 7$ code with Viterbi decoding and eight-level quantization as a function of the carrier phase tracking loop SNR γ_L. [*From Heller and Jacobs (1971). © 1971 IEEE.*]

Finally, we should point out some important results in the performance degradation due to carrier phase variations. Figure 8-2-27 illustrates the performance of a rate 1/2, $K = 7$ code with eight-level quantization and a carrier phase tracking loop SNR γ_L. Recall that in a PLL, the phase error has a variance that is inversely proportional to γ_L. The results in Fig. 8-2-27 indicate that the degradation is large when the loop SNR is small ($\gamma_L < 12$ dB), and causes the error rate performance to bottom out at relatively high error rate.

8-3 CODED MODULATION FOR BANDWIDTH-CONSTRAINED CHANNELS

In the treatment of block and convolutional codes in Sections 8-1 and 8-2, respectively, performance improvement was achieved by expanding the bandwidth of the transmitted signal by an amount equal to the reciprocal of the code rate. Recall for example that the improvement in performance achieved by an (n, k) binary block code with soft-decision decoding is approximately $10 \log_{10}(R_c d_{\min} - k \ln 2/\gamma_b)$ compared with uncoded binary or quaternary PSK. For example, when $\gamma_b = 10$ the (24, 12) extended Golay code gives a coding gain of 5 dB. This coding gain is achieved at a cost of doubling the bandwidth of the transmitted signal and, of course, at the additional cost in receiver implementation complexity. Thus, coding provides an effective method for trading bandwidth and implementation complexity against transmitter power. This situation applies to digital communications systems that are designed to operate in the power-limited region where $R/W < 1$.

In this section, we consider the use of coded signals for bandwidth-constrained channels. For such channels, the digital communications system is

designed to use bandwidth-efficient multilevel/phase modulation, such as PAM, PSK, DPSK, or QAM, and operates in the region where $R/W > 1$. When coding is applied to the bandwidth-constrained channel, a performance gain is desired without expanding the signal bandwidth. This goal can be achieved by increasing the number of signals over the corresponding uncoded system to compensate for the redundancy introduced by the code.

For example, suppose that a system employing uncoded four-phase PSK modulation achieves an $R/W = 2$ (bits/s)/Hz at an error probability of 10^{-6} For this error rate the SNR per bit is $\gamma_b = 10.5$ dB. We may try to reduce the SNR per bit by use of coded signals, but this must be done without expanding the bandwidth. If we choose a rate $R_c = 2/3$ code, it must be accompanied by an increase in the number of signal points from four (two bits per symbol) to eight (three bits per symbol). Thus, the rate 2/3 code used in conjunction with eight-phase PSK, for example, yields the same data throughput as uncoded four-phase PSK. However, we recall that an increase in the number of signal phases from four to eight requires an additional 4 dB approximately in signal power to maintain the same error rate. Hence, if coding is to provide a benefit, the performance gain of the rate 2/3 code must overcome this 4 dB penalty.

If the modulation is treated as a separate operation independent of the encoding, the use of very powerful codes (large-constraint-length convolutional codes or large-block-length block codes) is required to offset the loss and provide some significant coding gain. On the other hand, if the modulation is an integral part of the encoding process and is designed in conjunction with the code to increase the minimum euclidean distance between pairs of coded signals, the loss from the expansion of the signal set is easily overcome and a significant coding gain is achieved with relatively simple codes. The key to this integrated modulation and coding approach is to devise an effective method for mapping the coded bits into signal points such that the minimum euclidean distance is maximized. Such a method was developed by Ungerboeck (1982), based on the principle of *mapping by set partitioning*. We describe this principle by means of two examples.

Example 8-3-1: An 8-PSK Signal Constellation

Let us partition the eight-phase signal constellation shown in Fig. 8-3-1 into subsets of increasing minimum euclidean distance. In the eight-phase signal set, the signal points are located on a circle of radius $\sqrt{\mathscr{E}}$ and have a minimum distance separation of

$$d_0 = 2\sqrt{\mathscr{E}} \sin \tfrac{1}{8}\pi = \sqrt{(2 - \sqrt{2})\mathscr{E}} = 0.765\sqrt{\mathscr{E}}$$

In the first partitioning, the eight points are subdivided into two subsets of four points each, such that the minimum distance between points increases to $d_1 = \sqrt{2\mathscr{E}}$. In the second level of partitioning, each of the two subsets is subdivided into two subsets of two points, such that the minimum distance increases to $d_2 = 2\sqrt{\mathscr{E}}$. This results in four subsets of two points each.

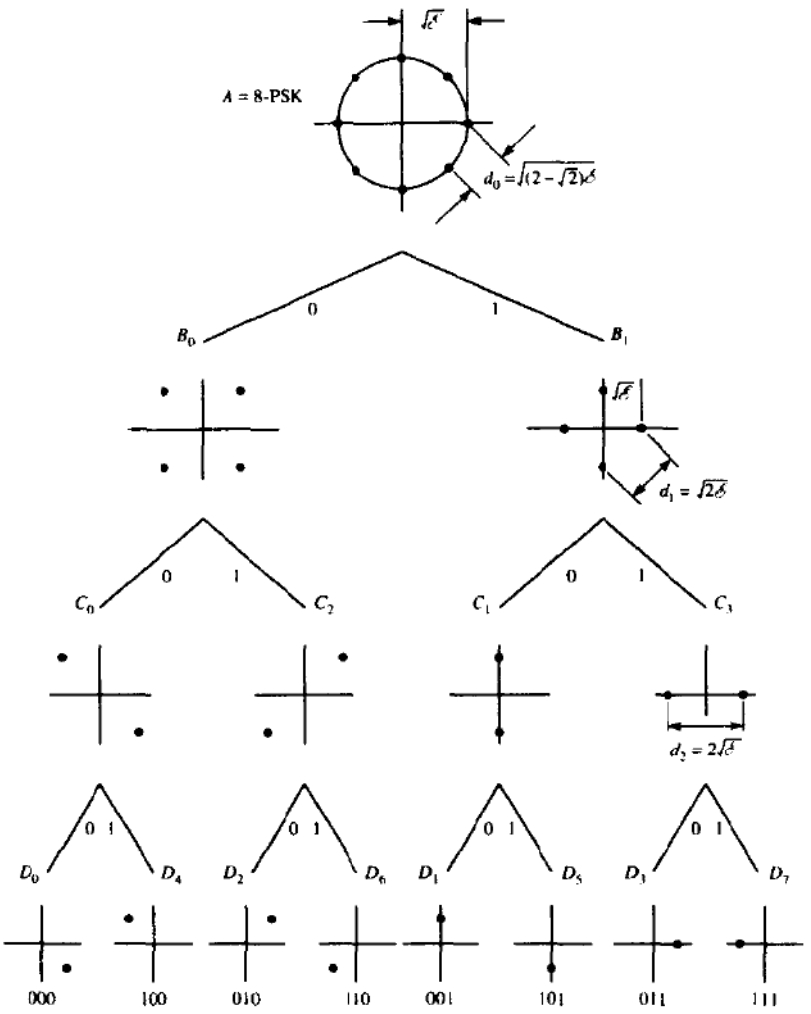


FIGURE 8-3-1 Set partitioning of an 8-PSK signal set.

Finally, the last stage of partitioning leads to eight subsets, where each subset contains a single point. Note that each level of partitioning increases the minimum euclidean distance between signal points. The results of these three stages of partitioning are illustrated in Fig. 8-3-1. The way in which the coded bits are mapped into the partitioned signal points is described below.

Example 8-3-2: A 16-QAM Signal Constellation

The 16-point rectangular signal constellation shown in Fig. 8-3-2 is first divided into two subsets by assigning alternate points to each subset as

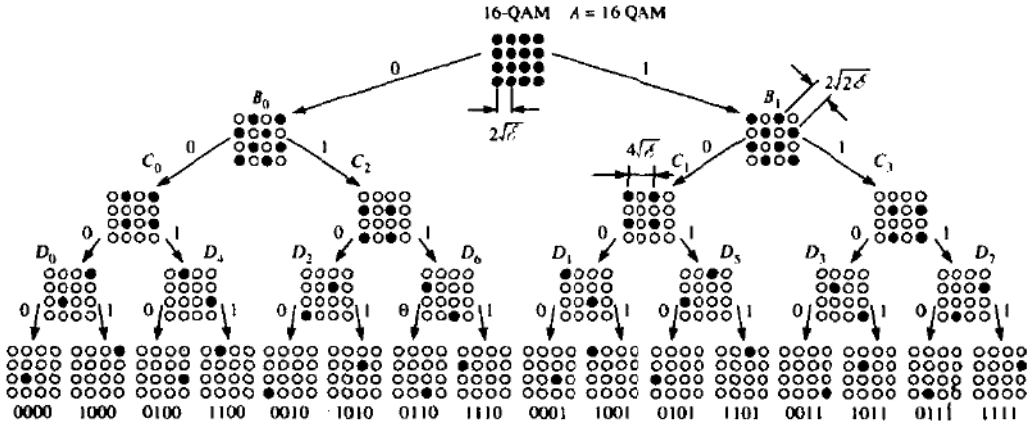


FIGURE 8-3-2 Set partitioning of 16-QAM signal.

illustrated in the figure. Thus, the distance between points is increased from $2\sqrt{\mathscr{E}}$ to $2\sqrt{2\mathscr{E}}$ by the first partitioning. Further partitioning of the two subsets leads to greater separation in euclidean distance between signal points as illustrated in Fig. 8-3-2. It is interesting to note that for the rectangular signal constellations, each level of partitioning increases the minimum euclidean distance by $\sqrt{2}$, i.e., $d_{i+1}/d_i = \sqrt{2}$ for all i.

In these two examples, the partitioning was carried out to the limit where each subset contains only a single point. In general, this may not be necessary. For example, the 16-point QAM signal constellation may be partitioned only twice, to yield four subsets of four points each. Similarly, the eight-phase PSK signal constellation can be partitioned twice, to yield four subsets of two points each.

The degree to which the signal is partitioned depends on the characteristics of the code. In general, the encoding process is performed as illustrated in Fig. 8-3-3. A block of m information bits is separated into two groups of length k_1

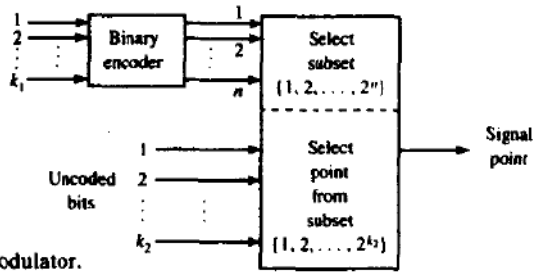


FIGURE 8-3-3 General structure of combined encoder/modulator.

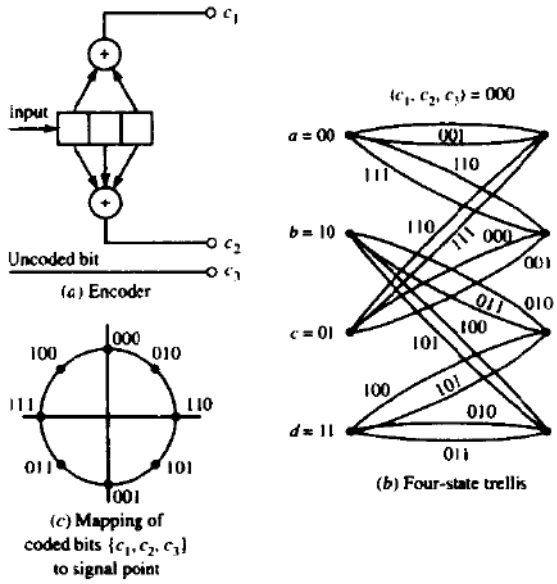


FIGURE 8-3-4 Four-state trellis-coded 8-PSK modulation.

and k_2. The k_1 bits are encoded into n bits while the k_2 bits are left uncoded. Then, the n bits from the encoder are used to select one of the 2^n possible subsets in the partitioned signal set while the k_2 bits are used to select one of the 2^{k_2} signal points in each subset. When $k_2 = 0$, all m information bits are encoded.

Example 8-3-3

Consider the use of the rate 1/2 convolutional code shown in Fig. 8-3-4 to encode one information bit while the second information bit is left uncoded. When used in conjunction with an eight-point signal constellation, e.g., eight-phase PSK or eight-point QAM, the two encoded bits are used to select one of the four subsets in the signal constellation, while the remaining information bit is used to select one of the two points within each subset. In this case, $k_1 = 1$ and $k_2 = 1$. The four-state trellis, which is shown in Fig. 8-3-4(b), is basically the trellis for the rate 1/2 convolution encoder with the addition of parallel paths in each transition to accommodate the uncoded bit c_3. Thus, the coded bits (c_1, c_2) are used to select one of the four subsets that contain two signal points each, while the uncoded bit is used to select one of the two signal points within each subset. Note that signal points within a subset are separated in distance by $d_2 = 2\sqrt{2}$. Hence, the euclidean distance between parallel paths is d_2. The mapping of coded bits (c_1, c_2, c_3) to signal points is illustrated in Fig. 8-3-4(c). As an alternative coding scheme, we may use a rate 2/3 convolutional encoder, and, thus, encode

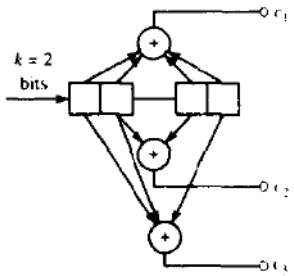


FIGURE 8-3-5 Rate 2/3 convolutional encoder for encoding both information bits.

both information bits as shown in Fig. 8-3-5. This encoding leads to an eight-state trellis and results in better performance, but also requires a more complex implementation of the decoder as described below.

Either block codes or convolutional codes may be used in conjunction with the partitioned signal constellation. In general, convolutional codes provide comparable coding gains to block codes and the availability of the Viterbi algorithm results in a simpler implementation for soft-decision decoding. For this reason, we limit our discussion to convolutional codes (linear trellis codes) and more generally to (nonlinear) trellis codes.

Trellis-Coded Modulation Let us consider the use of the 8-PSK signal constellation in conjunction with trellis codes. Uncoded four-phase PSK (4-PSK) is used as a reference in measuring coding gain. Uncoded 4-PSK employs the signal points in either subset B_0 or B_1 of Fig. 8-3-1, for which the minimum distance of the signal points is $\sqrt{2\mathscr{E}}$. Note that this signal corresponds to a trivial one-state trellis with four parallel state transitions as shown in Fig. 8-3-6(a). The subsets D_0, D_2, D_4, and D_6 are used as the signal points for the purpose of illustration.

For the coded 8-PSK modulation, we may use the four-state trellis shown in Fig. 8-3-6(b). Note that each branch in the trellis corresponds to one of the four subsets C_0, C_1, C_2, or C_3. For the eight-point constellation, each of the subsets C_0, C_1, C_2, and C_3, contains two signal points. Hence, the state transition C_0 contains the two signal points corresponding to the bits (000, 100) or (0, 4) in octal representation. Similarly, C_2 contains the two signal points corresponding to (010, 110), or to (2, 6) in octal, C_1 contains the points corresponding to (001, 101), or (1, 5) in octal, and C_3 contains the points corresponding to (011, 111), or (3, 7) in octal. Thus, each transition in the four-state trellis contains two parallel paths, as shown in more detail in Fig. 8-3-6(c). Note that any two signal paths that diverge from one state and remerge at the same state after more than one transition have a squared euclidean distance of $d_0^2 + 2d_1^2 = d_0^2 + d_2^2$ between them. For example, the

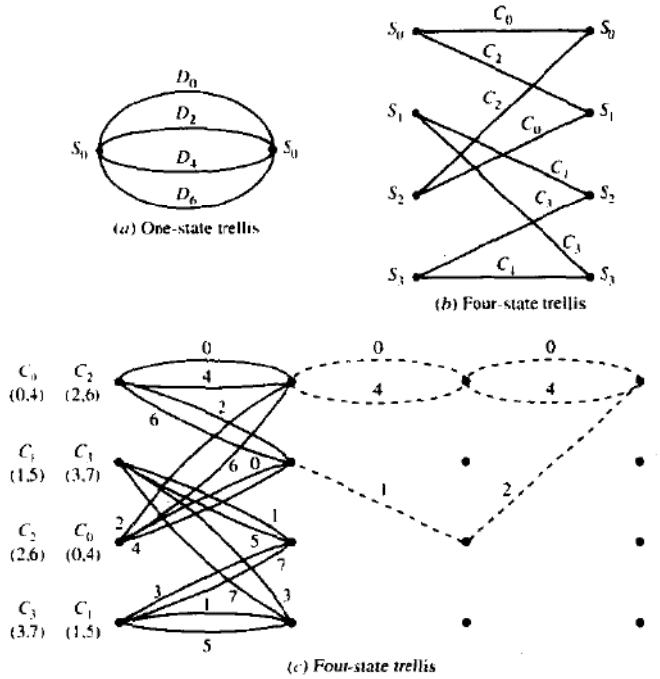


FIGURE 8-3-6 Uncoded 4-PSK and trellis-coded 8-PSK modulation.

signal paths 0, 0, 0 and 2, 1, 2 are separated by $d_0^2 + d_2^2 = [(0.765)^2 + 4]\mathscr{E} = 4.585\mathscr{E}$. On the other hand, the squared euclidean distance between parallel transitions is $d_2^2 = 4\mathscr{E}$. Hence, the minimum euclidean distance separation between paths that diverge from any state and remerge at the same state in the four-state trellis is $d_2 = 2\sqrt{\mathscr{E}}$. This minimum distance in the trellis code is called the *free euclidean distance* and denoted by D_{fed}.

In the four-state trellis of Fig. 8-3-6(b), $D_{\text{fed}} = 2\sqrt{\mathscr{E}}$. When compared with the euclidean distance $d_0 = \sqrt{2\mathscr{E}}$ for the uncoded 4-PSK modulation, we observe that the four-state trellis code gives a coding gain of 3 dB.

We should emphasize that the four-state trellis code illustrated in Fig. 8-3-6(b) is optimum in the sense that it provides the largest free euclidean distance. Clearly, many other four-state trellis codes can be constructed, including the one shown in Fig. 8-3-7, which consists of four distinct transitions from each state to all other states. However, neither this code nor any of the other possible four-state trellis codes gives a larger D_{fed}.

The construction of the optimum four-state trellis code for the eight-point constellation was performed on the basis of the following heuristic rules:

(a) Parallel transitions (when they occur) are assigned to signal points separated by the maximum euclidean distance, e.g., $d_2 = 2\sqrt{\mathscr{E}}$ for 8-PSK in the four subsets C_0, C_1, C_2, C_3.

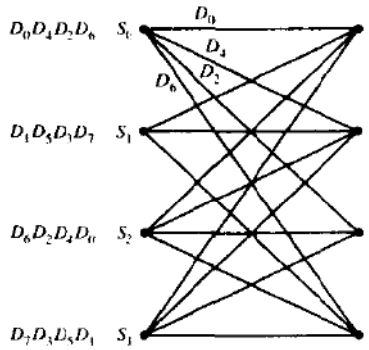


FIGURE 8-3-7 An alternative four-state trellis code.

(b) The transition originating from and merging into any state is assigned the subsets (C_0, C_2) or (C_1, C_3), which have a maximum distance $d_1 = \sqrt{2\mathscr{E}}$.

(c) The signal points should occur with equal frequency.

Note that rules (a) and (b) guarantee that the euclidean distance associated with single and multiple paths that diverge from any state and remerge in that state exceeds the euclidean distance of uncoded 4-PSK. Rule (c) guarantees that the trellis code will have a regular structure.

We should indicate that the specific mapping of coded bits into signal points, as illustrated in Fig. 8-3-1, where the eight signal points are represented in an equivalent binary form, is not important. Other mappings can be devised by permuting subsets in a way that preserves the main property of increased minimum distance among the subsets.

In the four-state trellis code, the parallel transitions were separated by the euclidean distance $2\sqrt{\mathscr{E}}$, which is also D_{fed}. Hence, the coding gain of 3 dB is limited by the distance of the parallel transitions. Larger gains in performance relative to uncoded 4-PSK can be achieved by using trellis codes with more states, which allow for the elimination of the parallel transitions. Thus, trellis codes with eight or more states would use distinct transitions to obtain a larger D_{fed}.

For example, in Fig. 8-3-8, we illustrate an eight-state trellis code due to Ungerboeck (1982) for the 8-PSK signal constellation. The state transitions for maximizing the free euclidean distance were determined from application of the three basic rules given above. In this case, note that the minimum squared euclidean distance is

$$D_{\text{fed}}^2 = d_0^2 + 2d_1^2 = 4.585\mathscr{E}$$

which, when compared with $d_0^2 = 2\mathscr{E}$ for uncoded 4-PSK, represents a gain of 3.6 dB. Ungerboeck (1982, 1987) has also found rate 2/3 trellis codes with 16, 32, 64, 128, and 256 states that achieve coding gains ranging from 4 to 5.75 dB for 8-PSK modulation.

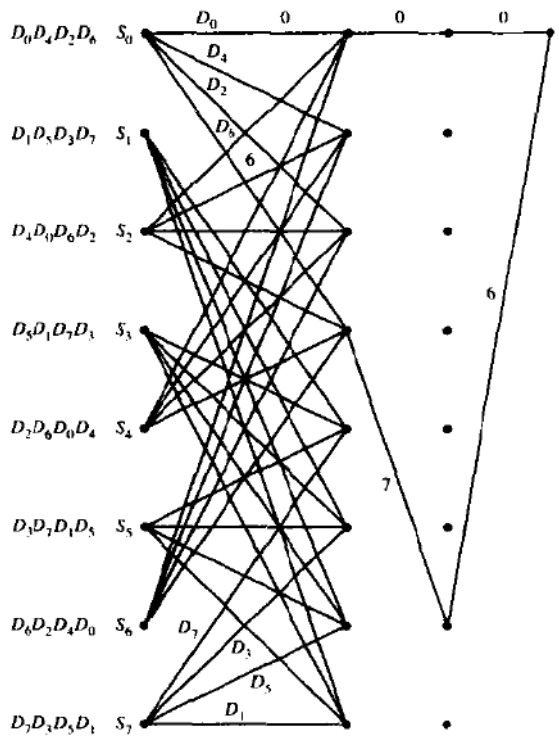


FIGURE 8-3-8 Eight-state trellis code for coded 8-PSK modulation.

The basic principle of set partitioning is easily extended to larger PSK signal constellations that yield greater bandwidth efficiency. For example, 3 (bits/s)/Hz can be achieved with either uncoded 8-PSK or with trellis-coded 16-PSK modulation. Ungerboeck (1987) has devised trellis codes and has evaluated the coding gains achieved by simple rate 1/2 and rate 2/3 convolutional codes for the 16-PSK signal constellations. The results are summarized below.

Soft-decision Viterbi decoding for trellis-coded modulation is accomplished in two steps. Since each branch in the trellis corresponds to a signal subset, the first step in decoding is to determine the best signal point within each subset, i.e., the point in each subset that is closest in distance to the received point. We may call this *subset decoding*. In the second step, the signal point selected from each subset and its squared distance metric are used for the corresponding branch in the Viterbi algorithm to determine the signal path through the code trellis that has the minimum sum of squared distances from the sequence of received (noisy channel output) signals.

The error rate performance of the trellis-coded signals in the presence of additive gaussian noise can be evaluated by following the procedure described in Section 8-2 for convolutional codes. Recall that this procedure involves the computation of the probability of error for all different error events and

summing these error event probabilities to obtain a union bound on the first-event error probability. Note, however, that, at high SNR, the first-event error probability is dominated by the leading term, which has the minimum distance D_{fed}. Consequently, at high SNR, the first-event error probability is well approximated as

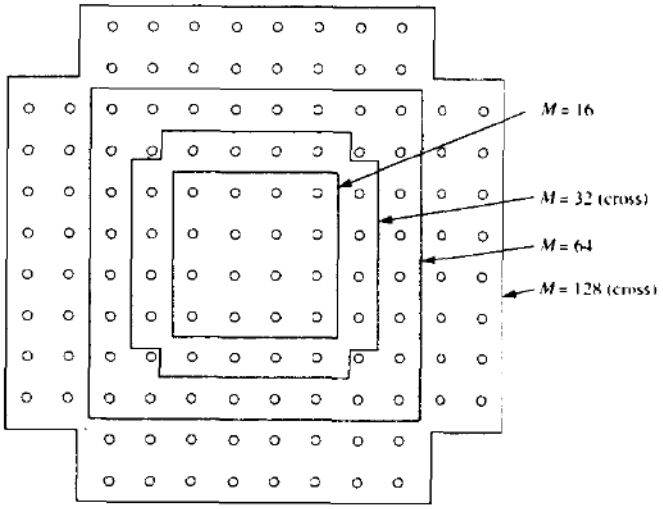
$$P_e \approx N_{\text{fed}} Q\left(\sqrt{\frac{D_{\text{fed}}^2}{2N_0}}\right) \tag{8-3-1}$$

where N_{fed} denotes the number of signal sequences with distance D_{fed} that diverge at any state and remerge at that state after one or more transitions.

In computing the coding gain achieved by trellis-coded modulation, we usually focus on the gain achieved by increasing D_{fed} and neglect the effect of N_{fed}. However, trellis codes with a large number of states may result in a large N_{fed} that cannot be ignored in assessing the overall coding gain.

In addition to the trellis-coded PSK modulations described above, powerful trellis codes have also been developed for PAM and QAM signal constellations. Of particular practical importance is the class of trellis-coded two-dimensional rectangular signal constellations. Figure 8-3-9 illustrates these signal constellations for M-QAM where $M = 16$, 32, 64, and 128. The $M = 32$ and 128 constellations have a cross pattern and are sometimes called *cross-constellations*. The underlying rectangular grid containing the signal points in M-QAM is called a *lattice of type* Z_2 (the subscript indicates the dimensionality of the space). When set partitioning is applied to this class of signal constellations, the minimum euclidean distance between successive partitions is $d_{i+1}/d_i = \sqrt{2}$ for all i, as previously observed in Example 8-3-2.

FIGURE 8-3-9 Rectangular two-dimensional (QAM) signal constellations.



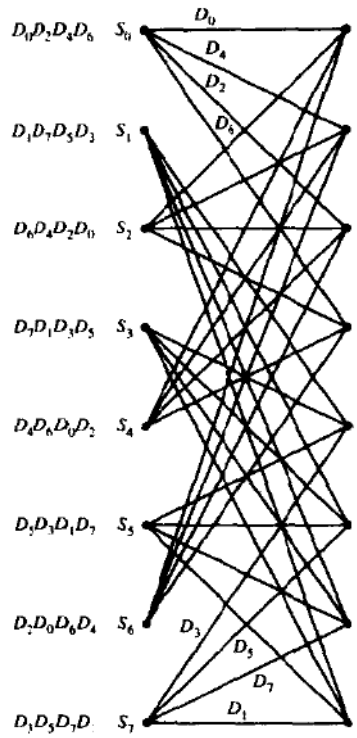


FIGURE 8-3-10 Eight-state trellis for rectangular QAM signal constellations.

Figure 8-3-10 illustrates an eight-state trellis code that can be used with any of the M-QAM rectangular signal constellations for which $M = 2^k$, where $k = 4, 5, 6, \ldots$, etc. With the eight-state trellis, we associate eight signal subsets, so that any of the M-QAM signals sets for $M \geq 16$ are suitable. For $M = 2^{m+1}$, two input bits ($k_1 = 2$) are encoded into $n = 3$ ($n = k_1 + 1$) bits that are used to select one of the eight states corresponding to the eight subsets. The additional $k_2 = m - k_1$ input bits are used to select signal points within a subset, and result in parallel transitions in the eight-state trellis. Hence, 16-QAM involves two parallel transitions in each branch of the trellis. More generally, the choice of an $M = 2^{m+1}$-point QAM signal constellation implies that the eight-state trellis contains 2^{m-2} parallel transitions in each branch.

The assignment of signal subsets to transitions is based on the same set of basic (heuristic) rules described above for the 8-PSK signal constellation. Thus, the four (branches) transitions originating from or leading to the same state are assigned either the subsets D_0, D_2, D_4, D_6 or D_1, D_3, D_5, D_7. Parallel transitions are assigned signal points contained within the corresponding subsets. This eight-state trellis code provides a coding gain of 4 dB. The euclidean distance of parallel transitions exceeds the free euclidean distance, and, hence, the code performance is not limited by parallel transitions.

Larger size trellis codes for M-QAM provide even larger coding gains. For

TABLE 8-3-1 CODING GAINS FOR TRELLIS-CODED PAM SIGNALS

Number of states	k_1	Code rate $\frac{k_1}{k_1+1}$	$m=1$ coding gain (dB) of 4-PAM versus uncoded 2-PAM	$m=2$ coding gain (dB) of 8-PAM versus uncoded 4-PAM	$m\to\infty$ asymptotic coding gain (dB)	$m\to\infty$ N_{fed}
4	1	1/2	2.55	3.31	3.52	4
8	1	1/2	3.01	3.77	3.97	4
16	1	1/2	3.42	4.18	4.39	8
32	1	1/2	4.15	4.91	5.11	12
64	1	1/2	4.47	5.23	5.44	36
128	1	1/2	5.05	5.81	6.02	66

Source: Ungerboeck (1987).

example, trellis codes with 2^ν states for an $M=2^{m+1}$ QAM signal constellation can be constructed by convolutionally encoding k_1 input bits into k_1+1 output bits. Thus, a rate $R_c=k_1/(k_1+1)$ convolutional code is employed for this purpose. Usually, the choice of $k_1=2$ provides a significant fraction of the total coding gain that is achievable. The additional $k_2=m-k_1$ input bits are uncoded, and are transmitted in each signal interval by selecting signal points within a subset.

Tables 8-3-1 to 8-3-3, taken from the paper by Ungerboeck (1987), provide a summary of coding gains achievable with trellis-coded modulation. Table 8-3-1 summarizes the coding gains achieved for trellis-coded (one-dimensional) PAM modulation with rate 1/2 trellis codes. Note that the coding gain with a 128-state trellis code is 5.8 dB for octal PAM, which is close to the channel cutoff rate R_0 and less than 4 dB from the channel capacity limit for error rates in the range of 10^{-6}–10^{-8}. We should also observe that the number of paths

TABLE 8-3-2 CODING GAINS FOR TRELLIS-CODED 16-PSK MODULATION

Number of states	k_1	Code rate $\frac{k_1}{k_1+1}$	$m=3$ coding gain (dB) of 16-PSK versus uncoded 8-PSK	$m\to\infty$ N_{fed}
4	1	1/2	3.54	4
8	1	1/2	4.01	4
16	1	1/2	4.44	8
32	1	1/2	5.13	8
64	1	1/2	5.33	2
128	1	1/2	5.33	2
256	2	2/3	5.51	8

Source: Ungerboeck (1987).

TABLE 8-3-3 CODING GAINS FOR TRELLIS-CODED QAM MODULATION

Number of states	k_1	Code rate $\frac{k_1}{k_1+1}$	$m=3$ gain (dB) of 16-QAM versus uncoded 8-QAM	$m=4$ gain (dB) of 32-QAM versus uncoded 16-QAM	$m=5$ gain (dB) of 64-QAM versus uncoded 32-QAM	$m=\infty$ asymptotic coding gain (dB)	N_{fed}
4	1	1/2	3.01	3.01	2.80	3.01	4
8	2	2/3	3.98	3.98	3.77	3.98	16
16	2	2/3	4.77	4.77	4.56	4.77	56
32	2	2/3	4.77	4.77	4.56	4.77	16
64	2	2/3	5.44	5.44	4.23	5.44	56
128	2	2/3	6.02	6.02	5.81	6.02	344
256	2	2/3	6.02	6.02	5.81	6.02	44

Source: Ungerboeck (1987).

N_{fed} with free euclidean distance D_{fed} becomes large with an increase in the number of states.

Table 8-3-2 lists the coding gain for trellis-coded 16-PSK. Again, we observe that the coding gain for eight or more trellis stages exceeds 4 dB, relative to uncoded 8-PSK. A simple rate 1/2 code yields 5.33 dB gain with a 128-stage trellis.

Table 8-3-3 contains the coding gains obtained with trellis-coded QAM signals. Relatively simple rate 2/3 trellis codes yield a gain of 6 dB with 128 trellis stages for $m = 3$ and 4.

The results in these tables clearly illustrate the significant coding gains that are achievable with relatively simple trellis codes. A 6 dB coding gain is close to the cutoff rate R_0 for the signal sets under consideration. Additional gains that would lead to transmission in the vicinity of the channel capacity bound are difficult to attain without a significant increase in coding/decoding complexity.

Since the channel capacity provides the ultimate limit on code performance, we should emphasize that continued partitioning of large signal sets quickly leads to signal point separation within any subset that exceeds the free euclidean distance of the code. In such cases, parallel transitions are no longer the limiting factor on D_{fed}. Usually, a partition to eight subsets is sufficient to obtain a coding gain of 5–6 dB with simple rate 1/2 or rate 2/3 trellis codes with either 64 or 128 trellis stages, as indicated in Tables 8-3-1 to 8-3-3.

Convolutional encoders for the linear trellis codes listed in Tables 8-3-1 to 8-3-3 for the M-PAM, M-PSK, and M-QAM signal constellations are given in the papers by Ungerboeck (1982, 1987). The encoders may be realized either with feedback or without feedback. For example Fig. 8-3-11 illustrates three feedback-free convolutional encoders corresponding to 4-, 8-, and 16-state trellis codes for 8-PSK and 16-QAM signal constellations. Equivalent realizations of these trellis codes based on systematic convolutional encoders with

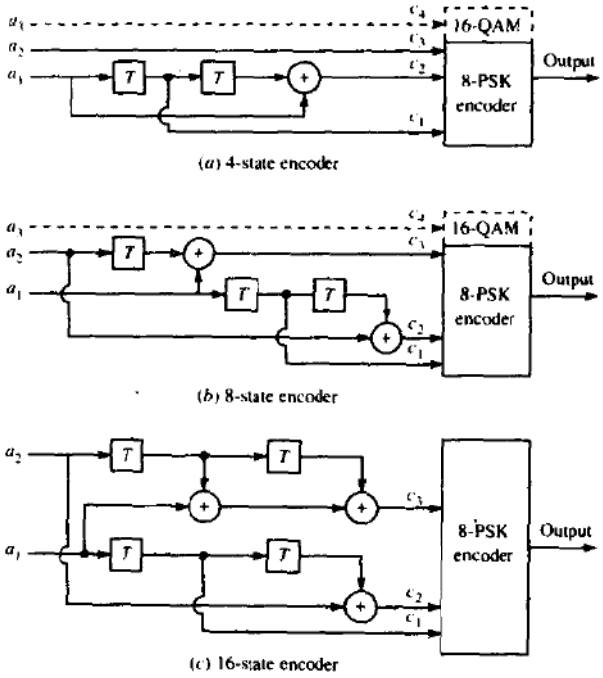


FIGURE 8-3-11 Minimal feedback-free convolutional encoders for 8-PSK and 16-QAM signals. [*From Ungerboeck (1982).* © *1982 IEEE.*]

feedback are shown in Fig. 8-3-12. Usually, the systematic convolutional encoders are preferred in practical applications.

A potential problem with linear trellis codes is that the modulated signal sets are not usually invariant to phase rotations. This poses a problem in practical applications where differential encoding is usually employed to avoid phase ambiguities when a receiver must recover the carrier phase after a temporary loss of signal. The problem of phase invariance and differential encoding/decoding was solved by Wei (1984a, b), who devised linear and nonlinear trellis codes that are rotationally invariant under either 180° or 90° phase rotations, respectively. For example, Fig. 8-3-13 illustrates a nonlinear eight-state convolutional encoder for a 32-QAM rectangular signal constellation that is invariant under 90° phase rotations. This trellis code has been adopted as an international standard for 9600 and 14,000 bits/s (high-speed) telephone line modems.

Trellis-coded modulation schemes have also been developed for multidimensional signals. In practical systems, multidimensional signals are transmitted as a sequence of either one-dimensional (PAM) or two-dimensional (QAM) signals. Trellis codes based on 4-, 8-, and 16-dimensional signal constellations have been constructed, and some of these codes have been

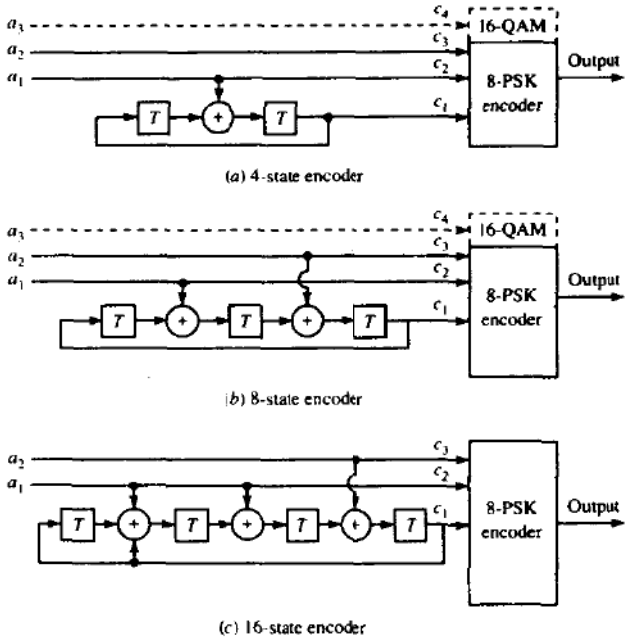
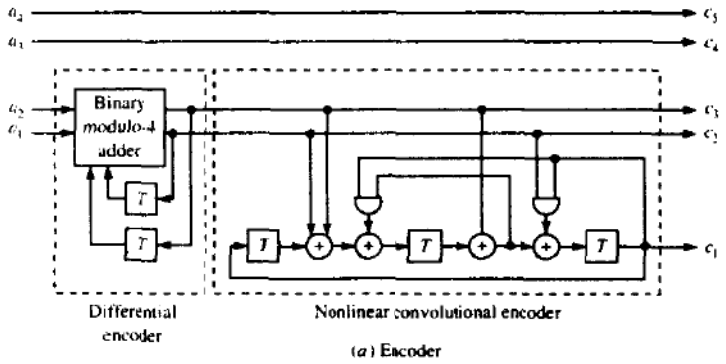


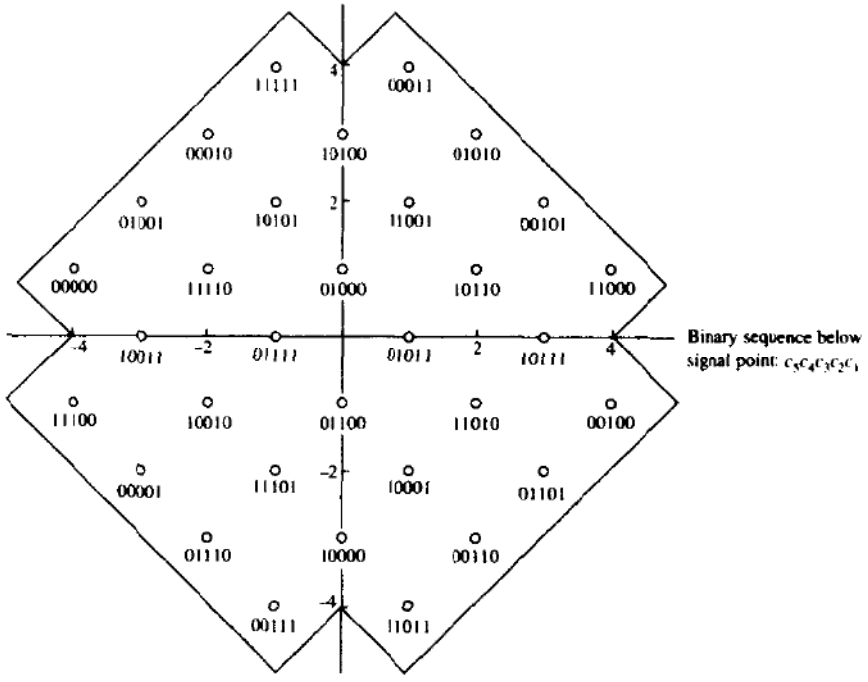
FIGURE 8-3-12 Equivalent realizations of systematic convolutional encoders with feedback for 8-PSK and 16-QAM. [*From Ungerboeck (1982).* © *1982 IEEE.*]

implemented in commercially available modems. A potential advantage of trellis-coded multidimensional signals is that we can use smaller constituent two-dimensional signal constellations that allow for a trade-off between coding gain and implementation complexity. The papers by Wei (1987), Ungerboeck (1987), Gersho and Lawrence (1984), and Forney *et al.* (1984) treat multidimensional signal constellations for trellis-coded modulation.

Finally, we should mention that a new design technique for trellis-coded modulation based on lattices and cosets of a sublattice has been described by Calderbank and Sloane (1987) and Forney (1988). This method for constructing trellis codes provides an alternative to the set partitioning method described above. However, the two methods are closely related. In this alternative method, a block of k_1 bits is fed to a convolutional encoder. Each block of k_1 input bits produces an output symbol that is a coset of the sublattice Λ', which is a subset of the chosen lattice. A second block of k_2 input bits is used to select one of the points in the coset at the output of the convolutional encoder. It is apparent that the cosets of the sublattice are akin to the subsets in set partitioning and the elements of the cosets are akin to the signal points within a subset. This new method has led to the discovery of new powerful trellis codes involving larger signal constellations, many of which are listed in the paper by Calderbank and Sloane (1987).



(a) Encoder



(b) 32-point QAM (cross) signal

FIGURE 8-3-13 Eight-state nonlinear convolutional encoder for 32-QAM signal set that exhibits invariance under 90° phase rotations.

8-4 BIBLIOGRAPHICAL NOTES AND REFERENCES

The pioneering work on coding and coded waveforms for digital communications was done by Shannon (1948a, b), Hamming (1950), and Golay (1949). These works were rapidly followed with papers on code performance by

Gilbert (1952), new codes by Muller (1954) and Reed (1954), and coding techniques for noisy channels by Elias (1954, 1955) and Slepian (1956).

During the period 1960–1970, there were a number of significant contributions in the development of coding theory and decoding algorithms. In particular, we cite the papers by Reed and Solomon (1960) on Reed–Solomon codes, the papers by Hocquenghem (1959) and Bose and Ray-Chaudhuri (1960a, b) on BCH codes, and the Ph.D dissertation of Forney (1966a) on concatenated codes. These works were followed by the papers of Goppa (1970, 1971) on the construction of a new class of linear cyclic codes, now called Goppa codes (see also Berlekamp, 1973), and the paper of Justesen (1972) on a constructive technique for asymptotically good codes. During this period, work on decoding algorithms was primarily focused on BCH codes. The first decoding algorithm for binary BCH codes was developed by Peterson (1960). A number of refinements and generalizations by Chien (1964), Forney (1965), Massey (1965), and Berlekamp (1968) led to the development of a computationally efficient algorithm for BCH codes, which is described in detail by Lin and Costello (1983).

In parallel with these developments on block codes are the developments in convolutional codes, which were invented by Elias (1955). The major problem in convolutional coding was decoding. Wozencraft and Reiffen (1961) described a sequential decoding algorithm for convolutional codes. This algorithm was later modified and refined by Fano (1963), and it is now called the *Fano algorithm*. Subsequently, the stack algorithm was devised by Ziganzirov (1966) and Jelinek (1969), and the Viterbi algorithm was devised by Viterbi (1967). The optimality and the relatively modest complexity for small constraint lengths have served to make the Viterbi algorithm the most popular in decoding of convolutional codes with $K \leq 10$.

One of the most important contributions in coding during the 1970s was the work of Ungerboeck and Csajka (1976) on coding for bandwidth-constrained channels. In this paper, it was demonstrated that a significant coding gain can be achieved through the introduction of redundancy in a bandwidth-constrained channel and trellis codes were described for achieving coding gains of 3–4 dB. This work has generated much interest among researchers and has led to a large number of publications over the past 10 years. A number of references can be found in the papers by Ungerboeck (1982, 1987) and Forney *et al.* (1984). Additional papers on coded modulation for bandwidth-constrained channels may also be found in the Special Issue on Voiceband Telephone Data Transmission, *IEEE Journal on Selected Areas in Communication* (September 1984). A comprehensive treatment of trellis-coded modulation is given in the book by Biglieri *et al.* (1991).

In addition to the references given above on coding, decoding, and coded signal design, we should mention the collection of papers published by the IEEE Press entitled *Key Papers in the Development of Coding Theory*, edited by Berlekamp (1974). This book contains important papers that were published in the first 25 years of coding theory. We should also cite the Special

Issue on Error-Correcting Codes, *IEEE Transactions on Communications* (October 1971).

PROBLEMS

8-1 The generator matrix for a linear binary code is

$$\mathbf{G} = \begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix}$$

a Express **G** in systematic $[\mathbf{I} \mid \mathbf{P}]$ form.
b Determine the parity check matrix **H** for the code.
c Construct the table of syndromes for the code.
d Determine the minimum distance of the code.
e Demonstrate that the code word corresponding to the informatin sequence 101 is orthogonal to **H**.

8-2 List the code words generated by the matrices given in (8-1-35) and (8-1-37), and, thus, demonstrate that these matrices generate the same set of code words.

8-3 The weight distribution of Hamming codes is known. Expressed as a polynomial in powers of x, the weight distribution for the binary Hamming codes of block length n is

$$A(x) = \sum_{i=0}^{n} A_i x^i$$

$$= \frac{1}{n+1}[(1+x)^n + n(1+x)^{(n-1)/2}(1-x)^{(n+1)/2}]$$

where A_i is the number of code words of weight i. Use this formula to determine the weight distribution of the (7, 4) Hamming code and check your result with the list of code words given in Table 8-1-2.

8-4 The polynomial

$$g(p) = p^4 + p + 1$$

is the generator for the (15, 11) Hamming binary code.
a Determine a generator matrix **G** for this code in systematic form.
b Determine the generator polynomial for the dual code.

8-5 For the (7, 4) cyclic Hamming code with generator polynomial $g(p) = p^3 + p^2 + 1$, construct an (8, 4) extended Hamming code and list all the code words. What is $d_{\min}$ for the extended code?

8-6 An (8, 4) linear block code is constructed by shortening a (15, 11) Hamming code generated by the generator polynomial $g(p) = p^4 + p + 1$.
a Construct the code words of the (8, 4) code and list them.
b What is the minimum distance of the (8, 4) code?

8-7 The polynomial $p^{15} + 1$ when factored yields

$$p^{15} + 1 = (p^4 + p^3 + 1)(p^4 + p^3 + p^2 + p + 1) \times (p^4 + p + 1)(p^2 + p + 1)(p + 1)$$

a Construct a systematic (15, 5) code using the generator polynomial

$$g(p) = (p^4 + p^3 + p^2 + p + 1)(p^4 + p + 1)(p^2 + p + 1)$$

b What is the minimum distance of the code?
c How many random errors per code word can be corrected?
d How many errors can be detected by this code?
e List the code words of a (15, 2) code constructed from the generator polynomial

$$g(p) = (p^{15} + 1)/(p^2 + p + 1)$$

and determine the minimum distance.

8-8 Construct the parity check matrices $\mathbf{H}_1$ and $\mathbf{H}_2$ corresponding to the generator matrices $\mathbf{G}_1$ and $\mathbf{G}_2$ given by (8-1-34) and (8-1-35), respectively.

8-9 Construct an extended (8, 4) code from the (7, 4) Hamming code by specifying the generator matrix and the parity check matrix.

8-10 A systematic (6, 3) code has the generator matrix

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}$$

Construct the standard array and determine the correctable error patterns and their corresponding syndromes.

8-11 Construct the standard array for the (7, 3) code with generator matrix

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{bmatrix}$$

and determine the correctable patterns and their corresponding syndromes.

8-12 Determine the correctable error patterns (of least weight) and their syndromes for the systematic (7, 4) cyclic Hamming code.

8-13 Prove that if the sum of two error patterns $\mathbf{e}_1$ and $\mathbf{e}_2$ is a valid code word $\mathbf{C}_i$ then each pattern has the same syndrome.

8-14 Let $g(p) = p^8 + p^6 + p^4 + p^2 + 1$ be a polynomial over the binary field.
a Find the lowest-rate cyclic code whose generator polynomial is $g(p)$. What is the rate of this code?
b Find the minimum distance of the code found in (a).
c What is the coding gain for the code found in (a).

8-15 The polynomial $g(p) = p + 1$ over the binary field is considered.
a Show that this polynomial can generate a cyclic code for any choice of n. Find the corresponding k.
b Find the systematic form of $\mathbf{G}$ and $\mathbf{H}$ for the code generated by $g(p)$.
c Can you say what type of code this generator polynomial generates?

8-16 Design a (6, 2) cyclic code by choosing the shortest possible generator polynomial.
a Determine the generator matrix $\mathbf{G}$ (in the systematic form) for this code and find all possible code words.
b How many errors can be corrected by this code?

8-17 Prove that any two n-tuples in the same row of a standard array add to produce a valid code word.

8-18 Beginning with a (15, 7) BCH code, construct a shortened (12, 4) code. Give the generator matrix for the shortened code.

8-19 In Section 8-1-2, it was indicated that when an (n, k) Hadamard code is mapped into waveforms by means of binary PSK, the corresponding $M = 2^k$ waveforms

are orthogonal. Determine the bandwidth expansion factor for the M orthogonal waveforms and compare this with the bandwidth requirements of orthogonal FSK detected coherently.

8-20 Show that the signaling waveforms generated from a maximum-length shift-register code by mapping each bit in a code word into a binary PSK signal are equicorrelated with correlation coefficient $\rho_x = -1/(M-1)$, i.e., the M waveforms form a simplex set.

8-21 Compute the error probability obtained with a (7, 4) Hamming code on an AWGN channel, both for hard-decision and soft-decision decoding. Use (8-1-50), (8-1-52), (8-1-82), (8-1-90), and (8-1-91).

8-22 Use the results in Section 2-1-6 to obtain the Chernoff bound for hard-decision decoding given by (8-1-89) and (8-1-90). Assume that the all-zero code word is transmitted and determine an upper bound on the probability that code word C_m, having weight w_m, is selected. This occurs if $\frac{1}{2}w_m$ or more bits are in error. To apply the Chernoff bound, define a sequence of w_m random variables as

$$X_i = \begin{cases} 1 & \text{with probability } p \\ -1 & \text{with probability } 1-p \end{cases}$$

where $i = 1, 2, \ldots, w_m$, and p is the probability of error. For the BSC, the $\{X_i\}$ are statistically independent.

8-23 A convolutional code is described by

$$\mathbf{g}_1 = [1 \quad 0 \quad 0], \quad \mathbf{g}_2 = [1 \quad 0 \quad 1], \quad \mathbf{g}_3 = [1 \quad 1 \quad 1]$$

a Draw the encoder corresponding to this code.
b Draw the state-transition diagram for this code.
c Draw the trellis diagram for this code.
d Find the transfer function and the free distance of this code.
e Verify whether or not this code is catastrophic.

8-24 The convolutional code of Problem 8-23 is used for transmission over a AWGN channel with hard-decision decoding. The output of the demodulator detector is (101001011110111 . . .). Using the Viterbi algorithm, find the transmitted sequence.

8-25 Repeat Problem 8-23 for a code with

$$\mathbf{g}_1 = [1 \quad 1 \quad 0], \quad \mathbf{g}_2 = [1 \quad 0 \quad 1], \quad \mathbf{g}_3 = [1 \quad 1 \quad 1]$$

8-26 The block diagram of a binary convolutional code is shown in Fig. P8-26.
a Draw the state diagram for the code.
b Find the transfer function of the code, $T(D)$.
c What is d_{free}, the minimum free distance of the code?

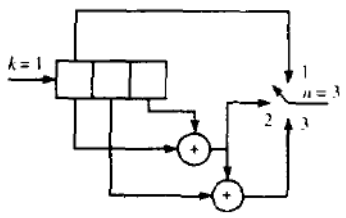


FIGURE P8-26

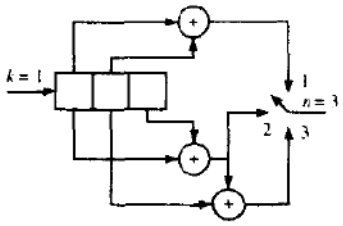


FIGURE P8-27

d Assume that a message has been encoded by this code and transmitted over a binary-symmetric channel with an error probability of $p = 10^{-5}$. If the received sequence is $\mathbf{r} = (110, 110, 110, 111, 010, 101, 101)$, using the Viterbi algorithm, find the transmitted bit sequence.

e Find an upper bound to the bit error probability of the code when the above binary-symmetric channel is employed. Make any reasonable approximation.

8-27 The block diagram of a (3, 1) convolutional code is shown in Fig. P8-27.

a Draw the state diagram of the code.

b Find the transfer function $T(D)$ of the code.

c Find the minimum free distance (d_{free}) of the code and show the corresponding path (at distance d_{free} from the all-zero code word) on the trellis.

d Assume that four information bits (x_1, x_2, x_3, x_4), followed by two zero bits, have been encoded and sent via a binary-symmetric channel with crossover probability equal to 0.1. The received sequence is (111, 111, 111, 111, 111, 111). Use the Viterbi decoding algorithm to find the most likely data sequence.

8-28 In the convolutional code generated by the encoder shown in Fig. P8-28.

a Find the transfer function of the code in the form $T(N, D)$.

b Find d_{free} of the code.

c If the code is used on a channel using hard-decision Viterbi decoding, assuming the crossover probability of the channel is $p = 10^{-6}$, use the hard-decision bound to find an upper bound on the average bit error probability of the code.

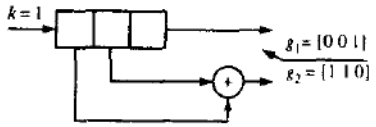


FIGURE P8-28

8-29 Figure P8-29 depicts a rate 1/2, constraint length $K = 2$, convolutional code.

a Sketch the tree diagram, the trellis diagram, and the state diagram.

b Solve for the transfer function $T(D, N, J)$ and, from this, specify the minimum free distance.

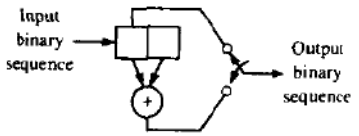


FIGURE P8-29

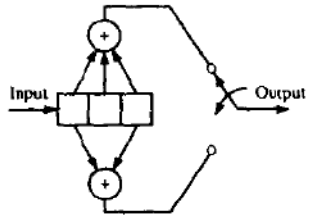


FIGURE P8-30

8-30 A rate 1/2, $K = 3$, binary convolutional encoder is shown in Fig. P8-30.
- **a** Draw the tree diagram, the trellis diagram, and the state diagram.
- **b** Determine the transfer function $T(D, N, J)$ and, from this, specify the minimum free distance.

8-31 Sketch the convolutional encoders for the following codes:
- **a** rate 1/2, $K = 5$, maximum free distance code (Table 8-2-1);
- **b** rate 1/3, $K = 5$, maximum free distance code (Table 8-2-2);
- **c** rate 2/3, $K = 2$, maximum free distance code (Table 8-2-8).

8-32 Draw the state diagram for the rate 2/3, $K = 2$, convolutional code indicated in Problem 8-31(c) and, for each transition, show the output sequence and the distance of the output sequence from the all-zero sequence.

8-33 Consider the $K = 3$, rate 1/2, convolutional code shown in Fig. P8-30. Suppose that the code is used on a binary symmetric channel and the received sequence for the first eight branches is 0001100000001001. Trace the decisions on a trellis diagram and label the survivors' Hamming distance metric at each node level. If a tie occurs in the metrics required for a decision, always choose the upper path (arbitrary choice).

8-34 Use the transfer function derived in Problem 8-30 for the $R_c = 1/2$, $K = 3$, convolutional code to compute the probability of a bit error for an AWGN channel with (a) hard-decision and (b) soft-decision decoding. Compare the performance by plotting the results of the computation on the same graph.

8-35 Use the generators given by (8-2-36) to obtain the encoder for a dual-3, rate 1/2 convolutional code. Determine the state diagram and derive the transfer function $T(D, N, J)$.

8-36 Draw the state diagram for the convolutional code generated by the encoder shown in Fig. P8-36 and, thus, determine if the code is catastrophic or noncatastrophic. Also, give an example of a rate 1/2, $K = 4$, convolutional encoder that exhibits catastrophic error propagation.

8-37 A trellis coded signal is formed as shown in Fig. P8-37 by encoding one bit by use

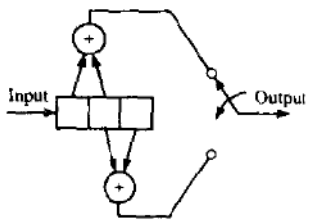


FIGURE P8-36

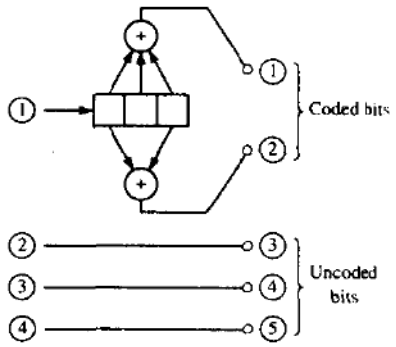


FIGURE P8-37

of a rate 1/2 convolutional code, while three additional information bits are left uncoded. Perform the set partitioning of a 32-QAM (cross) constellation and indicate the subsets in the partition. By how much is the distance between adjacent signal points increased as a result of partitioning?

8-38 Let $\mathbf{x}_1$ and $\mathbf{x}_2$ be two code words of length n with distance d and assume that these two code words are transmitted via a binary-symmetric channel with crossover probability p. Let $P(d)$ denote the error probability in transmission of these two code words.

a Show that

$$P(d) \leq \sum_{i=1}^{2^n} \sqrt{p(\mathbf{y}_i \mid \mathbf{x}_1) p(\mathbf{y}_i \mid \mathbf{x}_2)}$$

where the summation is over all binary sequences $\mathbf{y}_i$.

b From the above, conclude that

$$P(d) \leq [4p(1-p)]^{d/2}$$

9

SIGNAL DESIGN FOR BAND-LIMITED CHANNELS

In previous chapters, we considered the transmission of digital information through an additive gaussian noise channel. In effect, no bandwidth constraint was imposed on the signal design and the communication system design.

In this chapter, we consider the problem of signal design when the channel is band-limited to some specified bandwidth W Hz. Under this condition, the channel may be modeled as a linear filter having an equivalent lowpass frequency response $C(f)$ that is zero for $|f| > W$.

The first topic that is treated is the design of the signal pulse $g(t)$ in a linearly modulated signal, represented as

$$v(t) = \sum_n I_n g(t - nT)$$

that efficiently utilizes the total available channel bandwidth W. We shall see that when the channel is ideal for $|f| \leq W$, a signal pulse can be designed that allows us to transmit at symbol rates comparable to or exceeding the channel bandwidth W. On the other hand, when the channel is not ideal, signal transmission at a symbol rate equal to or exceeding W results in intersymbol interference (ISI) among a number of adjacent symbols.

The second topic that is treated in this chapter is the use of coding to shape the spectrum of the transmitted signal and, thus, to avoid the problem of ISI.

We begin our discussion with a general characterization of band-limited, linear filter channels.

9-1 CHARACTERIZATION OF BAND-LIMITED CHANNELS

Of the various channels available for digital communications, telephone channels are by far the most widely used. Such channels are characterized as

band-limited linear filters. This is certainly the proper characterization when frequency-division multiplexing (FDM) is used as a means for establishing channels in the telephone network. Recent additions to the telephone network employ pulse-code modulation (PCM) for digitizing and encoding the analog signal and time-division multiplexing (TDM) for establishing multiple channels. Nevertheless, filtering is still used on the analog signal prior to sampling and encoding. Consequently, even though the present telephone network employs a mixture of FDM and TDM for transmission, the linear filter model for telephone channels is still appropriate.

For our purposes, a band-limited channel such as a telephone channel will be characterized as a linear filter having an equivalent lowpass frequency response characteristic $C(f)$. Its equivalent lowpass impulse response is denoted by $c(t)$. Then, if a signal of the form

$$s(t) = \text{Re}\left[v(t)e^{j2\pi f_c t}\right] \tag{9-1-1}$$

is transmitted over a bandpass telephone channel, the equivalent lowpass received signal is

$$r_l(t) = \int_{-\infty}^{\infty} v(\tau)c(t-\tau)\,d\tau + z(t) \tag{9-1-2}$$

where the integral represents the convolution of $c(t)$ with $v(t)$, and $z(t)$ denotes the additive noise. Alternatively, the signal term can be represented in the frequency domain as $V(f)C(f)$, where $V(f)$ is the Fourier transform of $v(t)$.

If the channel is band-limited to W Hz then $C(f) = 0$ for $|f| > W$. As a consequence, any frequency components in $V(f)$ above $|f| = W$ will not be passed by the channel. For this reason, we limit the bandwidth of the transmitted signal to W Hz also.

Within the bandwidth of the channel, we may express the frequency response $C(f)$ as

$$C(f) = |C(f)|\,e^{j\theta(f)} \tag{9-1-3}$$

where $|C(f)|$ is the amplitude response characteristic and $\theta(f)$ is the phase response characteristic. Furthermore, the envelope delay characteristic is defined as

$$\tau(f) = -\frac{1}{2\pi}\frac{d\theta(f)}{df} \tag{9-1-4}$$

A channel is said to be *nondistorting* or *ideal* if the amplitude response $|C(f)|$ is constant for all $|f| \leq W$ and $\theta(f)$ is a linear function of frequency, i.e., $\tau(f)$ is a constant for all $|f| \leq W$. On the other hand, if $|C(f)|$ is not constant for all $|f| \leq W$, we say that the channel *distorts the transmitted signal* $V(f)$ *in amplitude*, and, if $\tau(f)$ is not constant for all $|f| \leq W$, we say that the channel *distorts the signal* $V(f)$ *in delay*.

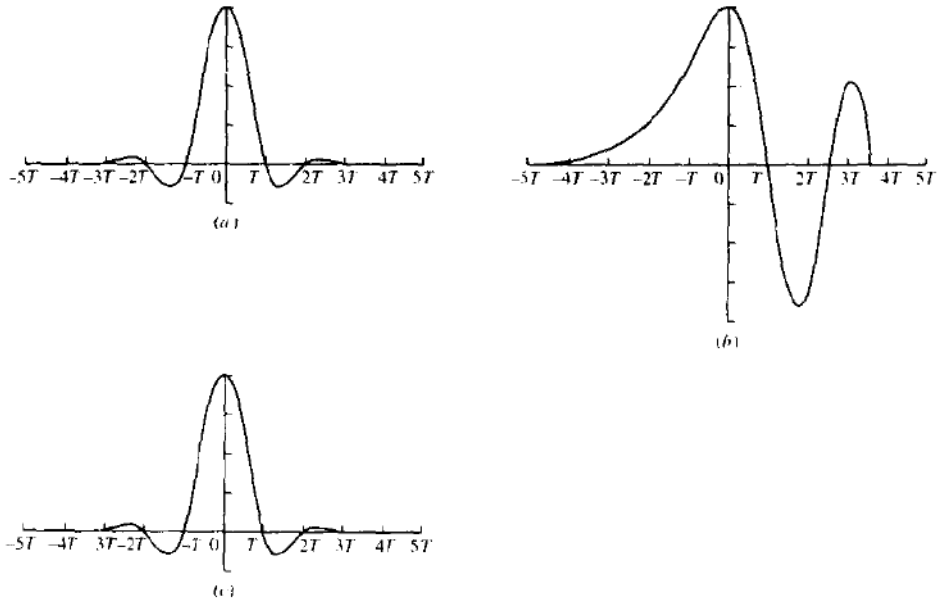


FIGURE 9-1-1 Effect of channel distortion: (*a*) channel input; (*b*) channel output; (*c*) equalizer output.

As a result of the amplitude and delay distortion caused by the nonideal channel frequency response characteristic $C(f)$, a succession of pulses transmitted through the channel at rates comparable to the bandwidth W are smeared to the point that they are no longer distinguishable as well-defined pulses at the receiving terminal. Instead, they overlap and, thus, we have intersymbol interference. As an example of the effect of delay distortion on a transmitted pulse, Fig. 9-1-1(*a*) illustrates a band-limited pulse having zeros periodically spaced in time at points labeled $\pm T$, $\pm 2T$, etc. If information is conveyed by the pulse amplitude, as in PAM, for example, then one can transmit a sequence of pulses, each of which has a peak at the periodic zeros of the other pulses. However, transmission of the pulse through a channel modeled as having a linear envelope delay characteristic $\tau(f)$ [quadratic phase $\theta(f)$] results in the received pulse shown in Fig. 9-1-1(*b*) having zero-crossings that are no longer periodically spaced. Consequently, a sequence of successive pulses would be smeared into one another and the peaks of the pulses would no longer be distinguishable. Thus, the channel delay distortion results in intersymbol interference. As will be discussed in Chapter 10, it is possible to compensate for the nonideal frequency response characteristic of the channel by use of a filter or equalizer at the demodulator. Figure 9-1-1(*c*) illustrates the output of a linear equalizer that compensates for the linear distortion in the channel.

The extent of the intersymbol interference on a telephone channel can be

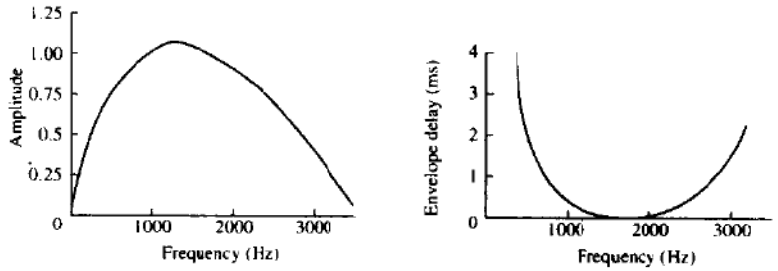


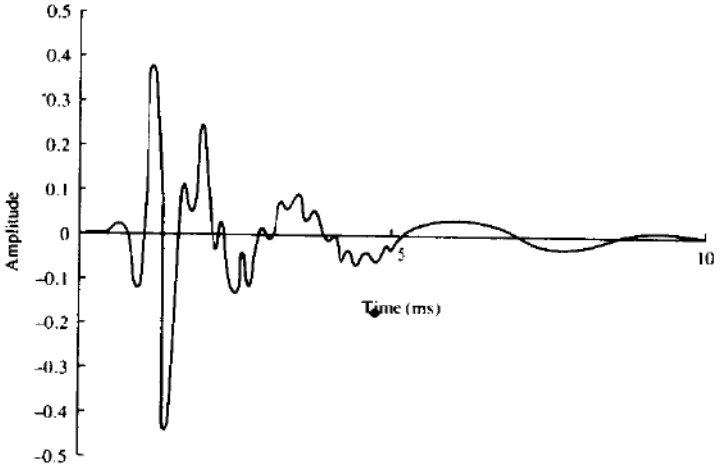
FIGURE 9-1-2 Average amplitude and delay characteristics of medium-range telephone channel.

appreciated by observing a frequency response characteristic of the channel. Figure 9-1-2 illustrates the measured average amplitude and delay as functions of frequency for a medium-range (180–725 mi) telephone channel of the switched telecommunications network as given by Duffy and Tratcher (1971). We observe that the usable band of the channel extends from about 300 Hz to about 3000 Hz. The corresponding impulse response of this average channel is shown in Fig. 9-1-3. Its duration is about 10 ms. In comparison, the transmitted symbol rates on such a channel may be of the order of 2500 pulses or symbols per second. Hence, intersymbol interference might extend over 20–30 symbols.

In addition to linear distortion, signals transmitted through telephone channels are subject to other impairments, specifically nonlinear distortion, frequency offset, phase jitter, impulse noise and thermal noise.

Nonlinear distortion in telephone channels arises from nonlinearities in

FIGURE 9-1-3 Impulse response of average channel with amplitude and delay shown in Fig. 9-1-2.



amplifiers and compandors used in the telephone system. This type of distortion is usually small and it is very difficult to correct.

A small *frequency offset*, usually less than 5 Hz, results from the use of carrier equipment in the telephone channel. Such an offset cannot be tolerated in high-speed digital transmission systems that use synchronous phase-coherent demodulation. The offset is usually compensated for by the carrier recovery loop in the demodulator.

Phase jitter is basically a low-index frequency modulation of the transmitted signal with the low frequency harmonics of the power line frequency (50–60 Hz). Phase jitter poses a serious problem in digital transmission of high rates. However, it can be tracked and compensated for, to some extent, at the demodulator.

Impulse noise is an additive disturbance. It arises primarily from the switching equipment in the telephone system. *Thermal* (gaussian) *noise* is also present at levels of 20–30 dB below the signal.

The degree to which one must be concerned with these channel impairments depends on the transmission rate over the channel and the modulation technique. For rates below 1800 bits/s ($R/W < 1$), one can choose a modulation technique, e.g., FSK, that is relatively insensitive to the amount of distortion encountered on typical telephone channels from all the sources listed above. For rates between 1800 and 2400 bits/s ($R/W \approx 1$), a more bandwidth-efficient modulation technique such as four-phase PSK is usually employed. At these rates, some form of compromise equalization is often employed to compensate for the average amplitude and delay distortion in the channel. In addition, the carrier recovery method is designed to compensate for the frequency offset. The other channel impairments are not that serious in their effects on the error rate performance at these rates. At transmission rates above 2400 bits/s ($R/W > 1$), bandwidth-efficient coded modulation techniques such as trellis-coded QAM, PAM, and PSK are employed. For such rates, special attention must be paid to linear distortion, frequency offset, and phase jitter. Linear distortion is usually compensated for by means of an adaptive equalizer. Phase jitter is handled by a combination of signal design and some type of phase compensation at the demodulator. At rates above 9600 bits/s, special attention must be paid not only to linear distortion, phase jitter, and frequency offset, but also to the other channel impairments mentioned above.

Unfortunately, a channel model that encompasses all the impairments listed above becomes difficult to analyze. For mathematical tractability the channel model that is adopted in this and the next two chapters is a linear filter that introduces amplitude and delay distortion and adds gaussian noise.

Besides the telephone channels, there are other physical channels that exhibit some form of time dispersion, and thus, introduce intersymbol interference. Radio channels such as shortwave ionospheric propagation (HF) and tropospheric scatter are two examples of time-dispersive channels. In these channels, time dispersion and, hence, intersymbol interference is the result of multiple propagation paths with different path delays. The number of paths

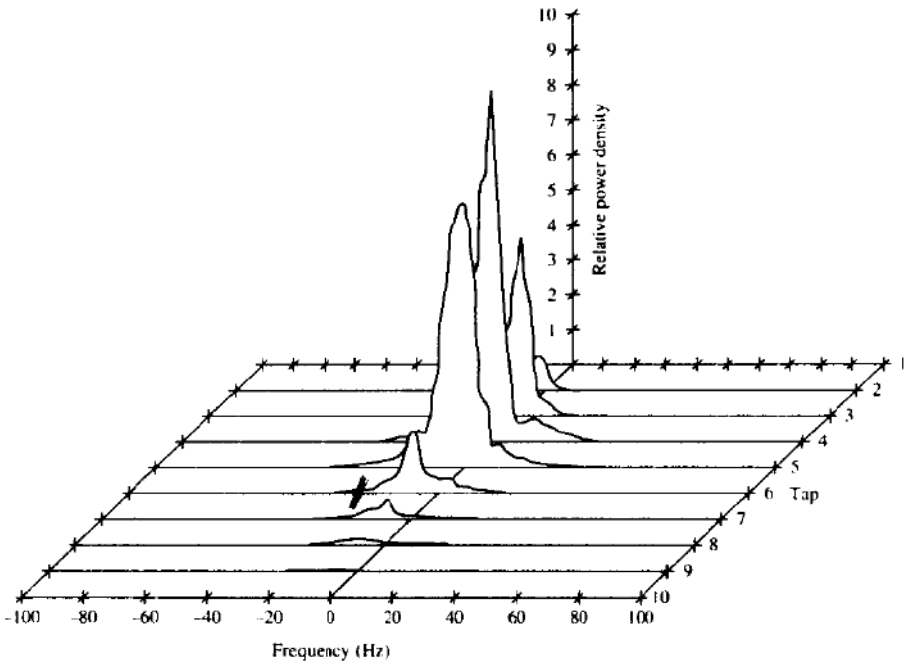


FIGURE 9-1-4 Scattering function of a medium-range tropospheric scatter channel.

and the relative time delays among the paths vary with time, and, for this reason, these radio channels are usually called *time-variant multipath channels*. The time-variant multipath conditions give rise to a wide variety of frequency response characteristics. Consequently the frequency response characterization that is used for telephone channels is inappropriate for time-variant multipath channels. Instead, these radio channels are characterized statistically, as explained in more detail in Chapter 14, in terms of the scattering function, which, in brief, is a two-dimensional representation of the average received signal power as a function of relative time delay and Doppler frequency.

For illustrative purposes, a scattering function measured on a medium-range (150 mi) trophospheric scatter channel is shown in Fig. 9-1-4. The total time duration (multipath spread) of the channel response is approximately 0.7 μs on the average, and the spread between "half-power points" in Doppler frequency is a little less than 1 Hz on the strongest path and somewhat larger on the other paths. Typically, if one is transmitting at a rate of 10^7 symbols/s over such a channel, the multipath spread of 0.7 μs will result in intersymbol interference that spans about seven symbols.

In this chapter, we deal exclusively with the linear time-invariant filter model for a band-limited channel. The adaptive equalization techniques presented in Chapters 10 and 11 for combating intersymbol interference are also applicable to time-invariant multipath channels, under the condition that

the time variations in the channel are relatively slow in comparison to the total channel bandwidth or, equivalently, to the symbol transmission rate over the channel.

9-2 SIGNAL DESIGN FOR BAND-LIMITED CHANNELS

It was shown in Chapter 4 that the equivalent lowpass transmitted signal for several different types of digital modulation techniques has the common form

$$v(t) = \sum_{n=0}^{\infty} I_n g(t - nT) \tag{9-2-1}$$

where $\{I_n\}$ represents the discrete information-bearing sequence of symbols and $g(t)$ is a pulse that, for the purposes of this discussion, is assumed to have a band-limited frequency response characteristic $G(f)$, i.e., $G(f) = 0$ for $|f| > W$. This signal is transmitted over a channel having a frequency response $C(f)$, also limited to $|f| \leq W$. Consequently, the received signal can be represented as

$$r_l(t) = \sum_{n=0}^{\infty} I_n h(t - nT) + z(t) \tag{9-2-2}$$

where

$$h(t) = \int_{-\infty}^{\infty} g(\tau)c(t - \tau)\, d\tau \tag{9-2-3}$$

and $z(t)$ represents the additive white Gaussian noise.

Let us suppose that the received signal is passed first through a filter and then sampled at a rate $1/T$ samples/s. We shall show in a subsequent section that the optimum filter from the point of view of signal detection is one matched to the received pulse. That is, the frequency response of the receiving filter is $H^*(f)$. We denote the output of the receiving filter as

$$y(t) = \sum_{n=0}^{\infty} I_n x(t - nT) + \nu(t) \tag{9-2-4}$$

where $x(t)$ is the pulse representing the response of the receiving filter to the input pulse $h(t)$ and $\nu(t)$ is the response of the receiving filter to the noise $z(t)$.

Now, if $y(t)$ is sampled at times $t = kT + \tau_0$, $k = 0, 1, \ldots$, we have

$$y(kT + \tau_0) \equiv y_k = \sum_{n=0}^{\infty} I_n x(kT - nT + \tau_0) + \nu(kT + \tau_0) \tag{9-2-5}$$

or, equivalently,

$$y_k = \sum_{n=0}^{\infty} I_n x_{k-n} + \nu_k, \qquad k = 0, 1, \ldots \tag{9-2-6}$$

where τ_0 is the transmission delay through the channel. The sample values can be expressed as

$$y_k = x_0\left(I_k + \frac{1}{x_0}\sum_{\substack{n=0\\ n\neq k}}^{\infty} I_n x_{k-n}\right) + \nu_k, \qquad k = 0, 1, \ldots \tag{9-2-7}$$

We regard x_0 as an arbitrary scale factor, which we arbitrarily set equal to unity for convenience. Then

$$y_k = I_k + \sum_{\substack{n=0\\ n\neq k}}^{\infty} I_n x_{k-n} + \nu_k \tag{9-2-8}$$

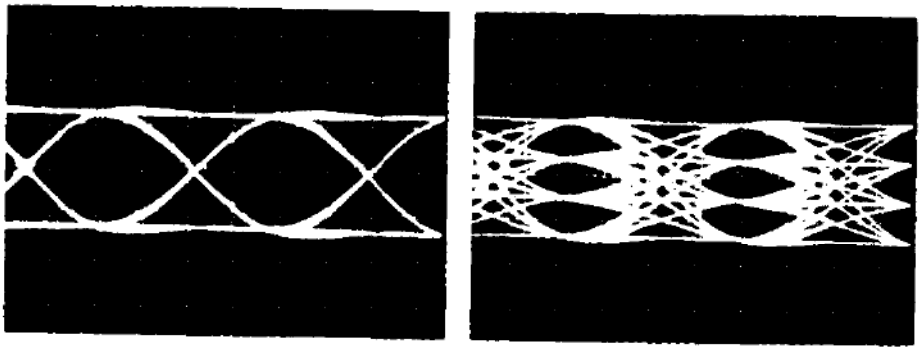
The term I_k represents the desired information symbol at the k th sampling instant, the term

$$\sum_{\substack{n=0\\ n\neq k}}^{\infty} I_n x_{k-n}$$

represents the intersymbol interference (ISI), and ν_k is the additive gaussian noise variable at the kth sampling instant.

The amount of intersymbol interference and noise in a digital communications system can be viewed on an oscilloscope. For PAM signals, we can display the received signal $y(t)$ on the vertical input with the horizontal sweep rate set at $1/T$. The resulting oscilloscope display is called an *eye pattern* because of its resemblance to the human eye. For example, Fig. 9-2-1 illustrates the eye patterns for binary and four-level PAM modulation. The effect of ISI is to cause the eye to close, thereby reducing the margin for additive noise to cause errors. Figure 9-2-2 graphically illustrates the effect of intersymbol interference in reducing the opening of a binary eye. Note that intersymbol interference distorts the position of the zero-crossings and causes

FIGURE 9-2-1 Examples of eye patterns for binary and quaternary amplitude shift keying (or PAM).



BINARY QUATERNARY

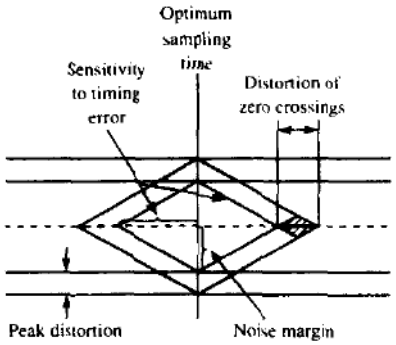


FIGURE 9-2-2 Effect of intersymbol interference on eye opening.

a reduction in the eye opening. Thus, it causes the system to be more sensitive to a synchronization error.

For PSK and QAM it is customary to display the "eye pattern" as a two-dimensional scatter diagram illustrating the sampled values $\{y_k\}$ that represent the decision variables at the sampling instants. Figure 9-2-3 illustrates such an eye pattern for an 8-PSK signal. In the absence of intersymbol interference and noise, the superimposed signals at the sampling instants would result in eight distinct points corresponding to the eight transmitted signal phases. Intersymbol interference and noise result in a deviation of the received samples $\{y_k\}$ from the desired 8-PSK signal. The larger the intersymbol interference and noise, the larger the scattering of the received signal samples relative to the transmitted signal points.

Below, we consider the problem of signal design under the condition that there is no intersymbol interference at the sampling instants.

9-2-1 DESIGN OF BAND-LIMITED SIGNALS FOR NO INTERSYMBOL INTERFERENCE—THE NYQUIST CRITERION

For the discussion in this section and in Section 9-2-2, we assume that the band-limited channel has ideal frequency response characteristics, i.e., $C(f) = 1$

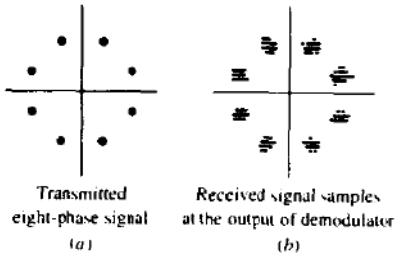


FIGURE 9-2-3 Two-dimensional digital "eye patterns."

for $|f| \leq W$. Then the pulse $x(t)$ has a spectral characteristic $X(f) = |G(f)|^2$, where

$$x(t) = \int_{W}^{W} X(f)e^{j2\pi ft}\,df \tag{9-2-9}$$

We are interested in determining the spectral properties of the pulse $x(t)$ and, hence, the transmitted pulse $g(t)$, that results in no intersymbol interference. Since

$$y_k = I_k + \sum_{\substack{n=0 \\ n\neq k}}^{\infty} I_n x_{k-n} + v_k \tag{9-2-10}$$

the condition for no intersymbol interference is

$$x(t = kT) \equiv x_k = \begin{cases} 1 & (k = 0) \\ 0 & (k \neq 0) \end{cases} \tag{9-2-11}$$

Below, we derive the necessary and sufficient condition on $X(f)$ in order for $x(t)$ to satisfy the above relation. This condition is known as the *Nyquist pulse-shaping criterion* or *Nyquist condition for zero ISI* and is stated in the following theorem.

Theorem (Nyquist)

The necessary and sufficient condition for $x(t)$ to satisfy

$$x(nT) = \begin{cases} 1 & (n = 0) \\ 0 & (n \neq 0) \end{cases} \tag{9-2-12}$$

is that its Fourier transform $X(f)$ satisfy

$$\sum_{m=-\infty}^{\infty} X(f + m/T) = T \tag{9-2-13}$$

Proof

In general, $x(t)$ is the inverse Fourier transform of $X(f)$. Hence,

$$x(t) = \int_{-\infty}^{\infty} X(f)e^{j2\pi ft}\,df \tag{9-2-14}$$

At the sampling instants $t = nT$, this relation becomes

$$x(nT) = \int_{-\infty}^{\infty} X(f)e^{j2\pi fnT}\,df \tag{9-2-15}$$

Let us break up the integral in (9-2-15) into integrals covering the finite range of $1/T$. Thus, we obtain

$$\begin{aligned} x(nT) &= \sum_{m=-\infty}^{\infty} \int_{(2m-1)/2T}^{(2m+1)/2T} X(f)e^{j2\pi fnT}\,df \\ &= \sum_{m=-\infty}^{\infty} \int_{-1/2T}^{1/2T} X(f+m/T)e^{j2\pi fnT}\,dt \\ &= \int_{-1/2T}^{1/2T} \left[\sum_{m=-\infty}^{\infty} X(f+m/T)\right] e^{j2\pi fnT}\,df \\ &= \int_{-1/2T}^{1/2T} B(f)e^{j2\pi fnT}\,df \end{aligned} \tag{9-2-16}$$

where we have defined $B(f)$ as

$$B(f) = \sum_{m=-\infty}^{\infty} X(f+m/T) \tag{9-2-17}$$

Obviously $B(f)$ is a periodic function with period $1/T$, and, therefore, it can be expanded in terms of its Fourier series coefficients $\{b_n\}$ as

$$B(f) = \sum_{n=-\infty}^{\infty} b_n e^{j2\pi nfT} \tag{9-2-18}$$

where

$$b_n = T\int_{-1/2T}^{1/2T} B(f)e^{-j2\pi nfT}\,df \tag{9-2-19}$$

Comparing (9-2-19) and (9-2-16), we obtain

$$b_n = Tx(-nT) \tag{9-2-20}$$

Therefore, the necessary and sufficient condition for (9-2-10) to be satisfied is that

$$b_n = \begin{cases} T & (n=0) \\ 0 & (n\neq 0) \end{cases} \tag{9-2-21}$$

which, when substituted into (9-2-18), yields

$$B(f) = T \tag{9-2-22}$$

or, equivalently,

$$\sum_{m=-\infty}^{\infty} X(f+m/T) = T \tag{9-2-23}$$

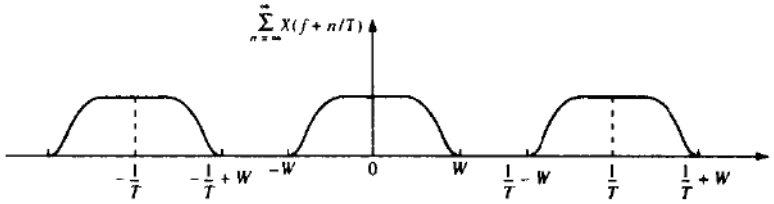


FIGURE 9-2-4 Plot of $B(f)$ for the case $T < 1/2W$.

This concludes the proof of the theorem.

Now suppose that the channel has a bandwidth of W. Then $C(f) \equiv 0$ for $|f| > W$ and, consequently, $X(f) = 0$ for $|f| > W$. We distinguish three cases.

1 When $T < 1/2W$, or, equivalently, $1/T > 2W$, since $B(f) = \sum_{n=-\infty}^{+\infty} X(f + n/T)$ consists of nonoverlapping replicas of $X(f)$, separated by $1/T$ as shown in Fig. 9-2-4, there is no choice for $X(f)$ to ensure $B(f) \equiv T$ in this case and there is no way that we can design a system with no ISI.

2 When $T = 1/2W$, or, equivalently, $1/T = 2W$ (the Nyquist rate), the replications of $X(f)$, separated by $1/T$, are as shown in Fig. 9-2-5. It is clear that in this case there exists only one $X(f)$ that results in $B(f) = T$, namely,

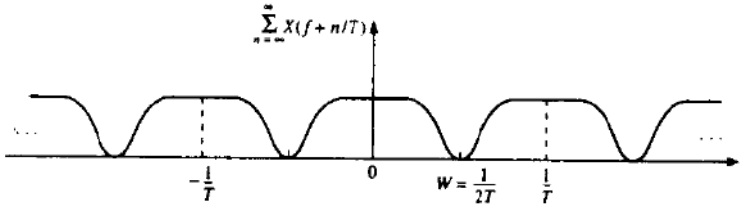
$$X(f) = \begin{cases} T & (|f| < W) \\ 0 & (\text{otherwise}) \end{cases} \tag{9-2-24}$$

which corresponds to the pulse

$$x(t) = \frac{\sin(\pi t/T)}{\pi t/T} \equiv \operatorname{sinc}\left(\frac{\pi t}{T}\right) \tag{9-2-25}$$

This means that the smallest value of T for which transmission with zero ISI is possible is $T = 1/2W$, and for this value, $x(t)$ has to be a sinc function. The difficulty with this choice of $x(t)$ is that it is noncausal and therefore nonrealizable. To make it realizable, usually a delayed version of it, i.e., $\operatorname{sinc}[\pi(t - t_0)/T]$ is used and t_0 is chosen such that for $t < 0$, we have $\operatorname{sinc}[\pi(t - t_0)/T] \approx 0$. Of course, with this choice of $x(t)$, the sampling time

FIGURE 9-2-5 Plot of $B(f)$ for the case $T = 1/2W$.



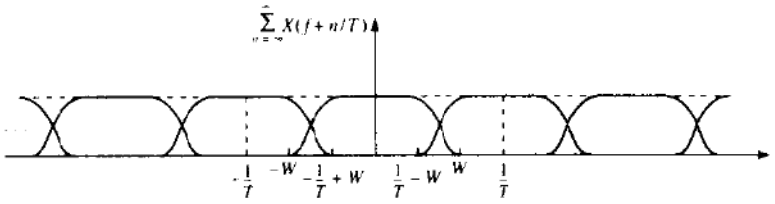


FIGURE 9-2-6 Plot of $B(f)$ for the case $T > 1/2W$.

must also be shifted to $mT + t_0$. A second difficulty with this pulse shape is that its rate of convergence to zero is slow. The tails of $x(t)$ decay as $1/t$; consequently, a small mistiming error in sampling the output of the matched filter at the demodulator results in an infinite series of ISI components. Such a series is not absolutely summable because of the $1/t$ rate of decay of the pulse, and, hence, the sum of the resulting ISI does not converge.

3 When $T > 1/2W$, $B(f)$ consists of overlapping replications of $X(f)$ separated by $1/T$, as shown in Fig. 9-2-6. In this case, there exist numerous choices for $X(f)$ such that $B(f) \equiv T$.

A particular pulse spectrum, for the $T > 1/2W$ case, that has desirable spectral properties and has been widely used in practice is the raised cosine spectrum. The raised cosine frequency characteristic is given as (see Problem 9-11)

$$X_{rc}(f) = \begin{cases} T & \left(0 \leq |f| \leq \dfrac{1-\beta}{2T}\right) \\ \dfrac{T}{2}\left\{1 + \cos\left[\dfrac{\pi T}{\beta}\left(|f| - \dfrac{1-\beta}{2T}\right)\right]\right\} & \left(\dfrac{1-\beta}{2T} \leq |f| \leq \dfrac{1+\beta}{2T}\right) \\ 0 & \left(|f| > \dfrac{1+\beta}{2T}\right) \end{cases} \tag{9-2-26}$$

where β is called the *rolloff factor*, and takes values in the range $0 \leq \beta \leq 1$. The bandwidth occupied by the signal beyond the Nyquist frequency $1/2T$ is called the *excess bandwidth* and is usually expressed as a percentage of the Nyquist frequency. For example, when $\beta = \frac{1}{2}$, the excess bandwidth is 50%, and when $\beta = 1$, the excess bandwidth is 100%. The pulse $x(t)$, having the raised cosine spectrum, is

$$\begin{aligned} x(t) &= \frac{\sin(\pi t/T)}{\pi t/T} \frac{\cos(\pi\beta t/T)}{1 - 4\beta^2 t^2/T^2} \\ &= \operatorname{sinc}(\pi t/T) \frac{\cos(\pi\beta t/T)}{1 - 4\beta^2 t^2/T^2} \end{aligned} \tag{9-2-27}$$

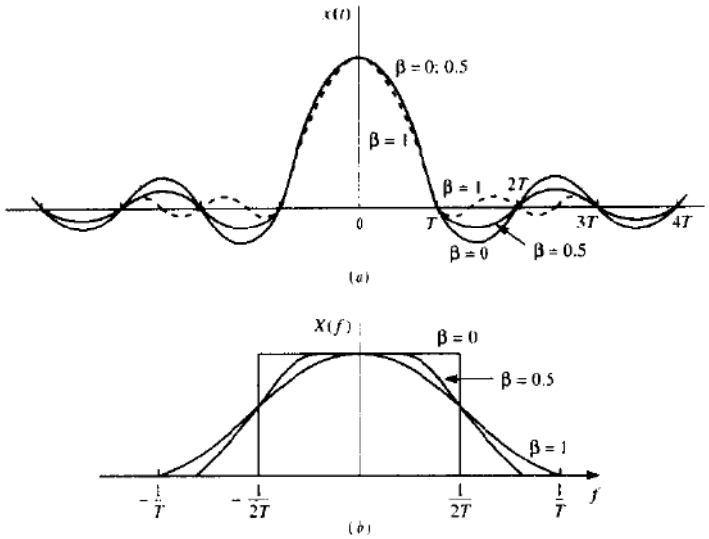


FIGURE 9-2-7 Pulses having a raised cosine spectrum.

Note that $x(t)$ is normalized so that $x(0) = 1$. Figure 9-2-7 illustrates the raised cosine spectral characteristics and the corresponding pulses for $\beta = 0$, $\frac{1}{2}$ and 1. Note that for $\beta = 0$, the pulse reduces to $x(t) = \text{sinc}(\pi t/T)$, and the symbol rate $1/T = 2W$. When $\beta = 1$, the symbol rate is $1/T = W$. In general, the tails of $x(t)$ decay as $1/t^3$ for $\beta > 0$. Consequently, a mistiming error in sampling leads to a series of ISI components that converges to a finite value.

Due to the smooth characteristics of the raised cosine spectrum, it is possible to design practical filters for the transmitter and the receiver that approximate the overall desired frequency response. In the special case where the channel is ideal, i.e., $C(f) = 1$, $|f| \leq W$, we have

$$X_{rc}(f) = G_T(f)G_R(f), \tag{9-2-28}$$

where $G_T(f)$ and $G_R(f)$ are the frequency responses of the two filters. In this case, if the receiver filter is matched to the transmitter filter, we have $X_{rc}(f) = G_T(f)G_R(f) = |G_T(f)|^2$. Ideally,

$$G_T(f) = \sqrt{|X_{rc}(f)|}\, e^{-j2\pi f t_0} \tag{9-2-29}$$

and $G_R(f) = G_T^*(f)$, where t_0 is some nominal delay that is required to ensure physical realizability of the filter. Thus, the overall raised cosine spectral characteristic is split evenly between the transmitting filter and the receiving filter. Note also that an additional delay is necessary to ensure the physical realizability of the receiving filter.

9-2-2 Design of Band-Limited Signals with Controlled ISI—Partial-Response Signals

As we have observed from our discussion of signal design for zero ISI, it is necessary to reduce the symbol rate $1/T$ below the Nyquist rate of $2W$ symbols/s to realize practical transmitting and receiving filters. On the other hand, suppose we choose to relax the condition of zero ISI and, thus, achieve a symbol transmission rate of $2W$ symbols/s. By allowing for a controlled amount of ISI, we can achieve this symbol rate.

We have already seen that the condition for zero ISI is $x(nT) = 0$ for $n \neq 0$. However, suppose that we design the band-limited signal to have controlled ISI at one time instant. This means that we allow one additional nonzero value in the samples $\{x(nT)\}$. The ISI that we introduce is deterministic or "controlled" and, hence, it can be taken into account at the receiver, as discussed below.

One special case that leads to (approximately), physically realizable transmitting and receiving filters is specified by the samples†

$$x(nT) = \begin{cases} 1 & (n = 0, 1) \\ 0 & (\text{otherwise}) \end{cases} \tag{9-2-30}$$

Now, using (9-2-20), we obtain

$$b_n = \begin{cases} T & (n = 0, -1) \\ 0 & (\text{otherwise}) \end{cases} \tag{9-2-31}$$

which, when substituted into (9-2-18), yields

$$B(f) = T + Te^{\,j2\pi fT} \tag{9-2-32}$$

As in the preceding section, it is impossible to satisfy the above equation for $T < 1/2W$. However, for $T = 1/2W$, we obtain

$$\begin{aligned} X(f) &= \begin{cases} \dfrac{1}{2W}(1 + e^{-j\pi f/W}) & (|f| < W) \\ 0 & (\text{otherwise}) \end{cases} \\ &= \begin{cases} \dfrac{1}{W} e^{-j\pi f/2W} \cos \dfrac{\pi f}{2W} & (|f| < W) \\ 0 & (\text{otherwise}) \end{cases} \end{aligned} \tag{9-2-33}$$

Therefore, $x(t)$ is given by

$$x(t) = \operatorname{sinc}(2\pi Wt) + \operatorname{sinc}[2\pi(Wt - \tfrac{1}{2})] \tag{9-2-34}$$

This pulse is called a *duobinary signal pulse.* It is illustrated along with its

†It is convenient to deal with samples of $x(t)$ that are normalized to unity for $n = 0, 1$.

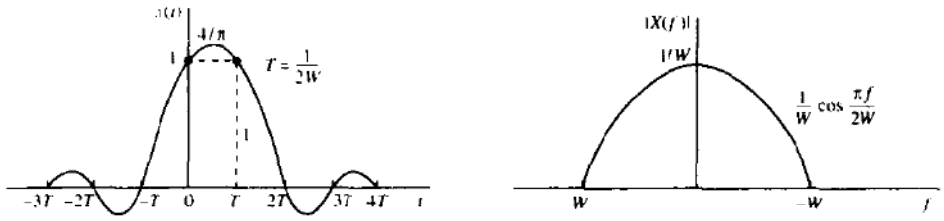


FIGURE 9-2-8 Time domain and frequency domain characteristics of a duobinary signal.

magnitude spectrum in Fig. 9-2-8. Note that the spectrum decays to zero smoothly, which means that physically realizable filters can be designed that approximate this spectrum very closely. Thus, a symbol rate of $2W$ is achieved.

Another special case that leads to (approximately) physically realizable transmitting and receiving filters is specified by the samples

$$x\left(\frac{n}{2W}\right) = x(nT) = \begin{cases} 1 & (n = -1) \\ -1 & (n = 1) \\ 0 & (\text{otherwise}) \end{cases} \tag{9-2-35}$$

The corresponding pulse $x(t)$ is given as

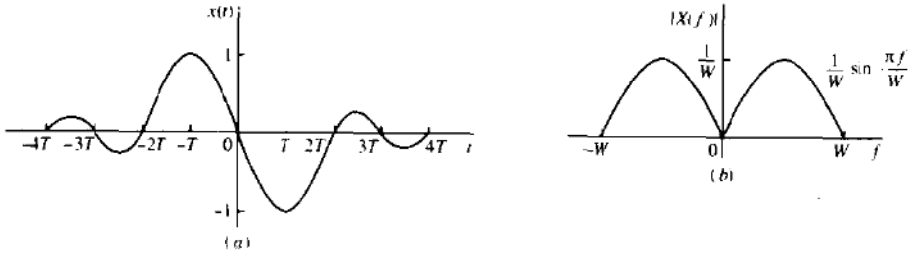
$$x(t) = \text{sinc}\left[\frac{\pi(t+T)}{T}\right] - \text{sinc}\left[\frac{\pi(t-T)}{T}\right] \tag{9-2-36}$$

and its spectrum is

$$X(f) = \begin{cases} \dfrac{1}{2W}\left(e^{j\pi f/W} - e^{-j\pi f/W}\right) = \dfrac{j}{W}\sin\dfrac{\pi f}{W} & |f| \leq W \\ 0 & |f| > W \end{cases} \tag{9-2-37}$$

This pulse and its magnitude spectrum are illustrated in Fig. 9-2-9. It is called a *modified duobinary signal pulse*. It is interesting to note that the spectrum of

FIGURE 9-2-9 Time domain and frequency domain characteristics of a modified duobinary signal.



this signal has a zero at $f = 0$, making it suitable for transmission over a channel that does not pass d.c.

One can obtain other interesting and physically realizable filter characteristics, as shown by Kretzmer (1966) and Lucky *et al.* (1968), by selecting different values for the samples $\{x(n/2W)\}$ and more than two nonzero samples. However, as we select more nonzero samples, the problem of unraveling the controlled ISI becomes more cumbersome and impractical.

In general, the class of bandlimited signals pulses that have the form

$$x(t) = \sum_{n=-\infty}^{\infty} x\left(\frac{n}{2W}\right) \operatorname{sinc}\left[2\pi W\left(t - \frac{n}{2W}\right)\right] \tag{9-2-38}$$

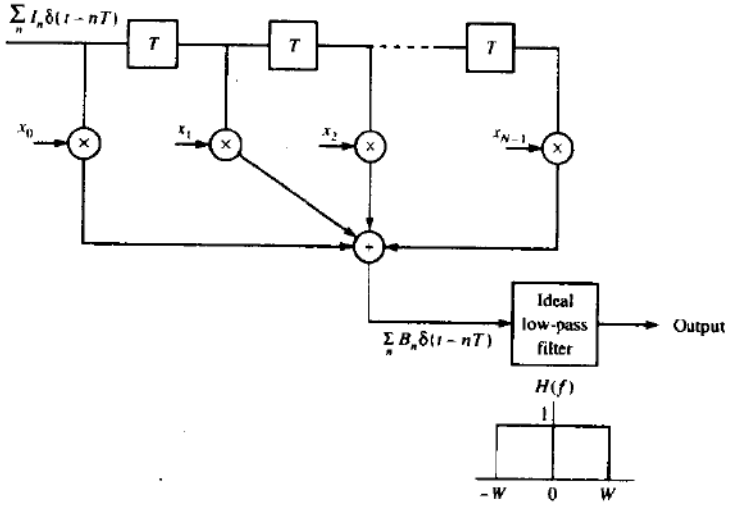
and their corresponding spectra

$$X(f) = \begin{cases} \dfrac{1}{2W} \displaystyle\sum_{n=-\infty}^{\infty} x\left(\frac{n}{2W}\right) e^{-jn\pi f/W} & (|f| \leq W) \\ 0 & (|f| \leq W) \end{cases} \tag{9-2-39}$$

are called *partial-response signals* when controlled ISI is purposely introduced by selecting two or more nonzero samples from the set $\{x(n/2W)\}$. The resulting signal pulses allow us to transmit information symbols at the Nyquist rate of $2W$ symbols/s. The detection of the received symbols in the presence of controlled ISI is described below.

Alternative Characterization of Partial-Response Signals We conclude this subsection by presenting another interpretation of a partial-response signal. Suppose that the partial-response signal is generated, as shown in Fig. 9-2-10, by passing the discrete-time sequence $\{I_n\}$ through a discrete-time filter

FIGURE 9-2-10 An alternative method for generating a partial-response signal.



with coefficients $x_n \equiv x(n/2W)$, $n = 0, 1, \ldots, N-1$, and using the output sequence $\{B_n\}$ from this filter to excite periodically with an input $B_n\delta(t - nT)$ an analog filter having an impulse response sinc $(2\pi Wt)$. The resulting output signal is identical to the partial-response signal given by (9-2-38).

Since

$$B_n = \sum_{k=0}^{N-1} x_k I_{n-k} \tag{9-2-40}$$

the sequence of symbols $\{B_n\}$ is correlated as a consequence of the filtering performed on the sequence $\{I_n\}$. In fact, the autocorrelation function of the sequence $\{B_n\}$ is

$$\begin{aligned}\phi(m) &= E(B_n B_{n+m}) \\ &= \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} x_k x_l E(I_{n-k} I_{n+m-l})\end{aligned} \tag{9-2-41}$$

When the input sequence is zero-mean and white,

$$E(I_{n-k} I_{n+m-l}) = \delta_{m+k-l} \tag{9-2-42}$$

where we have used the normalization $E(I_n^2) = 1$. Substitution of (9-2-42), into (9-2-41) yields the desired autocorrelation function for $\{B_n\}$ in the form

$$\phi(m) = \sum_{k=0}^{N-1-|m|} x_k x_{k+|m|}, \qquad m = 0, \pm 1, \ldots, \pm(N-1) \tag{9-2-43}$$

The corresponding power spectral density is

$$\begin{aligned}\Phi(f) &= \sum_{m=-(N-1)}^{N-1} \phi(m) e^{-j2\pi fmT} \\ &= \left| \sum_{m=0}^{N-1} x_m e^{-j2\pi fmT} \right|^2\end{aligned} \tag{9-2-44}$$

where $T = 1/2W$ and $|f| \leq 1/2T = W$.

9-2-3 Data Detection for Controlled ISI

In this section, we describe two methods for detecting the information symbols at the receiver when the received signal contains controlled ISI. One is a symbol-by-symbol detection method that is relatively easy to implement. The second method is based on the maximum-likelihood criterion for detecting a sequence of symbols. The latter method minimizes the probability of error but is a little more complex to implement. In particular, we consider the detection of the duobinary and the modified duobinary partial response signals. In both

cases, we assume that the desired spectral characteristic $X(f)$ for the partial response signal is split evenly between the transmitting and receiving filters, i.e., $|G_T(f)| = |G_R(f)| = |X(f)|^{1/2}$. This treatment is based on PAM signals, but it is easily generalized to QAM and PSK.

Symbol-by-Symbol Suboptimum Detection For the duobinary signal pulse, $x(nT) = 1$, for $n = 0, 1$, and zero otherwise. Hence, the samples at the output of the receiving filter (demodulator) have the form

$$y_m = B_m + \nu_m = I_m + I_{m-1} + \nu_m \tag{9-2-45}$$

where $\{I_m\}$ is the transmitted sequence of amplitudes and $\{\nu_m\}$ is a sequence of additive gaussian noise samples. Let us ignore the noise for the moment and consider the binary case where $I_m = \pm 1$ with equal probability. Then B_m takes on one of three possible values, namely, $B_m = -2, 0, 2$ with corresponding probabilities 1/4, 1/2, 1/4. If I_{m-1} is the detected symbol from the $(m-1)$ th signaling interval, its effect on B_m, the received signal in the m th signaling interval, can be eliminated by subtraction, thus allowing I_m to be detected. This process can be repeated sequentially for every received symbol.

The major problem with this procedure is that errors arising from the additive noise tend to propagate. For example, if I_{m-1} is in error, its effect on B_m is not eliminated but, in fact, it is reinforced by the incorrect subtraction. Consequently, the detection of B_m is also likely to be in error.

Error propagation can be avoided by *precoding* the data at the transmitter instead of eliminating the controlled ISI by subtraction at the receiver. The precoding is performed on the binary data sequence prior to modulation. From the data sequence $\{D_n\}$ of 1s and 0s that is to be transmitted, a new sequence $\{P_n\}$, called the *precoded sequence*, is generated. For the duobinary signal, the precoded sequence is defined as

$$P_m = D_m \ominus P_{m-1}, \qquad m = 1, 2, \ldots \tag{9-2-46}$$

where $\ominus$ denotes modulo-2 subtraction.† Then we set $I_m = -1$ if $P_m = 0$ and $I_m = 1$ if $P_m = 1$, i.e., $I_m = 2P_m - 1$. Note that this precoding operation is identical to that described in Section 4-3-2 in the context of our discussion of an NRZI signal.

The noise-free samples at the output of the receiving filter are given by

$$\begin{aligned} B_m &= I_m + I_{m-1} \\ &= (2P_m - 1) + (2P_{m-1} - 1) \\ &= 2(P_m + P_{m-1} - 1) \end{aligned} \tag{9-2-47}$$

Consequently,

$$P_m + P_{m-1} = \tfrac{1}{2}B_m + 1 \tag{9-2-48}$$

†Although this is identical to modulo-2 addition, it is convenient to view the precoding operation for duobinary in terms of modulo-2 subtraction.

TABLE 9-2-1 BINARY SIGNALING WITH DUOBINARY PULSES

Data sequence D_n		1	1	1	0	1	0	0	1	0	0	0	1	1	0	1
Precoded sequence P_n	0	1	0	1	1	0	0	0	1	1	1	1	0	1	1	0
Transmitted sequence I_m	−1	1	−1	1	1	−1	−1	−1	1	1	1	1	−1	1	1	−1
Received sequence B_n		0	0	0	2	0	−2	−2	0	2	2	2	0	0	2	0
Decoded sequence D_n		1	1	1	0	1	0	0	1	0	0	0	1	1	0	1

Since $D_m = P_m \oplus P_{m-1}$, it follows that the data sequence D_m is obtained from B_m using the relation

$$D_m = \tfrac{1}{2}B_m + 1 \pmod 2 \tag{9-2-49}$$

Consequently, if $B_m = \pm 2$ then $D_m = 0$, and if $B_m = 0$ then $D_m = 1$. An example that illustrates the precoding and decoding operations is given in Table 9-2-1. In the presence of additive noise, the sampled outputs from the receiving filter are given by (9-2-45). In this case $y_m = B_m + v_m$ is compared with the two thresholds set at +1 and −1. The data sequence $\{D_n\}$ is obtained according to the detection rule

$$D_m = \begin{cases} 1 & (|y_m| < 1) \\ 0 & (|y_m| \geq 1) \end{cases} \tag{9-2-50}$$

The extension from binary PAM to multilevel PAM signaling using the duobinary pulses is straightforward. In this case the M-level amplitude sequence $\{I_m\}$ results in a (noise-free) sequence

$$B_m = I_m + I_{m-1}, \qquad m = 1, 2, \ldots \tag{9-2-51}$$

which has $2M - 1$ possible equally spaced levels. The amplitude levels are determined from the relation

$$I_m = 2P_m - (M - 1) \tag{9-2-52}$$

where $\{P_m\}$ is the precoded sequence that is obtained from an M-level data sequence $\{D_m\}$ according to the relation

$$P_m = D_m \ominus P_{m-1} \pmod M \tag{9-2-53}$$

where the possible values of the sequence $\{D_m\}$ are $0, 1, 2, \ldots, M - 1$.

In the absence of noise, the samples at the output of the receiving filter may be expressed as

$$B_m = I_m + I_{m-1} = 2[P_m + P_{m-1} - (M - 1)] \tag{9-2-54}$$

TABLE 9-2-2 FOUR-LEVEL SIGNAL TRANSMISSION WITH DUOBINARY PULSES

Data sequence D_m		0	0	1	3	1	2	0	3	3	2	0	1	0
Precoded sequence P_m	0	0	0	1	2	3	3	1	2	1	1	3	2	2
Transmitted sequence I_m	−3	−3	−3	−1	1	3	3	−1	1	−1	−1	3	1	1
Received sequence B_m		−6	−6	−4	0	4	6	2	0	0	−2	2	4	2
Decoded sequence D_m		0	0	1	3	1	2	0	3	3	2	0	1	0

Hence,

$$P_m + P_{m-1} = \tfrac{1}{2}B_m + (M-1) \tag{9-2-55}$$

Since $D_m = P_m + P_{m-1} \pmod M$, it follows that

$$D_m = \tfrac{1}{2}B_m + (M-1) \pmod M \tag{9-2-56}$$

An example illustrating multilevel precoding and decoding is given in Table 9-2-2.

In the presence of noise, the received signal-plus-noise is quantized to the nearest of the possible signal levels and the rule given above is used on the quantized values to recover the data sequence.

In the case of the modified duobinary pulse, the controlled ISI is specified by the values $x(n/2W) = -1$, for $n = 1$, $x(n/2W) = 1$ for $n = -1$, and zero otherwise. Consequently, the noise-free sampled output from the receiving filter is given as

$$B_m = I_m - I_{m-2} \tag{9-2-57}$$

where the M-level sequence $\{I_m\}$ is obtained by mapping a precoded sequence according to the relation (9-2-52) and

$$P_m = D_m \ominus P_{m-2} \pmod M \tag{9-2-58}$$

From these relations, it is easy to show that the detection rule for recovering the data sequence $\{D_m\}$ from $\{B_m\}$ in the absence of noise is

$$D_m = \tfrac{1}{2}B_m \pmod M \tag{9-2-59}$$

As demonstrated above, the precoding of the data at the transmitter makes it possible to detect the received data on a symbol-by-symbol basis without having to look back at previously detected symbols. Thus, error propagation is avoided.

The symbol-by-symbol detection rule described above is not the optimum detection scheme for partial response signals due to the memory inherent in

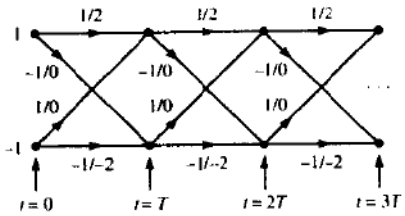


FIGURE 9-2-11 Trellis for duobinary partial response signal.

the received signal. Nevertheless, symbol-by-symbol detection is relatively simple to implement and is used in many practical applications involving duobinary and modified duobinary pulse signals. Its performance is evaluated in the following section.

Maximum-Likelihood Sequence Detection It is clear from the above discussion that partial-response waveforms are signal waveforms with memory. This memory is conveniently represented by a trellis. For example, the trellis for the duobinary partial-response signal for binary data transmission is illustrated in Fig. 9-2-11. For binary modulation, this trellis contains two states, corresponding to the two possible input values of I_m, i.e., $I_m = \pm 1$. Each branch in the trellis is labeled by two numbers. The first number on the left is the new data bit, i.e., $I_{m+1} = \pm 1$. This number determines the transition to the new state. The number on the right is the received signal level.

The duobinary signal has a memory of length $L = 1$. Hence, for binary modulation the trellis has $S_t = 2$ states. In general, for M-ary modulation, the number of trellis states is M^L.

The optimum maximum-likelihood (ML) sequence detector selects the most probable path through the trellis upon observing the received data sequence $\{y_m\}$ at the sampling instants $t = mT$, $m = 1, 2, \ldots$. In general, each node in the trellis will have M incoming paths and M corresponding metrics. One out of the M incoming paths is selected as the most probable, based on the values of the metrics and the other $M - 1$ paths and their metrics are discarded. The surviving path at each node is then extended to M new paths, one for each of the M possible input symbols, and the search process continues. This is basically the Viterbi algorithm for performing the trellis search.

For the class of partial response signals, the received sequence $\{y_m, 1 \leq m \leq N\}$ is generally described statistically by the joint pdf $f(\mathbf{y}_N \mid \mathbf{I}_N)$, where $\mathbf{y}_N = [y_1 \; y_2 \cdots y_N]'$ and $\mathbf{I}_N = [I_1 \; I_2 \cdots I_N]'$ and $N > L$. When the additive noise is zero-mean gaussian, $f(\mathbf{y}_N \; \mathbf{I}_N)$ is a multivariate gaussian pdf, i.e.,

$$f(\mathbf{y}_N \mid \mathbf{I}_N) = \frac{1}{(2\pi \det \mathbf{C})^{N/2}} \exp\left[-\tfrac{1}{2}(\mathbf{y}_N - \mathbf{B}_N)'\mathbf{C}^{-1}(\mathbf{y}_N - \mathbf{B}_N)\right] \qquad (9\text{-}2\text{-}60)$$

where $\mathbf{B}_N = [B_1 \; B_2 \cdots B_N]'$, is the mean of the vector $\mathbf{y}_N$ and $\mathbf{C}$ is the $N \times N$ covariance matrix of $\mathbf{y}_N$. Then, the ML sequence detector selects the sequence through the trellis that maximizes the pdf $f(\mathbf{y}_N \mid \mathbf{I}_N)$.

The computation for finding the most probable sequence through the trellis is simplified by taking the natural logarithms of $f(\mathbf{y}_N \mid \mathbf{I}_N)$. Thus,

$$\ln f(\mathbf{y}_N \mid \mathbf{I}_N) = -\tfrac{1}{2}N \ln (2\pi \det \mathbf{C}) - \tfrac{1}{2}(\mathbf{y}_N - \mathbf{B}_N)'\mathbf{C}^{-1}(\mathbf{y}_N - \mathbf{B}_N) \quad (9\text{-}2\text{-}61)$$

Given the received sequence $\{y_m\}$, the data sequence $\{I_m\}$ that maximizes $\ln f(\mathbf{y}_N \mid \mathbf{I}_N)$ is identical to the sequence $\{I_N\}$ that minimizes $(\mathbf{y}_N - \mathbf{B}_N)'\mathbf{C}^{-1}(\mathbf{y}_N - \mathbf{B}_N)$, i.e.

$$\hat{\mathbf{I}}_N = \arg\min_{\mathbf{I}_N} [(\mathbf{y}_N - \mathbf{B}_N)'\mathbf{C}^{-1}(\mathbf{y}_N - \mathbf{B}_N)] \quad (9\text{-}2\text{-}62)$$

The metric computations in the trellis search are complicated by the correlation of the noise samples at the output of the matched filter for the partial response signal. For example, in the case of the duobinary signal waveform, the correlation of the noise sequence $\{v_m\}$ is over two successive signal samples. Hence, v_m and v_{m+k} are correlated for $k = 1$ and uncorrelated for $k > 1$. In general, a partial response signal waveform with memory L will result in a correlated noise sequence at the output of the matched filter, which satisfies the condition $E[v_m v_{m+k}] = 0$ for $k > L$. In such a case, the Viterbi algorithm for performing the trellis search may be modified as described in Chapter 10.

Some simplification in the metric computations result if we ignore the noise correlation by assuming that $E(v_m v_{m+k}) = 0$ for $k > 0$. Then, by assumption, the covariance matrix $\mathbf{C} = \sigma_v^2 \mathbf{1}_N$, where $\sigma_v^2 = E[v_m^2]$ and $\mathbf{1}_N$ is the $N \times N$ identity matrix.† In this case, (9-2-62) simplifies to

$$\begin{aligned}\hat{\mathbf{I}}_N &= \arg\min_{\mathbf{I}_N} [(\mathbf{y}_N - \mathbf{B}_N)'(\mathbf{y}_N - \mathbf{B}_N)] \\ &= \arg\min_{\mathbf{I}_N} \left[\sum_{m=1}^{N} \left(y_m - \sum_{k=0}^{L} x_k I_{m-k} \right)^2 \right] \end{aligned} \quad (9\text{-}2\text{-}63)$$

where

$$B_m = \sum_{k=0}^{L} x_k I_{m-k}$$

and $x_k = x(kT)$ are the sampled values of the partial response signal waveform. In this case, the metric computations at each node of the trellis have the form

$$DM_m(\mathbf{I}_m) = DM_{m-1}(\mathbf{I}_{m-1}) + \left(y_m - \sum_{k=0}^{L} x_k I_{m-k} \right)^2 \quad (9\text{-}2\text{-}64)$$

where $DM_m(\mathbf{I}_m)$ are the distance metrics at time $t = mT$, $DM_{m-1}(\mathbf{I}_{m-1})$ are the distance metrics at time $t = (m-1)T$ and the second term on the right-hand side of (9-2-64) represents the new increments to the metrics based on the new received sample y_m.

†We are using $\mathbf{1}_N$ here to avoid confusion with $\mathbf{I}_N$.

As indicated in Section 5-1-4, ML sequence detection introduces a variable delay in detecting each transmitted information symbol. In practice, the variable delay is avoided by truncating the surviving sequences to N_t most recent symbols, where $N_t \gg 5L$, thus achieving a fixed delay. In the case that the M^L surviving sequences at time $t = mT$ disagree on the symbol I_{m-N_t}, the symbol in the most probable surviving sequence may be chosen. The loss in performance resulting from this truncation is negligible if $N_t > 5L$.

9-2-4 Signal Design for Channels with Distortion

In Sections 9-2-1 and 9-2-2, we described signal design criteria for the modulation filter at the transmitter and the demodulation filter at the receiver when the channel is ideal. In this section, we perform the signal design under the condition that the channel distorts the transmitted signal. We assume that the channel frequency response $C(f)$ is known for $|f| \leq W$ and that $C(f) = 0$ for $|f| > W$. The criterion for the optimization of the filter responses $G_T(f)$ and $G_R(f)$ is the maximization of the SNR at the output of the demodulation filter or equivalently, at the input to the detector. The additive channel noise is assumed to be gaussian with power spectral density $\Phi_{nn}(f)$. Figure 9-2-12 illustrates the overall system under consideration.

For the signal component at the output of the demodulator, we must satisfy the condition

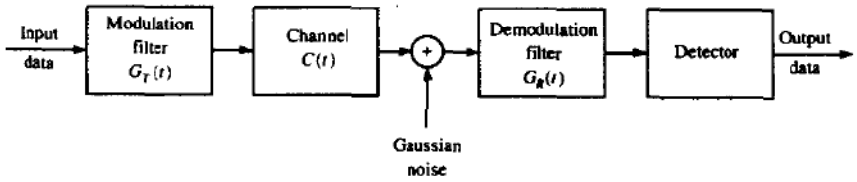
$$G_T(f)C(f)G_R(f) = X_d(f)e^{-j2\pi f t_0}, \quad |f| \leq W \tag{9-2-65}$$

where $X_d(f)$ is the desired frequency response of the cascade of the modulator, channel, and demodulator, and t_0 is a time delay that is necessary to ensure the physical realizability of the modulation and demodulation filters. The desired frequency response $X_d(f)$ may be selected to yield either zero ISI or controlled ISI at the sampling instants. We shall carry out the optimization for zero ISI by selecting $X_d(f) = X_{rc}(f)$, where $X_{rc}(f)$ is the raised cosine spectrum with an arbitrary rolloff factor.

The noise at the output of the demodulation filter may be expressed as

$$v(t) = \int_{-\infty}^{\infty} n(t-\tau)g_R(\tau)\,d\tau \tag{9-2-66}$$

FIGURE 9-2-12 System model for the design of the modulation and demodulation filters.



where $n(t)$ is the input to the filter. Since $n(t)$ is zero-mean gaussian, $v(t)$ is zero-mean gaussian, with a power spectral density

$$\Phi_{vv}(f) = \Phi_{nn}(f)\,|G_R(f)|^2 \tag{9-2-67}$$

For simplicity, we consider binary PAM transmission. Then, the sampled output of the matched filter is

$$y_m = x_0 I_m + v_m = I_m + v_m \tag{9-2-68}$$

where x_0 is normalized† to unity, $I_m = \pm d$, and v_m represents the noise term, which is zero-mean gaussian with variance

$$\sigma_v^2 = \int_{-\infty}^{\infty} \Phi_{nn}(f)\,|G_R(f)|^2\,df \tag{9-2-69}$$

Consequently, the probability of error is

$$P_2 = \frac{1}{\sqrt{2\pi}} \int_{d/\sigma_v}^{\infty} e^{-y^2/2}\,dy = Q(\sqrt{d^2/\sigma_v^2}) \tag{9-2-70}$$

The probability of error is minimized by maximizing the SNR $= d^2/\sigma_v^2$, or, equivalently, by minimizing the noise-to-signal ratio σ_v^2/d^2. But d^2 is related to the transmitted signal power as follows:

$$\begin{aligned} P_{av} &= \frac{E(I_m^2)}{T} \int_{-\infty}^{\infty} g_T^2(t)\,dt = \frac{d^2}{T} \int_{-\infty}^{\infty} g_T^2(t)\,dt \\ \frac{1}{d^2} &= \frac{1}{P_{av} T} \int_{-\infty}^{\infty} |G_T(f)|^2\,df \end{aligned} \tag{9-2-71}$$

However, $G_T(f)$ must be chosen to satisfy the zero ISI condition. Consequently,

$$|G_T(f)| = \frac{|X_{rc}(f)|}{|C(f)|\,|G_R(f)|}, \qquad |f| \leq W \tag{9-2-72}$$

and $G_T(f) = 0$ for $|f| \geq W$. Hence

$$\frac{1}{d^2} = \frac{1}{P_{av} T} \int_{-W}^{W} \frac{|X_{rc}(f)|^2}{|C(f)|^2\,|G_R(f)|^2}\,df \tag{9-2-73}$$

Therefore, the noise-to-signal ratio that must be minimized with respect to $|G_R(f)|$ for $|f| \leq W$ is

$$\frac{\sigma_v^2}{d^2} = \frac{1}{P_{av} T} \int_{-W}^{W} \Phi_{nn}(f)\,|G_R(f)|^2\,df \int_{-W}^{W} \frac{|X_{rc}(f)|^2}{|C(f)|^2\,|G_R(f)|^2}\,df \tag{9-2-74}$$

†By setting $x_0 = 1$ and $I_m = \pm d$, the scaling by x_0 is incorporated into the parameter d.

The optimum $|G_R(f)|$ can be found by applying the Cauchy–Schwartz inequality,

$$\int_{-\infty}^{\infty} |U_1(f)|^2\, df \int_{-\infty}^{\infty} |U_2(f)|^2\, df \geq \left[\int_{-\infty}^{\infty} |U_1(f)|\,|U_2(f)|\, df\right]^2 \tag{9-2-75}$$

where $|U_1(f)|$ and $|U_2(f)|$ are defined as

$$\begin{aligned} |U_1(f)| &= |\sqrt{\Phi_{nn}(f)}|\,|G_R(f)| \\ |U_2(f)| &= \frac{|X_{rc}(f)|}{|C(f)|\,|G_R(f)|} \end{aligned} \tag{9-2-76}$$

The minimum value of (9-2-74) is obtained when $|U_1(f)|$ is proportional to $|U_2(f)|$, or, equivalently, when

$$|G_R(f)| = K\frac{|X_{rc}(f)|^{1/2}}{[\Phi_{nn}(f)]^{1/4}\,|C(f)|^{1/2}}, \qquad |f| \leq W \tag{9-2-77}$$

where K is an arbitrary constant. The corresponding modulation filter has a magnitude characteristic

$$|G_T(f)| = \frac{1}{K}\frac{|X_{rc}(f)|^{1/2}\,[\Phi_{nn}(f)]^{1/4}}{|C(f)|^{1/2}}, \qquad |f| \leq W \tag{9-2-78}$$

Finally, the maximum SNR achieved by these optimum transmitting and receiving filters is

$$\frac{d^2}{\sigma_\nu^2} = \frac{P_{av}T}{\{\int_{-W}^{W} |X_{rc}(f)|\,[\Phi_{nn}(f)]^{1/2}\,|C(f)|^{-1}\, df\}^2} \tag{9-2-79}$$

We note that the optimum modulation and demodulation filters are specified in magnitude only. The phase characteristics for $G_T(f)$ and $G_R(f)$ may be selected so as to satisfy the condition in (9-2-65), i.e.,

$$\Theta_T(f) + \Theta_c(f) + \Theta_R(f) = 2\pi f t_0 \tag{9-2-80}$$

where $\Theta_T(f)$, $\Theta_c(f)$, and $\Theta_R(f)$ are the phase characteristics of the modulation filter, the channel, and the demodulation filter, respectively.

In the special case where the additive noise at the input to the demodulator is white gaussian with spectral density $\frac{1}{2}N_0$, the optimum filter characteristics specified by (9-2-77) and (9-2-78) reduce to

$$\begin{aligned} |G_R(f)| &= K_1\frac{|X_{rc}(f)|^{1/2}}{|C(f)|^{1/2}}, \qquad |f| \leq W \\ |G_T(f)| &= K_2\frac{|X_{rc}(f)|^{1/2}}{|C(f)|^{1/2}}, \qquad |f| \leq W \end{aligned} \tag{9-2-81}$$

where K_1 and K_2 are arbitrary scale factors. Note that, in this case, $|G_R(f)|$ is the matched filter to $|G_T(f)|$. The corresponding SNR at the detector, given by (9-2-79) reduces to

$$\frac{d^2}{\sigma_\nu^2} = \frac{2P_{av}T}{N_0}\left[\int_{-W}^{W} \frac{|X_{rc}(f)|}{|C(f)|}\,df\right]^{-2} \tag{9-2-82}$$

Example 9-2-1

Let us determine the optimum transmitting and receiving filters for a binary communication system that transmits data at a rate of 4800 bits/s over a channel with frequency (magnitude) response

$$|C(f)| = \frac{1}{\sqrt{1 + (f/W)^2}}, \qquad |f| \leq W \tag{9-2-83}$$

where $W = 4800$ Hz. The additive noise is zero-mean, white, gaussian with spectral density $\frac{1}{2}N_0 = 10^{-15}\,W/Hz$.

Since $W = 1/T = 4800$, we use a signal pulse with a raised cosine spectrum and $\beta = 1$. Thus,

$$\begin{aligned} X_{rc}(f) &= \tfrac{1}{2}T[1 + \cos(\pi T\,|f|)] \\ &= T\cos^2\left(\frac{\pi\,|f|}{9600}\right) \end{aligned} \tag{9-2-84}$$

Then,

$$|G_T(f)| = |G_R(f)| = \left[1 + \left(\frac{f}{4800}\right)^2\right]^{1/4} \cos\left(\frac{\pi\,|f|}{9600}\right), \qquad |f| \leq 4800 \tag{9-2-85}$$

and $|G_T(f)| = |G_R(f)| = 0$, otherwise. Figure 9-2-13 illustrates the filter characteristic $G_T(f)$.

One can now use these optimum filters to determine the amount of transmitted energy $\mathscr{E}$ required to achieve a specified error probability. This problem is left as an exercise for the reader.

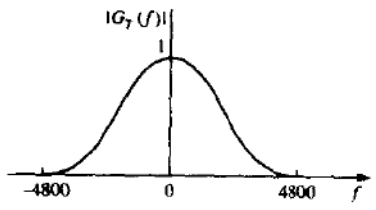


FIGURE 9-2-13 Frequency response of optimum transmitter filter.

9-3 PROBABILITY OF ERROR IN DETECTION OF PAM

In this section, we evaluate the performance of the receiver for demodulating and detecting an M-ary PAM signal in the presence of additive, white, gaussian noise at its input. First, we consider the case in which the transmitter and receiver filters $G_T(f)$ and $G_R(f)$ are designed for zero ISI. Then, we consider the case in which $G_T(f)$ and $G_R(f)$ are designed such that $x(t) = g_T(t) \star g_R(t)$ is either a duobinary signal or a modified duobinary signal.

9-3-1 Probability of Error for Detection of PAM with Zero ISI

In the absence of ISI, the received signal sample at the output of the receiving matched filter has the form

$$y_m = x_0 I_m + \nu_m \tag{9-3-1}$$

where

$$x_0 = \int_{-W}^{W} |G_T(f)|^2 \, df = \mathcal{E}_g \tag{9-3-2}$$

and ν_m is the additive gaussian noise that has zero mean and variance

$$\sigma_\nu^2 = \tfrac{1}{2}\mathcal{E}_g N_0 \tag{9-3-3}$$

In general, I_m takes one of M possible equally spaced amplitude values with equal probability. Given a particular amplitude level, the problem is to determine the probability of error.

The problem of evaluating the probability of error for digital PAM in a band-limited, additive white gaussian noise channel, in the absence of ISI, is identical to the evaluation of the error probability for M-ary PAM as given in Section 5-2. The final result that is obtained from the derivation is

$$P_M = \frac{2(M-1)}{M} Q\left(\sqrt{\frac{2\mathcal{E}_g}{N_0}}\right) \tag{9-3-4}$$

But $\mathcal{E}_g = 3\mathcal{E}_{\text{av}}/(M^2-1)$, $\mathcal{E}_{\text{av}} = k\mathcal{E}_{b\text{av}}$ is the average energy per symbol and $\mathcal{E}_{b\text{av}}$ is the average energy per bit. Hence,

$$P_M = \frac{2(M-1)}{M} Q\left(\sqrt{\frac{6(\log_2 M)\mathcal{E}_{b\text{av}}}{(M^2-1)N_0}}\right) \tag{9-3-5}$$

This is exactly the form for the probability of error of M-ary PAM derived in Section 5-2 (see (5-2-46)). In the treatment of PAM given in this chapter, we imposed the additional constraint that the transmitted signal is band-limited to the bandwidth allocated for the channel. Consequently, the transmitted signal pulses were designed to be band-limited and to have zero ISI.

In contrast, no bandwidth constraint was imposed on the PAM signals considered in Section 5-2. Nevertheless, the receivers (demodulators and detectors) in both cases are optimum (matched filters) for the corresponding

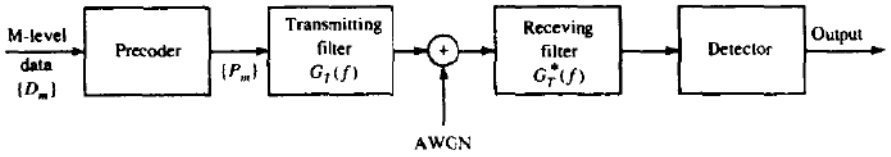


FIGURE 9-3-1 Block diagram of modulator and demodulator for partial-response signals.

transmitted signals. Consequently, no loss in error rate performance results from the bandwidth constraint when the signal pulse is designed for zero ISI and the channel does not distort the transmitted signal.

9-3-2 Probability of Error for Detection of Partial-Response Signals

In this section we determine the probability of error for detection of digital M-ary PAM signaling using duobinary and modified duobinary pulses. The channel is assumed to be an ideal bandlimited channel with additive white gaussian noise. The model for the communication system is shown in Fig. 9-3-1.

We consider two types of detectors. The first is the symbol-by-symbol detector and the second is the optimum ML sequence detector described in the previous section.

Symbol-by-Symbol Detector At the transmitter, the M-level data sequence $\{D_m\}$ is precoded as described previously. The precoder output is mapped into one of M possible amplitude levels. Then the transmitting filter with frequency response $G_T(f)$ has an output

$$v(t) = \sum_{n=-\infty}^{\infty} I_n g_T(t - nT) \tag{9-3-6}$$

The partial-response function $X(f)$ is divided equally between the transmitting and receiving filters. Hence, the receiving filter is matched to the transmitted pulse, and the cascade of the two filters results in the frequency characteristic

$$|G_T(f)G_R(f)| = |X(f)| \tag{9-3-7}$$

The matched filter output is sampled at $t = nT = n/2W$ and the samples are fed to the decoder. For the duobinary signal, the output of the matched filter at the sampling instant may be expressed as

$$y_m = I_m + I_{m-1} + \nu_m = B_m + \nu_m \tag{9-3-8}$$

where ν_m is the additive noise component. Similarly, the output of the matched filter for the modified duobinary signal is

$$y_m = I_m - I_{m-2} + \nu_m = B_m + \nu_m \tag{9-3-9}$$

For binary transmission, let $I_m = \pm d$, where $2d$ is the distance between signal levels. Then, the corresponding values of B_m are $(2d, 0, -2d)$. For M-ary PAM signal transmission, where $I_m = \pm d, \pm 3d, \ldots, \pm(M-1)d$, the received signal levels are $B_m = 0, \pm 2d, \pm 4d, \ldots, \pm 2(M-1)d$. Hence, the number of received levels is $2M-1$, and the scale factor d is equivalent to $x_0 = \mathscr{E}_g$.

The input transmitted symbols $\{I_m\}$ are assumed to be equally probable. Then, for duobinary and modified duobinary signals, it is easily demonstrated that, in the absence of noise, the received output levels have a (triangular) probability distribution of the form

$$P(B = 2md) = \frac{M - |m|}{M^2}, \qquad m = 0, \pm 1, \pm 2, \ldots, \pm(M-1) \tag{9-3-10}$$

where B denotes the noise-free received level and $2d$ is the distance between any two adjacent received signal levels.

The channel corrupts the signal transmitted through it by the addition of white gaussian noise with zero mean and power spectral density $\frac{1}{2}N_0$.

We assume that a symbol error occurs whenever the magnitude of the additive noise exceeds the distance d. This assumption neglects the rare event that a large noise component with magnitude exceeding d may result in a received signal level that yields a correct symbol decision. The noise component v_m is zero-mean gaussian with variance

$$\begin{aligned}\sigma_v^2 &= \tfrac{1}{2}N_0 \int_{-W}^{W} |G_R(f)|^2\, df \\ &= \tfrac{1}{2}N_0 \int_{-W}^{W} |X(f)|\, df = 2N_0/\pi\end{aligned} \tag{9-3-11}$$

for both the duobinary and the modified duobinary signals. Hence, an upper bound on the symbol probability of error is

$$\begin{aligned}P_M < & \sum_{m=-(M-2)}^{M-2} P(|y - 2md| > d \mid B = 2md) P(B = 2md) \\ & + 2P(y + 2(M-1)d > d \mid B = -2(M-1)d) P(B = -2(M-1)d) \\ = & P(|y| > d \mid b = 0)\left[2 \sum_{m=0}^{M-1} P(B = 2md) - P(B = 0) - P(B = -2(M-1)d)\right] \\ = & (1 - M^{-2}) P(|y| > d \mid B = 0)\end{aligned} \tag{9-3-12}$$

But

$$\begin{aligned}P(|y| > d \mid B = 0) &= \frac{2}{\sqrt{2\pi}\sigma_v} \int_d^{\infty} e^{-x^2/2\sigma_v^2}\, dx \\ &= 2Q(\sqrt{\pi d^2/2N_0})\end{aligned} \tag{9-3-13}$$

Therefore, the average probability of a symbol error is upper-bounded as

$$P_M < 2(1 - M^{-2})Q(\sqrt{\pi d^2/2N_0}) \tag{9-3-14}$$

The scale factor d in (9-3-14) can be eliminated by expressing it in terms of the average power transmitted into the channel. For the M-ary PAM signal in which the transmitted levels are equally probable, the average power at the output of the transmitting filter is

$$\begin{aligned} P_{av} &= \frac{E(I_m^2)}{T}\int_{-W}^{W} |G_T(f)|^2\, df \\ &= \frac{E(I_m^2)}{T}\int_{-W}^{W} |X(f)|\, df = \frac{4}{\pi T}E(I_m^2) \end{aligned} \tag{9-3-15}$$

where $E(I_m^2)$ is the mean square value of the M signal levels, which is

$$E(I_m^2) = \tfrac{1}{3}d^2(M^2 - 1) \tag{9-3-16}$$

Therefore,

$$d^2 = \frac{3\pi P_{av} T}{4(M^2 - 1)} \tag{9-3-17}$$

By substituting the value of d^2 from (9-3-17) into (9-3-14), we obtain the upper bound on the symbol error probability as

$$P_M < 2\left(1 - \frac{1}{M^2}\right)Q\left(\sqrt{\left(\frac{\pi}{4}\right)^2 \frac{6}{M^2 - 1}\frac{\mathscr{E}_{av}}{N_0}}\right) \tag{9-3-18}$$

where $\mathscr{E}_{av}$ is the average energy per transmitted symbol, which can be also expressed in terms of the average bit energy as $\mathscr{E}_{av} = k\mathscr{E}_{bav} = (\log_2 M)\mathscr{E}_{bav}$.

The expression in (9-3-18) for the probability of error of M-ary PAM holds for both duobinary and modified duobinary partial-response signals. If we compare this result with the error probability of M-ary PAM with zero ISI, which can be obtained by using a signal pulse with a raised cosine spectrum, we note that the performance of partial response duobinary or modified duobinary has a loss of $(\frac{1}{4}\pi)^2$, or 2.1 dB. This loss in SNR is due to the fact that the detector for the partial response signals makes decisions on a symbol-by symbol basis, thus ignoring the inherent memory contained in the received signal at the input to the detector.

Maximum-Likelihood Sequence Detector The ML sequence detector searches through the trellis for the most probable transmitted sequence $\{I_m\}$ as previously described in Section 9-2-3. At each stage of the search process the detector compares the metrics of paths that merge at each of the nodes and selects the path that is most probable at each node. The performance of the detector may be evaluated by determining the probability of error events, based on a euclidean distance metric, as was done for soft-decision decoding of convolutional codes. The general derivation is given in Section 10-1-4. In the

case of the duobinary and modified duobinary signals, it is demonstrated that the 2.1 dB loss inherent in the suboptimum symbol-by-symbol detector is completely recovered by the ML sequence detector.

9-3-3 Probability of Error for Optimum Signals in a Channel with Distortion

In Section 9-2-4, we derived the filter responses for the modulation and demodulation filters that maximize the SNR at the input to the detector when there is channel distortion. When the filters are designed for zero ISI at the sampling instants, the probability of error for M-ary PAM is

$$P_M = \frac{2(M-1)}{M} Q(\sqrt{d^2/\sigma_v^2}) \tag{9-3-19}$$

The parameter d is related to the average transmitted power as

$$\begin{aligned} P_{\text{av}} &= \frac{E[I_m^2]}{T} \int_{-W}^{W} |G_T(f)|^2 \, df \\ &= \frac{(M^2-1)d^2}{3T} \int_{-W}^{W} |G_T(f)|^2 \, df \end{aligned} \tag{9-3-20}$$

and the noise variance is given by (9-2-69). For AWGN, (9-3-19) may be expressed as

$$P_M = \frac{2(M-1)}{M} Q\left(\sqrt{\frac{6\mathscr{E}_{\text{av}}}{(M^2-1)N_0} \left[\int_{-W}^{W} \frac{|X_{rc}(f)|}{|C(f)|} df \right]^{-2}} \right) \tag{9-3-21}$$

Finally, we observe that the loss due to channel distortion is

$$20 \log_{10} \left[\int_{-W}^{W} \frac{|X_{rc}(f)|}{|C(f)|} df \right] \tag{9-3-22}$$

Note that when $C(f) = 1$ for $|f| \leq W$, the channel is ideal and

$$\int_{-W}^{W} X_{rc}(f) \, df = 1 \tag{9-3-23}$$

so that no loss is incurred. On the other hand, when there is amplitude distortion, $|C(f)| < 1$ for some range of frequencies in the band $|f| \leq W$ and, hence, there is a loss in SNR incurred, as given by (9-3-22). This loss is independent of channel phase distortion, because phase distortion has been perfectly compensated, as implied by (9-2-80). The loss given by (9-3-22) is due entirely to amplitude distortion and is a measure of the noise enhancement

resulting from the receiving filter, which compensates for the channel distortion.

9-4 MODULATION CODES FOR SPECTRUM SHAPING

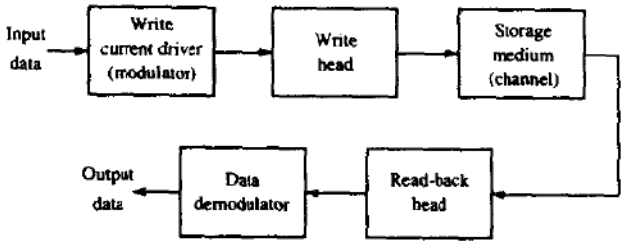
We have observed that the power spectral density of a digital communication signal can be controlled and shaped by selecting the transmitted signal pulse $g(t)$ and by introducing correlation through coding, which is used to combat channel distortion and noise in transmission. Coding for spectrum shaping is introduced following the channel encoding so that the spectrum of the transmitted signal matches the spectral characteristics of a baseband or equivalent lowpass channel.

Codes that are used for spectrum shaping are generally called either *modulation codes,* or *line codes,* or *data translation codes.* Such codes generally place restrictions on the sequence of bits into the modulator and, thus, introduce correlation and, hence, memory into the transmitted signal. It is this type of coding that is treated in this section.

Modulation codes are usually employed in magnetic recording, in optical recording, and in digital communications over cable systems to achieve spectral shaping and to eliminate or minimize the d.c. content in the transmitted (or stored) baseband signal. In magnetic recording channels, the modulation code is designed to increase the distance between transitions in the recorded waveform and, thus, intersymbol interference effects are also reduced.

As an example of the use of a modulation code, let us consider a magnetic recording system, which consists of the elements shown in the block diagram of Fig. 9-4-1. The binary data sequence to be stored is used to generate a write current. This current may be viewed as the output for the "modulator." The most commonly used method to map the information sequence into the write current waveform is NRZI, which was described in Section 4-3-2. Recall that in NRZI, a transition from one amplitude to another (A to $-$A or $-$A to A) occurs only when the information bit is a 1. No transition occurs when the information bit is a 0, i.e., the amplitude level remains the same as in the previous signal interval. The positive amplitude pulse results in magnetizing

FIGURE 9-4-1 Block diagram of magnetic storage read/write system.



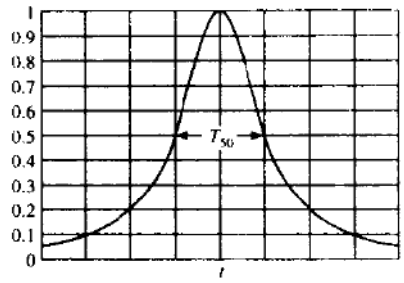


FIGURE 9-4-2 Read-back pulse in magnetic recording system.

the medium in one (direction) polarity and the negative pulse magnetizes the medium in the opposite (direction) polarity.

Since the input data sequence is basically random with equally probable 1 s and 0 s, we shall encounter level transitions from A to $-A$ or $-A$ to A with probability 1/2 for every data bit. The readback signal for a positive transition ($-A$ to A) is a pulse that is well-modeled mathematically as

$$g(t) = \frac{1}{1 + (2t/T_{50})^2} \tag{9-4-1}$$

where T_{50} is defined as the width of the pulse at its 50% amplitude level, as shown in Fig. 9-4-2. Similarly, the readback signal for a negative transition (A to $-A$) is the pulse $-g(t)$. The value of T_{50} is determined by the characteristics of the medium, the read/write heads, and the distance of the head to the medium.

Now, suppose we write a positive transition followed by a negative transition. Let's vary the time interval between the two transitions, which we denote as T_b (the bit time interval). Figure 9-4-3 illustrates the readback signal pulses, which are obtained by a superposition of $p(t)$ with $-p(t - T_b)$. The parameter $\Delta = T_{50}/T_b$ is defined as the *normalized density. The* closer the bit transitions (T_b small), the larger will be the value of the normalized density and, hence, the larger will be the packing density. We notice that as Δ is

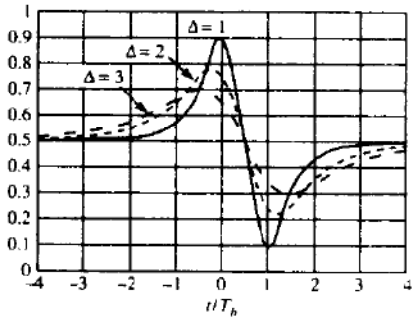


FIGURE 9-4-3 Read-back signal response to a pulse.

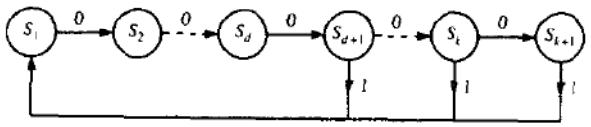
increased, the peak amplitudes of the readback signal are reduced and are also shifted in time from the desired time instants. In other words, the pulses interfere with one another, thus limiting the density with which we can write. This problem serves as a motivation to design modulation codes that take the original data sequence and transform (encode) it into another sequence that results in a write waveform in which amplitude transitions are spaced farther apart. For example, if we use NRZI, the encoded sequence into the modulator must contain one or more 0s between 1s.

The second problem encountered in magnetic recording is the need to avoid (or minimize) having a d.c. content in the modulated signal (the write current) due to the frequency response characteristics of the readback system and associated electronics. This requirement also arises in digital communication over cable channels. This problem can be overcome by altering (encoding) the data sequence into the modulator. A class of codes that satisfy these objectives are the modulation codes described below.

Runlength-Limited Codes Codes that have a restriction on the number of consecutive 1s or 0s in a sequence are generally called *runlength-limited codes*. These codes are generally described by two parameters, say d and κ, where d denotes the minimum number of 0s between two 1s in a sequence, and κ denotes the maximum number of 0s between two 1s in a sequence. When used with NRZI modulation, the effect of placing d zeros between successive 1s is to spread the transitions farther apart, thus reducing the overlap in the channel response due to successive transitions and hence reducing the intersymbol interference. Setting an upper limit κ on the runlength of 0s ensures that transitions occur frequently enough so that symbol timing information can be recovered from the received modulated signal. Runlength-limited codes are usually called (d, κ) codes.†

The (d, κ) code sequence constraints may be represented by a finite-state sequential machine with $\kappa + 1$ states, denoted as S_i, $1 \leq i \leq \kappa + 1$, as shown in Fig. 9-4-4. We observe that an output data bit 0 takes the sequence from state S_i to S_{i+1}, $i \leq \kappa$. The output data bit 1 takes the sequence to state S_1. The output bit from the encoder may be a 1 only when the sequence is in state S_i, $d + 1 \leq i \leq \kappa + 1$. When the sequence is in state $S_{\kappa+1}$, the output bit is always 1.

FIGURE 9-4-4 Finite-state sequential machine for a (d, κ)-coded sequence.



†In fact, they are usually called (d, k) codes, where k is the maximum runlength of zeros. We have substituted the Greek letter kappa κ for k, to avoid confusion with our previous use of k.

The finite-state sequential machine may also be represented by a *state transition matrix*, denoted as **D**, which is a square $(\kappa+1)\times(\kappa+1)$ with elements d_{ij}, where

$$d_{i1} = 1 \quad (i \geq d+1)$$
$$d_{ij} = \begin{cases} 1 & (j = i+1) \\ 0 & (\text{otherwise}) \end{cases} \tag{9-4-2}$$

Example 9-4-1

Let us determine the state transition matrix for a $(d, \kappa) = (1, 3)$ code. The $(1, 3)$ code has four states. From Fig. 9-4-4, we obtain its state transition matrix, which is

$$\mathbf{D} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix} \tag{9-4-3}$$

An important parameter of any (d, κ) code is the number of sequences of a certain length, say n, that satisfy the (d, κ) constraints. As n is allowed to increase, the number of sequences $N(n)$ that satisfy the (d, κ) constraint also increases. The number of information bits that can be uniquely represented with $N(n)$ code sequences is

$$k = \lfloor \log_2 N(n) \rfloor$$

where $\lfloor x \rfloor$ denotes the largest integer contained in x. The maximum code rate is then $R_c = k/n$.

The capacity of a (d, κ) code is defined as

$$C(d, \kappa) = \lim_{n \to \infty} \frac{1}{n} \log_2 N(n) \tag{9-4-4}$$

Clearly, $C(d, \kappa)$ is the maximum possible rate that can be achieved with the (d, κ) constraints. Shannon (1948) showed that the capacity is given as

$$C(d, \kappa) = \log_2 \lambda_{\max} \tag{9-4-5}$$

where $\lambda_{\max}$ is the largest real eigenvalue of the state transition matrix **D**.

Example 9-4-2

Let us determine the capacity of a $(d, \kappa) = (1, 3)$ code. Using the state-transition matrix given in Example 9-4-1 for the $(1, 3)$ code, we have

$$\det(\mathbf{D} - \lambda \mathbf{I}) = \det \begin{bmatrix} -\lambda & 1 & 0 & 0 \\ 1 & -\lambda & 1 & 0 \\ 1 & 0 & -\lambda & 1 \\ 1 & 0 & 0 & -\lambda \end{bmatrix}$$
$$= \lambda^4 - \lambda^2 - \lambda - 1 = 0 \tag{9-4-6}$$

TABLE 9-4-1 CAPACITY $C(d, \kappa)$ VERSUS RUNLENGTH PARAMETERS d AND κ

κ	$d=0$	$d=1$	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$
2	.8791	.4057					
3	.9468	.5515	.2878				
4	.9752	.6174	.4057	.2232			
5	.9881	.6509	.4650	.3218	.1823		
6	.9942	.6690	.4979	.3746	.2269	.1542	
7	.9971	.6793	.5174	.4057	.3142	2281	.1335
8	.9986	.6853	.5293	.4251	.3432	.2709	.1993
9	.9993	.6888	.5369	.4376	.3620	.2979	.2382
10	.9996	.6909	.5418	.4460	.3746	.3158	.2633
11	.9998	.6922	.5450	.4516	.3833	.3285	.2804
12	.9999	.6930	.5471	.4555	.3894	.3369	.2924
13	.9999	.6935	.5485	.4583	.3937	.3432	.3011
14	.9999	.6938	.5495	.4602	.3968	.3478	.3074
15	.9999	.6939	.5501	.4615	.3991	.3513	.3122
∞	1.000	.6942	.5515	.4650	.4057	.3620	.3282

The maximum real root of this polynomial is found to be $\lambda_{max} = 1.4656$. Therefore, the capacity $C(1, 3) = \log_2 \lambda_{max} = 0.5515$.

The capacities of (d, κ) codes for $0 \leq d \leq 6$ and $2 \leq \kappa \leq 15$ are given in Table 9-4-1. We observe that $C(d, \kappa) < \frac{1}{2}$ for $d \geq 3$ and any value of κ. The most commonly used codes for magnetic recording employ $d \leq 2$; hence, their rate R_c is at least $\frac{1}{2}$.

Now let us turn our attention to the construction of some runlength-limited codes. In general, (d, κ) codes can be constructed either as fixed-length codes or as variable-length codes. In a fixed-length code, each bit or block of k bits is encoded into a block of $n > k$ bits.

In principle, the construction of a fixed-length code is straightforward. For a given block length n, we may select the subset of the 2^n code words that satisfy the specified runlength constraints. From this subset, we eliminate code words that do not satisfy the runlength constraints when concatenated. Thus, we obtain a set of code words that satisfy the constraints and can be used in the mapping of the input data bits to the encoder. The encoding and decoding operations can be performed by use of a look-up table.

Example 9-4-3

Let us construct a $d = 0$, $\kappa = 2$ code of length $n = 3$, and determine its efficiency. By listing all the code words, we find that the following five code words satisfy the (0, 2) constraint: (0 1 0), (0 1 1), (1 0 1), (1 1 0), (1 1 1). We may select any four of these code words and use then to encode the pairs of

data bits (00, 01, 10, 11). Thus, we have a rate $k/n = 2/3$ code that satisfies the (0, 2) constraint.

The fixed-length code in this example is not very efficient. The capacity is $C(0, 2) = 0.8791$, so that this code has an *efficiency* of

$$\text{efficiency} = \frac{R_c}{C(d, \kappa)} = \frac{2/3}{0.8791} = 0.76$$

Surely, better (0, 2) codes can be constructed by increasing the block length n.

In the following example, we place no restriction on the maximum runlength of zeros.

Example 9-4-4

Let us construct a $d = 1$, $\kappa = \infty$ code of length $n = 5$. In this case, we are placing no constraint on the number of consecutive zeros. To construct the code, we select from the set of 32 possible code words those that satisfy the $d = 1$ constraint. There are eight such code words, which implies that we can encode three information bits with each code word. The code is given in Table 9-4-2. Note that the first bit of each code word is a 0, whereas the last bit may be either 0 or 1. Consequently, the $d = 1$ constraint is satisfied when these code words are concatenated. This code has a rate $R_c = 3/5$. When compared with the capacity $C(1, \infty) = 0.6942$ obtained from Table 9-4-1, the code efficiency is 0.864, which is quite acceptable.

The code construction method described in the two examples above produces fixed-length (d, κ) codes that are *state-independent*. By state-independent, we mean that fixed-length code words can be concatenated without violating the (d, κ) constraints. In general, fixed-length state-independent (d, κ) codes require large block lengths, except in cases such as those in the examples above where d is small. Simpler (shorter-length) codes

TABLE 9-4-2 FIXED LENGTH $d = 1$, $\kappa = \infty$ CODE

Input data bits	Output coded sequence
0 0 0	0 0 0 0 0
0 0 1	0 0 0 0 1
0 1 0	0 0 0 1 0
0 1 1	0 0 1 0 0
1 0 0	0 0 1 0 1
1 0 1	0 1 0 0 0
1 1 0	0 1 0 0 1
1 1 1	0 1 0 1 0

are generally possible by allowing for state-dependence and for variable length code words. Below, we consider codes for which both the input blocks to the encoder and the output blocks may have variable length. For the code words to be uniquely decodable at the receiver, the variable-length code should satisfy the prefix condition, described in Chapter 3.

Example 9-4-5

A very simple uniquely decodable variable-length $d = 0$, $\kappa = 2$ code is

$$0 \rightarrow 01$$
$$10 \rightarrow 10$$
$$11 \rightarrow 11$$

The code in the above example has a fixed output block size but a variable input block size. In general, both the input and output blocks may be variable. The following example illustrates the latter case.

Example 9-4-6

Let us construct a (2, 7) variable block size code. The solution to this code construction is certainly not unique, nor is it trivial. We picked this example because the (2, 7) code has been widely used by IBM in many of its disk storage systems. The code is listed in Table 9-4-3. We observe that the input data blocks of 2, 3, and 4 bits are mapped into output data blocks of 4, 6, and 8 bits, respectively. Hence, the code rate is $R_c = 1/2$. Since this is the code rate for all code words, the code is called a *fixed-rate* code. This code has an efficiency of 0.5/0.5174 = 0.966. Note that this code satisfies the prefix condition.

TABLE 9-4-3 CODE BOOK FOR VARIABLE-LENGTH (2, 7) CODE

Input data bits	Output coded sequence
1 0	1 0 0 0
1 1	0 1 0 0
0 1 1	0 0 0 1 0 0
0 1 0	0 0 1 0 0 0
0 0 0	1 0 0 1 0 0
0 0 1 1	0 0 1 0 0 1 0 0
0 0 1 0	0 0 0 0 1 0 0 0

TABLE 9-4-4 ENCODER FOR (1,3) MILLER CODE

Input data bits	Output coded sequence
0	x 0
1	0 1

$x = 0$, if preceding input bit is 1
$x = 1$, if preceding input bit is 0

Another code that has been widely used in magnetic recording is the rate 1/2, $(d, \kappa) = (1, 3)$ code in Table 9-4-4. We observe that when the information bit is a 0, the first output bit is 1 if the previous input bit was 0, or a 0 if the previous input bit was a 1. When the information bit is a 1, the encoder output is 01. Decoding of this code is simple. The first bit of the two-bit block is redundant and may be discarded. The second bit is the information bit. This code is usually called the *Miller code*. We observe that this is a state-dependent code, which is described by the state diagram shown in Fig. 9-4-5. There are two states labeled S_1 and S_2 with transitions as shown in the figure. When the encoder is a state S_1, an input bit 1 results in the encoder staying in state S_1 and outputs 01. This is denoted as 1/01. If the input bit is a 0, the encoder enters state S_2 and outputs 00. This is denoted as 0/00. Similarly, if the encoder is in state S_2, an input bit 0 causes no transition and the encoder output is 10. On the other hand, if the input bit is a 1, the encoder enters state S_1 and outputs 01. Figure 9-4-6 shows the trellis for the Miller code.

The Mapping of Coded Bits into Signal Waveforms The output sequence from a (d, κ) encoder is mapped by the modulator into signal waveforms for transmission over the channel. If the binary digit 1 is mapped into a rectangular pulse of amplitude A and the binary digit 0 is mapped into a

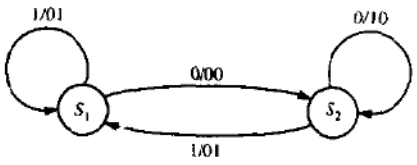


FIGURE 9-4-5 State diagrams for $d = 1$, $\kappa = 3$ (Miller) code.

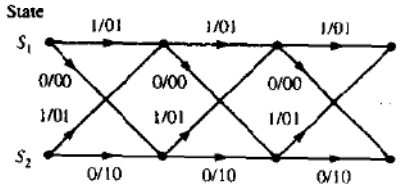


FIGURE 9-4-6 Trellis for $d = 1$, $\kappa = 3$ (Miller) code.

rectangular pulse of amplitude $-A$, the result is a (d, κ) coded NRZ modulated signal. Note that the duration of the rectangular pulses is $T_c = R_c/R_b = R_c T_b$, where R_b is the information (bit) rate into the encoder, T_b is the corresponding (uncoded) bit interval, and R_c is the code rate for the (d, κ) code.

When the (d, κ) code is a state-independent fixed-length code with code rate $R_c = k/n$, we may consider each n-bit block as generating one signal waveform of duration nT_c. Thus, we have $M = 2^k$ signal waveforms, one for each of the 2^k possible k-bit data blocks. These coded waveforms have the general form given by (4-3-6) and (4-3-38). In this case, there is no dependence between the transmission of successive waveforms.

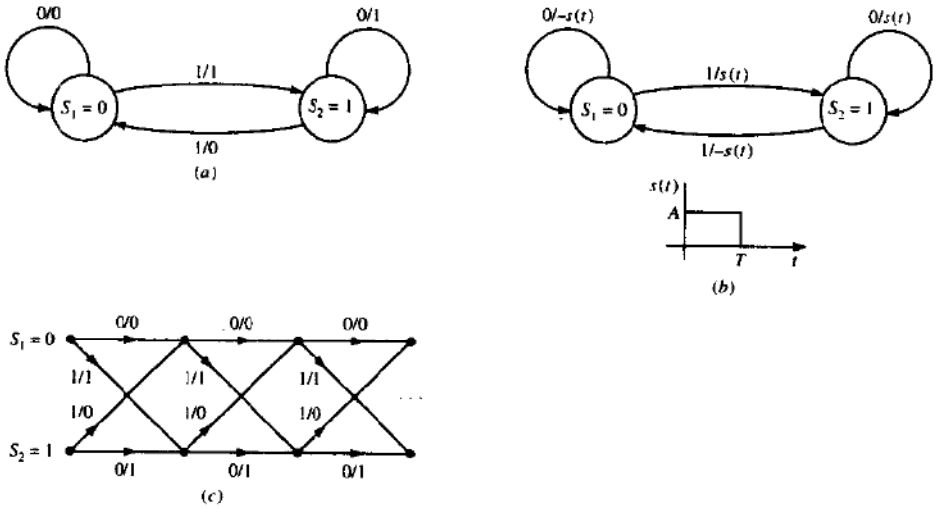
In contrast to the situation considered above, the modulation signal is no longer memoryless when NRZI is used and/or the (d, κ) code is state-dependent. Let us consider the effect of mapping the coded bits into an NRZI signal waveform.

Recall that the state dependence in the NRZI signal is due to the differential encoding of the information sequence. The differential encoding is a form of precoding, which is described mathematically as

$$p_k = d_k \oplus p_{k-1}$$

where $\{d_k\}$ is the binary sequence into the precoder, $\{p_k\}$ is the output binary sequence from the precoder, and $\oplus$ denotes modulo-2 addition. This encoding is characterized by the state diagram shown in Fig. 9-4-7(a). Then, the sequence $\{p_k\}$ is transmitted by NRZ. Thus, when $p_k = 1$, the modulator output is a rectangular pulse of amplitude A, and when $p_k = 0$, the modulator

FIGURE 9-4-7 State and trellis diagrams for NRZI signal.



output is a rectangular pulse of amplitude $-A$. When the signal waveforms are superimposed on the state diagram of Fig. 9-4-7(a), we obtain the corresponding state diagram shown in Fig. 9-4-7(b). The corresponding trellis is shown in Fig. 9-4-7(c).

When the output of a state-dependent (d, κ) encoder is followed by an NRZI modulator, we may simply combine the two-state diagrams into a single-state diagram for the (d, κ) code with precoding. A similar combination can be performed with the corresponding trellises. The following example illustrates the approach for the (1, 3) Miller code followed by NRZI modulation.

Example 9-4-7

Let us determine the state diagram of the combined (1, 3) Miller code followed by the precoding inherent in NRZI modulation. Since the (1, 3) Miller code has two states and the precoder has two states, the state diagram for the combined encoder has four states, which we denote as $(S_M, S_N) = (\sigma_1, s_1), (\sigma_1, s_2), (\sigma_2, s_1), (\sigma_2, s_2)$, where $S_M = \{\sigma_1, \sigma_2\}$ represents the two states of the Miller code and $S_N = \{s_1, s_2\}$ represents the two states of the precoder for NRZI. For each data input bit into the Miller encoder, we obtain two output bits which are then precoded to yield two precoded output bits. The resulting state diagram is shown in Fig. 9-4-8, where the first bit denotes the information bit into the Miller encoder and the next two bits represent the corresponding output of the precoder.

The trellis diagram for the Miller precoded sequence may be obtained directly from the combined state diagram or from a combination of the trellises of the two codes. The result of this combination is the four-state trellis, one stage of which is shown in Fig. 9-4-9.

It is left as an exercise for the reader to show that the four signal waveforms obtained by mapping each pair of bits of the Miller-precoded sequence into an

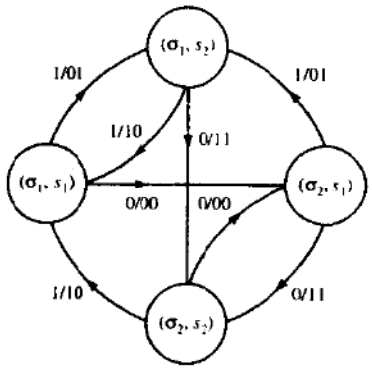


FIGURE 9-4-8 State diagram of the Miller code followed by the precoder.

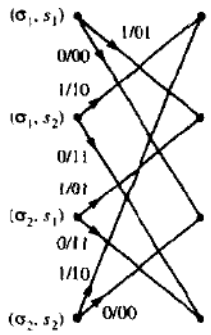


FIGURE 9-4-9 One stage of trellis diagram for the Miller code followed by the precoder.

NRZ signal are biorthogonal and that the resulting modulated signal waveform is identical to the delay modulation that was described in Section 4-3-2.

From the state diagram of a state-dependent runlength-limited code, one can obtain the transition probability matrix, as described in Section 4-3-2. Then, the power spectral density of the code may be determined, as shown in Section 4-4-3.

9-5 BIBLIOGRAPHICAL NOTES AND REFERENCES

The pioneering work on signal design for bandwidth-constrained channels was done by Nyquist (1928). The use of binary partial response signals was originally proposed by Lender (1963), and was later generalized by Kretzmer (1966). Other early work on problems dealing with intersymbol interference (ISI) and transmitter and receiver optimization with constraints on ISI was done by Gerst and Diamond (1961), Tufts (1965), Smith (1965), and Berger and Tufts (1967). "Faster than Nyquist" transmission has been studied by Mazo (1975) and Foschini (1984).

Modulation codes were also first introduced by Shannon (1948). Some of the early work on the construction of runlength-limited codes is found in the papers by Freiman and Wyner (1964), Gabor (1967), Franaszek (1968, 1969, 1970), Tang and Bahl (1970), and Jacoby (1977). More recent work is found in papers by Adler Coppersmith and Hassner (1983), and Karabed and Siegel (1991). The motivation for most of the work on runlength-limited codes was provided by applications to magnetic and optical recording. A well-written tutorial paper on runlength-limited codes has been published by Immink (1990).

PROBLEMS

9-1 A channel is said to be *distortionless* if the response $y(t)$ to an input $x(t)$ is $Kx(t - t_0)$, where K and t_0 are constants. Show that if the frequency response of the channel is $A(f)e^{j\theta(f)}$, where $A(f)$ and $\theta(f)$ are real, the necessary and

sufficient conditions for distortionless transmission are $A(f) = K$ and $\theta(f) = 2\pi f t_0 \pm n\pi$, $n = 0, 1, 2, \ldots$.

9-2 The raised-cosine spectral characteristic is given by (9-2-26).

a Show that the corresponding impulse response is

$$x(t) = \frac{\sin(\pi t/T)}{\pi t/T} \frac{\cos(\beta \pi t/T)}{1 - 4\beta^2 t^2/T^2}$$

b Determine the Hilbert transform of $x(t)$ when $\beta = 1$.

c Does $\hat{x}(t)$ possess the desirable properties of $x(t)$ that make it appropriate for data transmission? Explain.

d Determine the envolope of the SSB suppressed-carrier signal generated from $x(t)$.

9-3 **a** Show that (Poisson sum formula)

$$x(t) = \sum_{k=-\infty}^{\infty} g(t)h(t - kT) \Rightarrow X(f) = \frac{1}{T} \sum_{n=-\infty}^{\infty} H\left(\frac{n}{T}\right) G\left(f - \frac{n}{T}\right)$$

Hint: Make a Fourier-series expansion of the periodic factor

$$\sum_{k=-\infty}^{\infty} h(t - kT)$$

b Using the result in (a), verify the following versions of the Poisson sum:

$$\sum_{k=-\infty}^{\infty} h(kT) = \frac{1}{T} \sum_{n=-\infty}^{\infty} H\left(\frac{n}{T}\right) \tag{i}$$

$$\sum_{k=-\infty}^{\infty} h(t - kT) = \frac{1}{T} \sum_{n=-\infty}^{\infty} H\left(\frac{n}{T}\right) \exp\left(\frac{j2\pi nt}{T}\right) \tag{ii}$$

$$\sum_{k=-\infty}^{\infty} h(kT) \exp(-j2\pi kTf) = \frac{1}{T} \sum_{n=-\infty}^{\infty} H\left(f - \frac{n}{T}\right) \tag{iii}$$

c Derive the condition for no intersymbol interference (Nyquist criterion) by using the Poisson sum formula.

9-4 Suppose a digital communications system employs gaussian-shaped pulses of the form

$$x(t) = \exp(-\pi a^2 t^2)$$

To reduce the level of intersymbol interference to a relatively small amount, we impose the condition that $x(T) = 0.01$, where T is the symbol interval. The bandwidth W of the pulse $x(t)$ is defined as that value of W for which $X(W)/X(0) = 0.01$, where $X(f)$ is the Fourier transform of $x(t)$. Determine the value of W and compare this value to that of raised-cosine spectrum with 100% rolloff.

9-5 A band-limited signal having bandwidth W can be represented as

$$x(t) = \sum_{n=-\infty}^{\infty} x_n \frac{\sin[2\pi W(t - n/2W)]}{2\pi W(t - n/2W)}$$

a Determine the spectrum $X(f)$ and plot $|X(f)|$ for the following cases:

$$x_0 = 2, \quad x_1 = 1, \quad x_2 = -1, \quad x_n = 0, \quad n \neq 0, 1, 2 \tag{i}$$

$$x_{-1} = -1, \quad x_0 = 2, \quad x_1 = -1, \quad x_n = 0, \quad n \neq -1, 0, 1 \tag{ii}$$

b Plot $x(t)$ for these two cases.

c If these signals are used for binary signal transmission, determine the number of received levels possible at the sampling instants $t = nT = n/2W$, and the probabilities of occurrence of the received levels. Assume that the binary digits at the transmitter are equally probable.

9-6 A 4 kHz bandpass channel is to be used for transmission of data at a rate of 9600 bits/s. If $\frac{1}{2}N_0 = 10^{-10}$ W/Hz is the spectral density of the additive, zero-mean gaussian noise in the channel, design a QAM modulation and determine the average power that achieves a bit error probability of 10^{-6}. Use a signal pulse with a raised-cosine spectrum having a roll-off factor of at least 50%.

9-7 Determine the bit rate that can be transmitted through a 4 kHz voice-band telephone (bandpass) channel if the following modulation methods are used: (a) binary PAM; (b) four-phase PSK; (c) 8-point QAM; (d) binary orthogonal FSK, with noncoherent detection; (e) orthogonal four-FSK with noncoherent detection; (f) orthogonal 8-FSK with noncoherent detection. For (a)–(c), assume that the transmitter pulse shape has a raised-cosine spectrum with a 50% roll-off.

9-8 An ideal voice-band telephone line channel has a bandpass frequency response characteristic spanning the frequency range 600–3000 Hz.

a Design an $M = 4$ PSK (quadrature PSK or QPSK) system for transmitting data at a rate of 2400 bits/s and a carrier frequency $f_c = 1800$ Hz. For spectral shaping, use a raised-cosine frequency-response characteristic. Sketch a block diagram of the system and describe the functional operation of each block.

b Repeat (a) for a bit rate $R = 4800$ bits/s.

9-9 A voice-band telephone channel passes the frequencies in the band from 300 to 3300 Hz. It is desired to design a modem that transmits at a symbol rate of 2400 symbols/s, with the objective of achieving 9600 bits/s. Select an appropriate QAM signal constellation, carrier frequency, and the roll-off factor of a pulse with a raised cosine spectrum that utilizes the entire frequency band. Sketch the spectrum of the transmitted signal pulse and indicate the important frequencies.

9-10 A communication system for a voice-band (3 kHz) channel is designed for a received SNR at the detector of 30 dB when the transmitter power is $P_S = -3$ dBW. Determine the value of P_S if it is desired to expand the bandwidth of the system to 10 kHz, while maintaining the same SNR at the detector.

9-11 Show that a pulse having the raised cosine spectrum given by (9-2-26) satisfies the Nyquist criterion given by (9-2-13) for any value of the roll-off factor β.

9-12 Show that, for any value of β, the raised cosine spectrum given by (9-2-26) satisfies

$$\int_{-\infty}^{\infty} X_{rc}(f)\, df = 1$$

[*Hint:* Use the fact that $X_{rc}(f)$ satisfies the Nyquist criterion given by (9-2-13).]

9-13 The Nyquist criterion gives the necessary and sufficient condition for the spectrum $X(f)$ of the pulse $x(t)$ that yields zero ISI. Prove that for any pulse that is band-limited to $|f| < 1/T$, the zero-ISI condition is satisfied if Re $[X(f)]$, for $f > 0$, consists of a rectangular function plus an arbitrary odd function around $f = 1/2T$, and Im $[X(f)]$ is any arbitrary even function around $f = 1/2T$.

9-14 A voice-band telephone channel has a passband characteristic in the frequency range 300 Hz $< f <$ 3000 Hz.

a Select a symbol rate and a power efficient constellation size to achieve 9600 bits/s signal transmission.

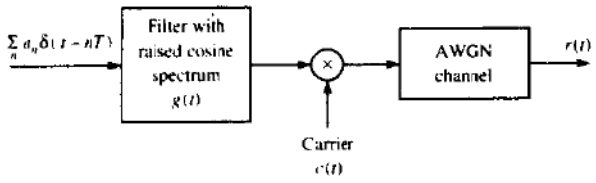


FIGURE P9-16

b If a square-root raised cosine pulse is used for the transmitter pulse $g(t)$, select the roll-off factor. Assume that the channel has an ideal frequency response characteristic.

9-15 Design an M-ary PAM system that transmits digital information over an ideal channel with bandwidth $W = 2400$ Hz. The bit rate is 14 400 bit/s. Specify the number of transmitted points, the number of received signal points using a duobinary signal pulse, and the required $\mathscr{E}_b$ to achieve an error probability of 10^{-6}. The additive noise is zero-mean gaussian with a power spectral density 10^{-4} W/Hz.

9-16 A binary PAM signal is generated by exciting a raised cosine roll-off filter with a 50% roll-off factor and is then DSB-SC amplitude-modulated on a sinusoidal carrier as illustrated in Fig. P9-16. The bit rate is 2400 bit/s.

a Determine the spectrum of the modulated binary PAM signal and sketch it.

b Draw the block diagram illustrating the optimum demodulator/detector for the received signal, which is equal to the transmitted signal plus additive white gaussian noise.

9-17 The elements of the sequence $\{a_n\}_{n=-\infty}^{\infty}$ are independent binary random variables taking values of ± 1 with equal probability. This data sequence is used to modulate the basic pulse $g(t)$ shown in Fig. P9-17(a). The modulated signal is

$$X(t) = \sum_{n=-\infty}^{+\infty} a_n g(t - nT)$$

a Find the power spectral density of $X(t)$.

b If $g_1(t)$ (shown in Fig. 9-17b) is used instead of $g(t)$, how would the power spectrum in (a) change?

c In (b) assume we want to have a null in the spectrum at $f = 1/3T$. This is done by a precoding of the form $b_n = a_n + \alpha a_{n-3}$. Find the α that provides the desired null.

d Is it possible to employ a precoding of the form $b_n = a_n + \sum_{i=1}^{N} \alpha_i a_{n-i}$ for some finite N such that the final power spectrum will be identical to zero for $1/3T \leq |f| \leq 1/2T$? If yes, how? If no, why? [*Hint:* Use properties of analytic functions.]

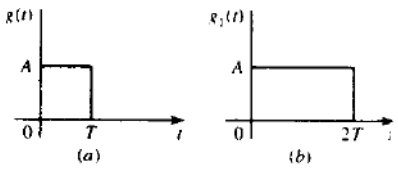


FIGURE P9-17

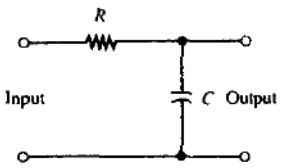


FIGURE P9-22

9-18 Consider the transmission of data via PAM over a voice-band telephone channel that has a bandwidth of 3000 Hz. Show how the symbol rate varies as a function of the excess bandwidth. In particular, determine the symbol rate for an excess bandwidth of 25%, 33%, 50%, 67%, 75%, and 100%.

9-19 The binary sequence 10010110010 is the input to a precoder whose output is used to modulate a duobinary transmitting filter. Construct a table as in Table 9-2-1 showing the precoded sequence, the transmitted amplitude levels, the received signal levels and the decoded sequence.

9-20 Repeat Problem 9-19 for a modified duobinary signal pulse.

9-21 A precoder for a partial response signal fails to work if the desired partial response at $n = 0$ is zero modulo M. For example, consider the desired response for $M = 2$:

$$x(nT) = \begin{cases} 2 & (n = 0) \\ 1 & (n = 1) \\ -1 & (n = 2) \\ 0 & (\text{otherwise}) \end{cases}$$

Show why this response cannot be precoded.

9-22 Consider the RC lowpass filter shown in Fig. P9-22, where $\tau = RC = 10^{-6}$.

a Determine and sketch the envelope (group) delay of the filter as a function of frequency.

b Suppose that the input to the filter is a lowpass signal of bandwidth $\Delta f = 1$ kHz. Determine the effect of the RC filter on this signal.

9-23 A microwave radio channel has a frequency response

$$C(f) = 1 + 0.3 \cos 2\pi fT$$

Determine the frequency response characteristic of the optimum transmitting and receiving filters that yield zero ISI at a rate of $1/T$ symbols/s and have a 50% excess bandwidth. Assume that the additive noise spectrum is flat.

9-24 $M = 4$ PAM modulation is used for transmitting at a bit rate of 9600 bit/s on a channel having a frequency response

$$C(f) = \frac{1}{1 + j(f/2400)}$$

for $|f| \leq 2400$, and $C(f) = 0$ otherwise. The additive noise is zero-mean, white Gaussian with power spectral density $\frac{1}{2}N_0$ W/Hz. Determine the (magnitude) frequency response characteristic of the optimum transmitting and receiving filters.

9-25 Determine the capacity of a $(0, 1)$ runlength-limited code. Compare its capacity with that of a $(1, \infty)$ code and explain the relationship.

9-26 A ternary signal format is designed for a channel that does not pass d.c. The

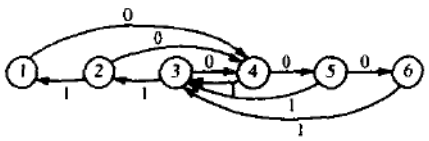


FIGURE P9-31

binary input information sequence is transmitted by mapping a 1 into either a positive pulse or a negative pulse, and a zero is transmitted by the absence of a pulse. Hence, for the transmission of 1s, the polarity of the pulses alternate. This is called an AMI (alternate mark inversion) code. Determine the capacity of the code.

9-27 Give an alternative description of the AMI code described in Problem 9-26 using the running digit sum (RDS) with the constraint that the RDS can take only the values 0 and +1.

9-28 (kBnT codes) From Problem 9-26, note that the AMI code is a "pseudo-ternary" code in that it transmits one bit per symbol using a ternary alphabet, which has the capacity of $\log_2 3 = 1.58$ bits. Such a code does not provide sufficient spectral shaping. Better spectral shaping is achieved by the class of block codes designated as kBnT, where k denotes the number of information bits and n denotes the number of ternary symbols per block. By selecting the largest k possible for each n, we obtain the following table:

k	n	Code
1	1	1B1T
3	2	3B2T
4	3	4B3T
6	4	6B4T

Determine the efficiency of these codes by computing the ratio of the code in bits/symbol divided by $\log_2 3$. Note that 1B1T is the AMI code.

9-29 This problem deals with the capacity of two (d, κ) codes.

a Determine the capacity of a (d, κ) code that has the following state transition matrix:

$$\mathbf{D} = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$$

b Repeat (a) for

$$\mathbf{D} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$$

c Comment on the differences between (a) and (b).

9-30 A simplified model of the telegraph code consists of two symbols (Blahut, 1990). A dot consists of one time unit of line closure followed by one time unit of line

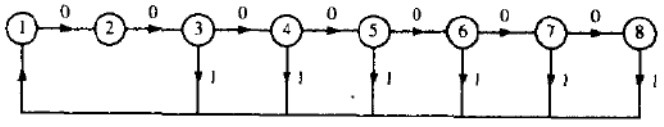


FIGURE P9-32

open. A dash consists of three units of line closure followed by one time unit of line open.

a Viewing this code as a constrained code with symbols of equal duration, give the constraints.

b Determine the state-transition matrix.

c Determine the capacity.

9-31 Determine the state-transition matrix for the runlength-constrained code described by the state diagram shown in Fig. P9-31. Sketch the corresponding trellis.

9-32 Determine the state-transition matrix for the $(2, 7)$ runlength-limited code specified by the state diagram shown in Fig. P9-32.

10

COMMUNICATION THROUGH BAND-LIMITED LINEAR FILTER CHANNELS

In Chapter 9, we focused on the design of the modulator and demodulator filters for band-limited channels. The design procedure was based on the assumption that the (ideal or non-ideal) channel response characteristic $C(f)$ was known a priori. However, in practical digital communications systems that are designed to transmit at high speed through band-limited channels, the frequency response $C(f)$ of the channel is not known with sufficient precision to design optimum filters for the modulator and demodulator. For example, in digital communication over the dial-up telephone network, the communication channel will be different every time we dial a number, because the channel route will be different. This is an example of a channel whose characteristics are unknown a priori. There are other types of channels, e.g., wireless channels such as radio channels and underwater acoustic channels, whose frequency response characteristics are time-variant. For such channels, it is not possible to design optimum fixed demodulation filters.

In this chapter, we consider the problem of receiver design in the presence of channel distortion, which is not known a priori, and AWGN. The channel distortion results in intersymbol interference, which, if left uncompensated, causes high error rates. The solution to the ISI problem is to design a receiver that employs a means for compensating or reducing the ISI in the received signal. The compensator for the ISI is called an *equalizer*.

Three types of equalization methods are treated in this chapter. One is based on the maximum-likelihood (ML) sequence detection criterion, which is optimum from a probability of error viewpoint. A second equalization method is based on the use of a linear filter with adjustable coefficients. The third equalization method that is described exploits the use of previous detected

symbols to suppress the ISI in the present symbol being detected, and it is called *decision-feedback equalization*. We begin with the derivation of the optimum detector for channels with ISI.

10-1 OPTIMUM RECEIVER FOR CHANNELS WITH ISI AND AWGN

In this section, we derive the structure of the optimum demodulator and detector for digital transmission through a nonideal, band-limited channel with additive gaussian noise. We begin with the transmitted (equivalent lowpass) signal given by (9-2-1). The received (equivalent lowpass) signal is expressed as

$$r_l(t) = \sum_n I_n h(t - nT) + z(t) \tag{10-1-1}$$

where $h(t)$ represents the response of the channel to the input signal pulse $g(t)$ and $z(t)$ represents the additive white gaussian noise.

First we demonstrate that the optimum demodulator can be realized as a filter matched to $h(t)$, followed by a sampler operating at the symbol rate $1/T$ and a subsequent processing algorithm for estimating the information sequence $\{I_n\}$ from the sample values. Consequently, the samples at the output of the matched filter are sufficient for the estimation of the sequence $\{I_n\}$.

10-1-1 Optimum Maximum-Likelihood Receiver

Let us expand the received signal $r_l(t)$ in the series

$$r_l(t) = \lim_{N \to \infty} \sum_{k=1}^{N} r_k f_k(t) \tag{10-1-2}$$

where $\{f_k(t)\}$ is a complete set of orthonormal functions and $\{r_k\}$ are the observable random variables obtained by projecting $r_l(t)$ onto the set $\{f_k(t)\}$. It is easily shown that

$$r_k = \sum_n I_n h_{kn} + z_k, \qquad k = 1, 2, \ldots \tag{10-1-3}$$

where h_{kn} is the value obtained from projecting $h(t - nT)$ onto $f_k(t)$, and z_k is the value obtained from projecting $z(t)$ onto $f_k(t)$. The sequence $\{z_k\}$ is gaussian with zero mean and covariance

$$\tfrac{1}{2}E(z_k^* z_m) = N_0 \delta_{km} \tag{10-1-4}$$

The joint probability density function of the random variables

$\mathbf{r}_N \equiv [r_1\ r_2\ \ldots\ r_N]$ conditioned on the transmitted sequence $\mathbf{I}_p \equiv [I_1\ I_2\ \ldots\ I_p]$, where $p \leq N$, is

$$p(\mathbf{r}_N\, 3\mathbf{I}_p) = \left(\frac{1}{2\pi N_0}\right)^N \exp\left(-\frac{1}{2N_0}\sum_{k=1}^{N}\left|r_k - \sum_n I_n h_{kn}\right|^2\right) \tag{10-1-5}$$

In the limit as the number N of observable random variables approaches infinity, the logarithm of $p(\mathbf{r}_N\, 3\mathbf{I}_p)$ is proportional to the metrics $PM(\mathbf{I}_p)$, defined as

$$\begin{aligned} PM(\mathbf{I}_p) &= -\int_{-\infty}^{\infty}\left|r_l(t) - \sum_n I_n h(t-nT)\right|^2 dt \\ &= -\int_{-\infty}^{\infty}|r_l(t)|^2\,dt + 2\,\mathrm{Re}\sum_n\left[I_n^*\int_{-\infty}^{\infty} r_l(t)h^*(t-nT)\,dt\right] \\ &\quad -\sum_n\sum_m I_n^* I_m\int_{-\infty}^{\infty} h^*(t-nT)h(t-mT)\,dt \end{aligned} \tag{10-1-6}$$

The maximum-likelihood estimates of the symbols $I_1, I_2, \ldots, I_p$ are those that maximize this quantity. Note, however, that the integral of $|r_l(t)|^2$ is common to all metrics, and, hence, it may be discarded. The other integral involving $r(t)$ gives rise to the variables

$$y_n \equiv y(nT) = \int_{-\infty}^{\infty} r_l(t)h^*(t-nT)\,dt \tag{10-1-7}$$

These variables can be generated by passing $r(t)$ through a filter matched to $h(t)$ and sampling the output at the symbol rate $1/T$. The samples $\{y_n\}$ form a set of sufficient statistics for the computation of $PM(\mathbf{I}_p)$ or, equivalently, of the correlation metrics

$$CM(\mathbf{I}_p) = 2\,\mathrm{Re}\left(\sum_n I_n^* y_n\right) - \sum_n\sum_m I_n^* I_m x_{n-m} \tag{10-1-8}$$

where, by definition, $x(t)$ is the response of the matched filter to $h(t)$ and

$$x_n \equiv x(nT) = \int_{-\infty}^{\infty} h^*(t)h(t+nT)\,dt \tag{10-1-9}$$

Hence, $x(t)$ represents the output of a filter having an impulse response $h^*(-t)$ and an excitation $h(t)$. In other words, $x(t)$ represents the autocorrelation function of $h(t)$. Consequently, $\{x_n\}$ represents the samples of the autocorrelation function of $h(t)$, taken periodically at $1/T$. We are not particularly concerned with the noncausal characteristic of the filter matched to $h(t)$, since, in practice, we can introduce a sufficiently large delay to ensure causality of the matched filter.

If we substitute for $r_l(t)$ in (10-1-7) using (10-1-1), we obtain

$$y_k = \sum_n I_n x_{k-n} + \nu_k \tag{10-1-10}$$

where ν_k denotes the additive noise sequence of the output of the matched filter, i.e.,

$$\nu_k = \int_{-\infty}^{\infty} z(t)h^*(t - kT)\,dt \tag{10-1-11}$$

The output of the demodulator (matched filter) at the sampling instants is corrupted by ISI as indicated by (10-1-10). In any practical system, it is reasonable to assume that the ISI affects a finite number of symbols. Hence, we may assume that $x_n = 0$ for $|n| > L$. Consequently, the ISI observed at the output of the demodulator may be viewed as the output of a finite state machine. This implies that the channel output with ISI may be represented by a trellis diagram, and the maximum-likelihood estimate of the information sequence $(I_1, I_2, \ldots, I_p)$ is simply the most probable path through the trellis given the received demodulator output sequence $\{y_n\}$. Clearly, the Viterbi algorithm provides an efficient means for performing the trellis search.

The metrics that are computed for the MLSE of the sequence $\{I_k\}$ are given by (10-1-8). It can be seen that these metrics can be computed recursively in the Viterbi algorithm, according to the relation

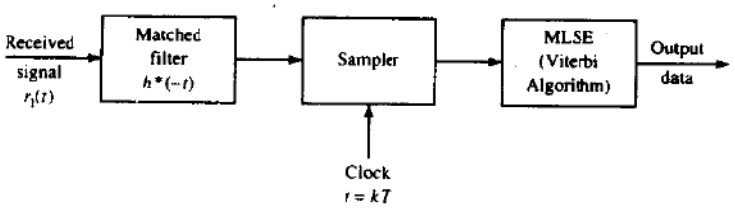
$$CM_n(\mathbf{I}_n) = CM_{n-1}(\mathbf{I}_{n-1}) + \text{Re}\left[I_n^*\left(2y_n - x_0 I_n - 2\sum_{m=1}^{L} x_m I_{n-m}\right)\right] \tag{10-1-12}$$

Figure 10-1-1 illustrates the block diagram of the optimum receiver for an AWGN channel with ISI.

10-1-2 A Discrete-Time Model for a Channel with ISI

In dealing with band-limited channels that result in ISI, it is convenient to develop an equivalent discrete-time model for the analog (continuous-time) system. Since the transmitter sends discrete-time symbols at a rate $1/T$ symbols/s and the sampled output of the matched filter at the receiver is also a discrete-time signal with samples occurring at a rate $1/T$ per second, it follows that the cascade of the analog filter at the transmitter with impulse response $g(t)$, the channel with impulse response $c(t)$, the matched filter at the receiver with impulse response $h^*(-t)$, and the sampler can be represented by

FIGURE 10-1-1 Optimum receiver for an AWGN channel with ISI.



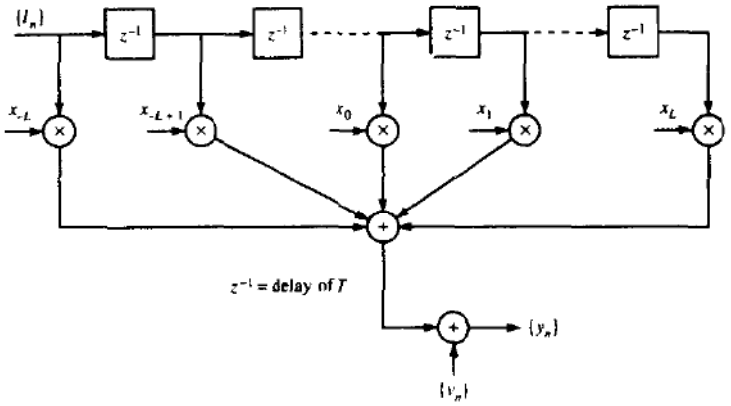


FIGURE 10-1-2 Equivalent discrete-time model of channel with intersymbol interference.

an equivalent discrete-time transversal filter having tap gain coefficients $\{x_k\}$. Consequently, we have an equivalent discrete-time transversal filter that spans a time interval of $2LT$ seconds. Its input is the sequence of information symbols $\{I_k\}$ and its output is the discrete-time sequence $\{y_k\}$ given by (10-1-10). The equivalent discrete-time model is shown in Fig. 10-1-2.

The major difficulty with this discrete-time model occurs in the evaluation of performance of the various equalization or estimation techniques that are discussed in the following sections. The difficulty is caused by the correlations in the noise sequence $\{v_k\}$ at the output of the matched filter. That is, the set of noise variables $\{v_k\}$ is a gaussian-distributed sequence with zero mean and autocorrelation function (see Problem 10-5)

$$\tfrac{1}{2}E(v_i^* v_j) = \begin{cases} N_0 x_{k-j} & (|k-j| \leq L) \\ 0 & (\text{otherwise}) \end{cases} \tag{10-1-13}$$

Hence, the noise sequence is correlated unless $x_k = 0$, $k \neq 0$. Since it is more convenient to deal with the white noise sequence when calculating the error rate performance, it is desirable to whiten the noise sequence by further filtering the sequence $\{y_k\}$. A discrete-time noise-whitening filter is determined as follows.

Let $X(z)$ denote the (two-sided) z transform of the sampled autocorrelation function $\{x_k\}$, i.e.,

$$X(z) = \sum_{k=-L}^{L} x_k z^{-k} \tag{10-1-14}$$

Since $x_k = x_{-k}^*$, it follows that $X(z) = X^*(z^{-1})$ and the $2L$ roots of $X(z)$ have the symmetry that if ρ is a root, $1/\rho^*$ is also a root. Hence, $X(z)$ can be factored and expressed as

$$X(z) = F(z)F^*(z^{-1}) \tag{10-1-15}$$

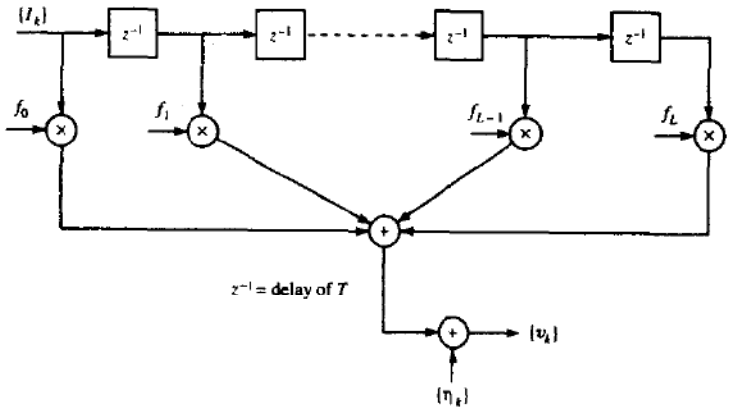
where $F(z)$ is a polynomial of degree L having the roots $\rho_1, \rho_2, \ldots, \rho_L$ and $F^*(z^{-1})$ is a polynomial of degree L having the roots $1/\rho_1^*, 1/\rho_2^*, \ldots, 1/\rho_L^*$. Then an appropriate noise-whitening filter has a z transform $1/F^*(z^{-1})$. Since there are 2^L possible choices for the roots of $F^*(z^{-1})$, each choice resulting in a filter characteristic that is identical in magnitude but different in phase from other choices of the roots, we propose to choose the unique $F^*(z^{-1})$ having minimum phase, i.e., the polynomial having all its roots inside the unit circle. Thus, when all the roots of $F^*(z^{-1})$ are inside the unit circle, $1/F^*(z^{-1})$ is a physically realizable, stable, recursive discrete-time filter.† Consequently, passage of the sequence $\{y_k\}$ through the digital filter $1/F^*(z^{-1})$ results in an output sequence $\{v_k\}$ that can be expressed as

$$v_k = \sum_{n=0}^{L} f_n I_{k-n} + \eta_k \tag{10-1-16}$$

where $\{\eta_k\}$ is a white gaussian noise sequence and $\{f_k\}$ is a set of tap coefficients of an equivalent discrete-time transversal filter having a transfer function $F(z)$. In general, the sequence $\{v_k\}$ is complex-valued.

In summary, the cascade of the transmitting filter $g(t)$, the channel $c(t)$, the matched filter $h^*(-t)$, the sampler, and the discrete-time noise-whitening filter $1/F^*(z^{-1})$ can be represented as an equivalent discrete-time transversal filter having the set $\{f_k\}$ as its tap coefficients. The additive noise sequence $\{\eta_k\}$ corrupting the output of the discrete-time transversal filter is a white gaussian noise sequence having zero mean and variance N_0. Figure 10-1-3 illustrates the model of the equivalent discrete-time system with white noise. We refer to this model as the *equivalent discrete-time white noise filter model.*

FIGURE 10-1-3 Equivalent discrete-time model of intersymbol interference channel with WGN.



†By removing the stability condition, we can also show $F^*(z^{-1})$ to have roots on the unit circle.

Example 10-1-1

Suppose that the transmitter signal pulse $g(t)$ has duration T and unit energy and the received signal pulse is $h(t) = g(t) + ag(t - T)$. Let us determine the equivalent discrete-time white-noise filter model. The sample autocorrelation function is given by

$$x_k = \begin{cases} a^* & (k = -1) \\ 1 + |a|^2 & (k = 0) \\ a & (k = 1) \end{cases} \tag{10-1-17}$$

The z transform of x_k is

$$\begin{aligned} X(z) &= \sum_{k=-1}^{1} x_k z^{-k} \\ &= a^* z + (1 + |a|^2) + a z^{-1} \\ &= (a z^{-1} + 1)(a^* z + 1) \end{aligned} \tag{10-1-18}$$

Under the assumption that $|a| > 1$, one chooses $F(z) = az^{-1} + 1$, so that the equivalent transversal filter consists of two taps having tap gain coefficients $f_0 = 1$, $f_1 = a$. Note that the correlation sequence $\{x_k\}$ may be expressed in terms of the $\{f_n\}$ as

$$x_k = \sum_{n=0}^{L-k} f_n^* f_{n+k}, \qquad k = 0, 1, 2, \ldots, L \tag{10-1-19}$$

When the channel impulse response is changing slowly with time, the matched filter at the receiver becomes a time-variable filter. In this case, the time variations of the channel/matched-filter pair result in a discrete-time filter with time-variable coefficients. As a consequence, we have time-variable intersymbol interference effects, which can be modeled by the filter illustrated in Fig. 10-1-3, where the tap coefficients are slowly varying with time.

The discrete-time white noise linear filter model for the intersymbol interference effects that arise in high-speed digital transmission over nonideal band-limited channels will be used throughout the remainder of this chapter in our discussion of compensation techniques for the interference. In general, the compensation methods are called *equalization techniques* or *equalization algorithms*.

10-1-3 The Viterbi Algorithm for the Discrete-Time White Noise Filter Model

MLSE of the information sequence $\{I_k\}$ is most easily described in terms of the received sequence $\{v_k\}$ at the output of the whitening filter. In the presence of

intersymbol interference that spans $L + 1$ symbols (L interfering components), the MLSE criterion is equivalent to the problem of estimating the state of a discrete-time finite-state machine. The finite-state machine in this case is the equivalent discrete-time channel with coefficients $\{f_k\}$, and its state at any instant in time is given by the L most recent inputs, i.e., the state at time k is

$$S_k = (I_{k-1}, I_{k-2}, \ldots, I_{k-L}) \tag{10-1-20}$$

where $I_k = 0$ for $k \leq 0$. Hence, if the information symbols are M-ary, the channel filter has M^L states. Consequently, the channel is described by an M^L-state trellis and the Viterbi algorithm may be used to determine the most probable path through the trellis.

The metrics used in the trellis search are akin to the metrics used in soft-decision decoding of convolutional codes. In brief, we begin with the samples $v_1, v_2, \ldots, v_{L+1}$, from which we compute the M^{L+1} metrics

$$\sum_{k=1}^{L+1} \ln p(v_k \mid I_k, I_{k-1}, \ldots, I_{k-L}) \tag{10-1-21}$$

The M^{L+1} possible sequences of $I_{L+1}, I_L, \ldots, I_2, I_1$ are subdivided into M^L groups corresponding to the M^L states $(I_{L+1}, I_L, \ldots, I_2)$. Note that the M sequences in each group (state) differ in I_1 and correspond to the paths through the trellis that merge at a single node. From the M sequences in each of the M^L states, we select the sequence with the largest probability (with respect to I_1) and assign to the surviving sequence the metric

$$\begin{aligned} PM_1(\mathbf{I}_{L+1}) &\equiv PM_1(I_{L+1}, I_L, \ldots, I_2) \\ &= \max_{I_1} \sum_{k=1}^{L+1} \ln p(v_k \mid I_k, I_{k-1}, \ldots, I_{k-L}) \end{aligned} \tag{10-1-22}$$

The $M - 1$ remaining sequences from each of the M^L groups are discarded. Thus, we are left with M^L surviving sequences and their metrics.

Upon reception of v_{L+2}, the M^L surviving sequences are extended by one stage, and the corresponding M^{L+1} probabilities for the extended sequences are computed using the previous metrics and the new increment, which is $\ln p(v_{L+2} \mid I_{L+2}, I_{L+1}, \ldots, I_2)$. Again, the M^{L+1} sequences are subdivided into M^L groups corresponding to the M^L possible states $(I_{L+2}, \ldots, I_3)$ and the most probable sequence from each group is selected, while the other $M - 1$ sequences are discarded.

The procedure described continues with the reception of subsequent signal samples. In general, upon reception of v_{L+k}, the metrics†

$$PM_k(\mathbf{I}_{L+k}) = \max_{I_k} [\ln p(v_{L+k} \mid I_{L+k}, \ldots, I_k) + PM_{k-1}(\mathbf{I}_{L+k-1})] \tag{10-1-23}$$

†We observe that the metrics $PM_k(\mathbf{I})$ are simply related to the euclidean distance metrics $DM_k(\mathbf{I})$ when the additive noise is gaussian.

that are computed give the probabilities of the M^L surviving sequences. Thus, as each signal sample is received, the Viterbi algorithm involves first the computation of the M^{L+1} probabilities

$$\ln p(v_{L+k} \mid I_{L+k}, \ldots, I_k) + PM_{k-1}(\mathbf{I}_{L+k-1}) \tag{10-1-24}$$

corresponding to the M^{L+1} sequences that form the continuations of the M^L surviving sequences from the previous stage of the process. Then the M^{L+1} sequences are subdivided into M^L groups, with each group containing M sequences that terminate in the same set of symbols $I_{L+k}, \ldots, I_{k+1}$ and differ in the symbol I_k. From each group of M sequences, we select the one having the largest probability as indicated by (10-1-23), while the remaining $M-1$ sequences are discarded. Thus, we are left again with M^L sequences having the metrics $PM_k(\mathbf{I}_{L+k})$.

As indicated previously, the delay in detecting each information symbol is variable. In practice, the variable delay is avoided by truncating the surviving sequences to the q most recent symbols, where $q \gg L$, thus achieving a fixed delay. In the case that the M^L surviving sequences at time k disagree on the symbol I_{k-q}, the symbol in the most probable sequence may be chosen. The loss in performance resulting from this suboptimum decision procedure is negligible if $q \geq 5L$.

Example 10-1-2

For illustrative purposes, suppose that a duobinary signal pulse is employed to transmit four-level ($M=4$) PAM. Thus, each symbol is a number selected from the set $\{-3, -1, 1, 3\}$. The controlled intersymbol interference in this partial response signal is represented by the equivalent discrete-time channel model shown in Fig. 10-1-4. Suppose we have received v_1 and v_2, where

$$\begin{aligned} v_1 &= I_1 + \eta_1 \\ v_2 &= I_2 + I_1 + \eta_2 \end{aligned} \tag{10-1-25}$$

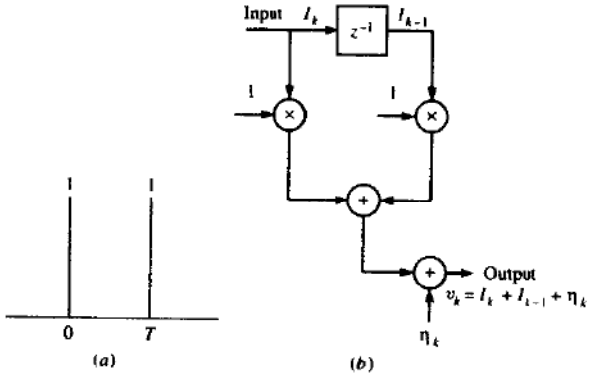


FIGURE 10-1-4 Equivalent discrete-time model for intersymbol interference resulting from a duobinary pulse.

and $\{\eta_i\}$ is a sequence of statistically independent zero-mean gaussian noise. We may now compute the 16 metrics

$$PM_1(I_2, I_1) = -\sum_{k=1}^{2}\left(v_k - \sum_{j=0}^{1} I_{k-j}\right)^2, \qquad I_1, I_2 = \pm 1, \pm 3 \qquad (10\text{-}1\text{-}26)$$

where $I_k = 0$ for $k \leq 0$.

Note that any subsequently received signals $\{v_i\}$ do not involve I_1. Hence, at this stage, we may discard 12 of the 16 possible pairs $\{I_1, I_2\}$. This step is illustrated by the tree diagram shown in Fig. 10-1-5. In other words, after computing the 16 metrics corresponding to the 16 paths in the tree diagram,

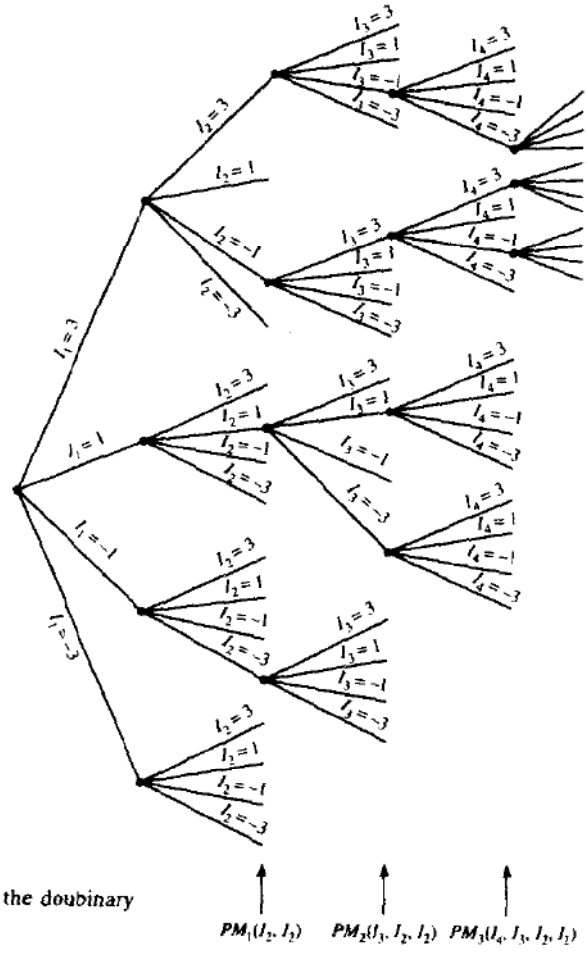


FIGURE 10-1-5 Tree diagram for Viterbi decoding of the doubinary pulse.

we discard three out of the four paths that terminate with $I_2 = 3$ and save the most probable of these four. Thus, the metric for the surviving path is

$$PM_1(I_2 = 3, I_1) = \max_{I_1} \left[- \sum_{k=1}^{2} \left(v_k - \sum_{j=0}^{1} I_{k-j} \right)^2 \right]$$

The process is repeated for each set of four paths terminating with $I_2 = 1$, $I_2 = -1$, and $I_2 = -3$. Thus four paths and their corresponding metrics survive after v_1 and v_2 are received.

When v_3 is received, the four paths are extended as shown in Fig. 10-1-5, to yield 16 paths and 16 corresponding metrics, given by

$$PM_2(I_3, I_2, I_1) = PM_1(I_2, I_1) - \left(v_3 - \sum_{j=0}^{1} I_{3-j} \right)^2 \tag{10-1-27}$$

Of the four paths terminating with the $I_3 = 3$, we save the most probable. This procedure is again repeated for $I_3 = 1$, $I_3 = -1$, and $I_3 = -3$. Consequently, only four paths survive at this stage. The procedure is then repeated for each subsequently received signal v_k for $k > 3$.

10-1-4 Performance of MLSE for Channels with ISI

We shall now determine the probability of error for MLSE of the received information sequence when the information is transmitted via PAM and the additive noise is gaussian. The similarity between a convolutional code and a finite-duration intersymbol interference channel implies that the method for computing the error probability for the latter carries over from the former. In particular, the method for computing the performance of soft-decision decoding of a convolutional code by means of the Viterbi algorithm, described in Section 8-2-3, applies with some modification.

In PAM signaling with additive gaussian noise and intersymbol interference, the metrics used in the Viterbi algorithm may be expressed as in (10-1-23), or equivalently, as

$$PM_{k-L}(\mathbf{I}_k) = PM_{k-L-1}(\mathbf{I}_{k-1}) - \left(v_k - \sum_{j=0}^{L} f_j I_{k-j} \right)^2 \tag{10-1-28}$$

where the symbols $\{I_n\}$ may take the values $\pm d, \pm 3d, \ldots, \pm(M-1)d$, and $2d$ is the distance between successive levels. The trellis has M^L states, defined at time k as

$$S_k(I_{k-1}, I_{k-2}, \ldots, I_{k-L}) \tag{10-1-29}$$

Let the estimated symbols from the Viterbi algorithm be denoted by $\{\bar{I}_n\}$ and the corresponding estimated state at time k by

$$\bar{S}_k = (\bar{I}_{k-1}, \bar{I}_{k-2}, \ldots, \bar{I}_{k-L}) \tag{10-1-30}$$

Now suppose that the estimated path through the trellis diverges from the correct path at time k and remerges with the correct path at time $k+l$. Thus, $\tilde{S}_k = S_k$ and $\tilde{S}_{k+l} = S_{k+l}$, but $\tilde{S}_m \neq S_m$ for $k < m < k+l$. As in a convolutional code, we call this an *error event*. Since the channel spans $L+1$ symbols, it follows that $l \geq L+1$.

For such an error event, we have $\tilde{I}_k \neq I_k$ and $\tilde{I}_{k+l-L-1} \neq I_{k+l-L-1}$, but $\tilde{I}_m = I_m$ for $k-L \leq m \leq k-1$ and $k+l-L \leq m \leq k+l-1$. It is convenient to define an error vector $\boldsymbol{\varepsilon}$ corresponding to this error event as

$$\boldsymbol{\varepsilon} = [\varepsilon_k \quad \varepsilon_{k+1} \quad \cdots \quad \varepsilon_{k+l-L-1}] \tag{10-1-31}$$

where the components of $\boldsymbol{\varepsilon}$ are defined as

$$\varepsilon_j = \frac{1}{2d}(I_j - \tilde{I}_j), \qquad j = k, k+1, \ldots, k+l-L-1 \tag{10-1-32}$$

The normalization factor of $2d$ in (10-1-32) results in elements ε_j that take on the values $\pm 1, \pm 2, \pm 3, \ldots, \pm(M-1)$. Moreover, the error vector is characterized by the properties that $\varepsilon_k \neq 0$, $\varepsilon_{k+l-L-1} \neq 0$, and there is no sequence of L consecutive elements that are zero. Associated with the error vector in (10-1-31) is the polynomial of degree $l-L-1$,

$$\varepsilon(z) = \varepsilon_k + \varepsilon_{k+1}z^{-1} + \varepsilon_{k+2}z^{-2} + \ldots + \varepsilon_{k+l-L-1}z^{-(l-L-1)} \tag{10-1-33}$$

We wish to determine the probability of occurrence of the error event that begins at time k and is characterized by the error vector $\boldsymbol{\varepsilon}$ given in (10-1-31), or, equivalently, by the polynomial given in (10-1-33). To accomplish this, we follow the procedure developed by Forney (1972). Specifically, for the error event $\boldsymbol{\varepsilon}$ to occur, the following three subevents E_1, E_2, and E_3 must occur:

E_1: at time k, $\tilde{S}_k = S_k$;

E_2: the information symbols $I_k, I_{k+1}, \ldots, I_{k+l-L-1}$ when added to the scaled error sequence $2d(\varepsilon_k, \varepsilon_{k+1}, \ldots, \varepsilon_{k+1-L-1})$ must result in an allowable sequence, i.e., the sequence $\tilde{I}_k, \tilde{I}_{k+1}, \ldots, \tilde{I}_{k+l-L-1}$ must have values selected from $\pm d, \pm 3d, \pm \cdots \pm (M-1)d$;

E_3: for $k \leq m < k+l$, the sum of the branch metrics of the estimated path exceed the sum of the branch metrics of the correct path.

The probability of occurrence of E_3 is

$$P(E_3) = P\left[\sum_{i=k}^{k+l-1}\left(v_i - \sum_{j=0}^{L} f_j \tilde{I}_{i-j}\right)^2 < \sum_{i=k}^{k+l-1}\left(v_i - \sum_{j=0}^{L} f_j I_{i-j}\right)^2\right] \tag{10-1-34}$$

But

$$v_i = \sum_{j=0}^{L} f_j I_{i-j} + \eta_i \tag{10-1-35}$$

where $\{\eta_i\}$ is a real-valued white gaussian noise sequence. Substitution of (10-1-35) into (10-1-34) yields

$$P(E_3) = P\left[\sum_{i=k}^{k+l-1}\left(\eta_i + 2d\sum_{j=0}^{L} f_j\varepsilon_{i-j}\right)^2 < \sum_{i=k}^{k+l-1}\eta_i^2\right]$$
$$= P\left[4d\sum_{i=k}^{k+l-1}\eta_i\left(\sum_{j=0}^{L} f_j\varepsilon_{i-j}\right) < -4d^2\sum_{i=k}^{k+l-1}\left(\sum_{j=0}^{L} f_j\varepsilon_{i-j}\right)^2\right] \quad (10\text{-}1\text{-}36)$$

where $\varepsilon_j = 0$ for $j < k$ and $j > k + l - L - 1$. If we define

$$\alpha_i = \sum_{j=0}^{L} f_j\varepsilon_{i-j} \quad (10\text{-}1\text{-}37)$$

then (10-1-36) may be expressed as

$$P(E_3) = P\left(\sum_{i=k}^{k+l-1}\alpha_i\eta_i < -d\sum_{i=k}^{k+l-1}\alpha_i^2\right) \quad (10\text{-}1\text{-}38)$$

where the factor of $4d$ common to both terms has been dropped. Now (10-1-38) is just the probability that a linear combination of statistically independent guassian random variables is less than some negative number. Thus

$$P(E_3) = Q\left(\sqrt{\frac{2d^2\sum_{i=k}^{k+l-1}\alpha_i^2}{N_0}}\right) \quad (10\text{-}1\text{-}39)$$

For convenience, we define

$$\delta^2(\boldsymbol{\varepsilon}) = \sum_{i=k}^{k+l-1}\alpha_i^2 = \sum_{i=k}^{k+l-1}\left(\sum_{j=0}^{L} f_j\varepsilon_{i-j}\right)^2 \quad (10\text{-}1\text{-}40)$$

where $\varepsilon_j = 0$ for $j < k$ and $j > k + l - L - 1$. Note that the $\{\alpha_i\}$ resulting from the convolution of $\{f_i\}$ with $\{\varepsilon_j\}$ are the coefficients of the polynomial

$$\alpha(z) = F(z)\varepsilon(z)$$
$$= \alpha_k + \alpha_{k+1}z^{-1} + \ldots + \alpha_{k+l-1}z^{-(l-1)} \quad (10\text{-}1\text{-}41)$$

Furthermore, $\delta^2(\boldsymbol{\varepsilon})$ is simply equal to the coefficient of z^0 in the polynomial

$$\alpha(z)\alpha(z^{-1}) = F(z)F(z^{-1})\varepsilon(z)\varepsilon(z^{-1})$$
$$= X(z)\varepsilon(z)\varepsilon(z^{-1}) \quad (10\text{-}1\text{-}42)$$

We call $\delta^2(\boldsymbol{\varepsilon})$ the *euclidean weight* of the error event $\boldsymbol{\varepsilon}$.

An alternative method for representing the result of convolving $\{f_j\}$ with $\{\varepsilon_j\}$ is the matrix form

$$\boldsymbol{\alpha} = \mathbf{ef}$$

where $\boldsymbol{\alpha}$ is an l-dimensional vector, $\mathbf{f}$ is an $(L+1)$-dimensional vector, and $\mathbf{e}$ is an $l \times (L+1)$ matrix, defined as

$$\boldsymbol{\alpha} = \begin{bmatrix} \alpha_k \\ \alpha_{k+1} \\ \vdots \\ \alpha_{k+l-1} \end{bmatrix} \qquad \mathbf{f} = \begin{bmatrix} f_0 \\ f_1 \\ \vdots \\ f_L \end{bmatrix}$$

$$\mathbf{e} = \begin{bmatrix} \varepsilon_k & 0 & 0 & \cdots & 0 & \cdots & 0 \\ \varepsilon_{k+1} & \varepsilon_k & 0 & \cdots & 0 & \cdots & 0 \\ \varepsilon_{k+2} & \varepsilon_{k+1} & \varepsilon_k & \cdots & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots & & \vdots \\ \varepsilon_{k+l-1} & \cdots & \cdots & \cdots & \cdots & \cdots & \varepsilon_{k+l-L-1} \end{bmatrix} \tag{10-1-43}$$

Then

$$\begin{aligned} \delta^2(\boldsymbol{\varepsilon}) &= \boldsymbol{\alpha}'\boldsymbol{\alpha} \\ &= \mathbf{f}'\mathbf{e}'\mathbf{ef} \\ &= \mathbf{f}'\mathbf{Af} \end{aligned} \tag{10-1-44}$$

where $\mathbf{A}$ is an $(L+1) \times (L+1)$ matrix of the form

$$\mathbf{A} = \mathbf{e}'\mathbf{e} = \begin{bmatrix} \beta_0 & \beta_1 & \beta_2 & \cdots & \beta_L \\ \beta_1 & \beta_0 & \beta_1 & \cdots & \beta_{L-1} \\ \beta_2 & \beta_1 & \beta_0 & \beta_1 & \beta_{L-2} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \beta_L & \cdots & \cdots & \cdots & \beta_0 \end{bmatrix} \tag{10-1-45}$$

and

$$\beta_m = \sum_{i=k}^{k+l-1-m} \varepsilon_i \varepsilon_{i+m} \tag{10-1-46}$$

We may use either (10-1-40) and (10-1-41) or (10-1-45)–(10-1-46) in evaluating the error rate performance. We consider these computations later. For now we conclude that the probability of the subevent E_3, given by (10–1–39), may be expressed as

$$\begin{aligned} P(E_3) &= Q\left(\sqrt{\frac{2d^2}{N_0}\,\delta^2(\boldsymbol{\varepsilon})}\right) \\ &= Q\left(\sqrt{\frac{6}{M^2-1}\,\gamma_{\text{av}}\delta^2(\boldsymbol{\varepsilon})}\right) \end{aligned} \tag{10-1-47}$$

where we have used the relation

$$d^2 = \frac{3}{M^2-1}\,TP_{\text{av}} \tag{10-1-48}$$

to eliminate d^2 and $\gamma_{\text{av}} = TP_{\text{av}}/N_0$. Note that, in the absence of intersymbol

interference, $\delta^2(\boldsymbol{\varepsilon}) = 1$ and $P(E_3)$ is proportional to the symbol error probability of M-ary PAM.

The probability of the subevent E_2 depends only on the statistical properties of the input sequence. We assume that the information symbols are equally probable and that the symbols in the transmitted sequence are statistically independent. Then, for an error of the form $|\varepsilon_i| = j$, $j = 1, 2, \ldots, M-1$, there are $M - j$ possible values of I_i such that

$$I_i = \tilde{I}_i + 2d\varepsilon_i$$

Hence

$$P(E_2) = \prod_{i=0}^{l-L-1} \frac{M - |i|}{M} \tag{10-1-49}$$

The probability of the subevent E_1 is much more difficult to compute exactly because of its dependence on the subevent E_3. That is, we must compute $P(E_1 \mid E_3)$. However, $P(E_1 \mid E_3) = 1 - P_M$, where P_M is the symbol error probability. Hence $P(E_1 \mid E_3)$ is well approximated (and upper-bounded) by unity for reasonably low symbol error probabilities. Therefore, the probability of the error event $\boldsymbol{\varepsilon}$ is well approximated and upper-bounded as

$$P(\boldsymbol{\varepsilon}) \leq Q\left(\sqrt{\frac{6}{M^2-1}\gamma_{av}\delta^2(\boldsymbol{\varepsilon})}\right) \prod_{i=0}^{l-L-1} \frac{M - |i|}{M} \tag{10-1-50}$$

Let E be the set of all error events $\boldsymbol{\varepsilon}$ starting at time k and let $w(\boldsymbol{\varepsilon})$ be the corresponding number of nonzero components (Hamming weight or number of symbol errors) in each error event $\boldsymbol{\varepsilon}$. Then the probability of a symbol error is upper-bounded (union bound) as

$$\begin{aligned} P_M &\leq \sum_{\boldsymbol{\varepsilon} \in E} w(\boldsymbol{\varepsilon}) P(\boldsymbol{\varepsilon}) \\ &\leq \sum_{\boldsymbol{\varepsilon} \in E} w(\boldsymbol{\varepsilon}) Q\left(\sqrt{\frac{6}{M^2-1}\gamma_{av}\delta^2(\boldsymbol{\varepsilon})}\right) \prod_{i=0}^{l-L-1} \frac{M - |i|}{M} \end{aligned} \tag{10-1-51}$$

Now let D be the set of all $\delta(\boldsymbol{\varepsilon})$. For each $\delta \in D$, let E_δ be the subset of error events for which $\delta(\boldsymbol{\varepsilon}) = \delta$. Then (10-1-51) may be expressed as

$$\begin{aligned} P_M &\leq \sum_{\delta \in D} Q\left(\sqrt{\frac{6}{M^2-1}\gamma_{av}\delta^2}\right)\left[\sum_{\boldsymbol{\varepsilon} \in E_\delta} w(\boldsymbol{\varepsilon}) \prod_{i=0}^{l-L-1} \frac{M - |i|}{M}\right] \\ &\leq \sum_{\delta \in D} K_\delta Q\left(\sqrt{\frac{6}{M^2-1}\gamma_{av}\delta^2}\right) \end{aligned} \tag{10-1-52}$$

where

$$K_\delta = \sum_{\boldsymbol{\varepsilon} \in E_\delta} w(\boldsymbol{\varepsilon}) \prod_{i=0}^{l-L-1} \frac{M - |i|}{M} \tag{10-1-53}$$

The expression for the error probability in (10-1-52) is similar to the form of the error probability for a convolutional code with soft-decision decoding given

by (8-2-26). The weighting factors $\{K_\delta\}$ may be determined by means of the error state diagram, which is akin to the state diagram of a convolutional encoder. This approach has been illustrated by Forney (1972) and Viterbi and Omura (1979).

In general, however, the use of the error state diagram for computing P_M is tedious. Instead, we may simplify the computation of P_M by focusing on the dominant term in the summation of (10-1-52). Due to the exponential dependence of each term in the sum, the expression P_M is dominated by the term corresponding to the minimum value of δ, denoted as δ_{min}. Hence the symbol error probability may be approximated as

$$P_M \approx K_{\delta_{min}} Q\left(\sqrt{\frac{6}{M^2-1}\gamma_{av}\delta_{min}^2}\right) \tag{10-1-54}$$

where

$$K_{\delta_{min}} = \sum_{\varepsilon \in E_{\delta_{min}}} w(\varepsilon) \prod_{i=0}^{i=L-1} \frac{M-|i|}{M} \tag{10-1-55}$$

In general, $\delta_{min}^2 \leq 1$. Hence, $10\log\delta_{\min}^2$ represents the loss in SNR due to intersymbol interference.

The minimum value of δ may be determined either from (10-1-40) or from evaluation of the quadratic form in (10-1-44) for different error sequences. In the following two examples we use (10-1-40).

Example 10-1-3

Consider a two-path channel ($L=1$) with arbitrary coefficients f_0 and f_1 satisfying the constraint $f_0^2+f_1^2=1$. The channel characteristic is

$$F(z) = f_0 + f_1 z^{-1} \tag{10-1-56}$$

For an error event of length n,

$$\varepsilon(z) = \varepsilon_0 + \varepsilon_1 z^{-1} + \ldots + \varepsilon_{n-1} z^{-(n-1)}, \quad n \geq 1 \tag{10-1-57}$$

The product $\alpha(z) = F(z)\varepsilon(z)$ may be expressed as

$$\alpha(z) = \alpha_0 + \alpha_1 z^{-1} + \ldots + \alpha_n z^{-n} \tag{10-1-58}$$

where $\alpha_0 = \varepsilon_0 f_0$ and $\alpha_n = f_1 \varepsilon_{n-1}$. Since $\varepsilon_0 \neq 0$, $\varepsilon_{n-1} \neq 0$, and

$$\delta^2(\varepsilon) = \sum_{k=0}^{n} \alpha_k^2 \tag{10-1-59}$$

it follows that

$$\delta_{\min}^2 \geq f_0^2 + f_1^2 = 1$$

Indeed, $\delta_{\min}^2 = 1$ when a single error occurs, i.e. $\varepsilon(z) = \varepsilon_0$. Thus, we conclude that there is no loss in SNR in maximum-likelihood sequence

estimation of the information symbols when the channel dispersion has length 2.

Example 10-1-4

The controlled intersymbol interference in a partial response signal may be viewed as having been generated by a time-dispersive channel. Thus, the intersymbol interference from a duobinary pulse may be represented by the (normalized) channel characteristic

$$F(z) = \sqrt{\tfrac{1}{2}} + \sqrt{\tfrac{1}{2}} z^{-1} \tag{10-1-60}$$

Similarly, the representation for a modified duobinary pulse is

$$F(z) = \sqrt{\tfrac{1}{2}} - \sqrt{\tfrac{1}{2}} z^{-2} \tag{10-1-61}$$

The minimum distance $\delta_{\min}^2 = 1$ for any error event of the form

$$\varepsilon(z) = \pm(1 - z^{-1} - z^{-2} \ldots - z^{-(n-1)}), \quad n \geq 1 \tag{10-1-62}$$

for the channel given by (10-1-60) since

$$\alpha(z) = \pm\sqrt{\tfrac{1}{2}} \mp \sqrt{\tfrac{1}{2}} z^{-n}$$

Similarly, when

$$\varepsilon(z) = \pm(1 + z^{-2} - z^{-4} + \ldots + z^{-2(n-1)}), \quad n \geq 1 \tag{10-1-63}$$

$\delta_{\min}^2 = 1$ for the channel given by (10-1-61), since

$$\alpha(z) = \pm\sqrt{\tfrac{1}{2}} \mp \sqrt{\tfrac{1}{2}} z^{-2n}$$

Hence MLSE of these two partial response signals results in no loss in SNR. In contrast, the suboptimum symbol-by-symbol detection described previously resulted in a 2.1 dB loss.

The constant $K_{\delta_{\min}}$ is easily evaluated for these two signals. With precoding, the number of output symbol errors (Hamming weight) associated with the error events in (10-1-62) and (10-1-63) is two. Hence,

$$K_{\delta_{\min}} = 2 \sum_{n=1}^{\infty} \left(\frac{M-1}{M}\right)^n = 2(M-1) \tag{10-1-64}$$

On the other hand, without precoding, these error events result in n symbol errors, and, hence,

$$K_{\delta_{\min}} = 2 \sum_{n=1}^{\infty} n\left(\frac{M-1}{M}\right)^n = 2M(M-1) \tag{10-1-65}$$

As a final exercise, we consider the evaluation of $\delta_{\min}^2$ from the quadratic

form in (10-1-44). The matrix **A** of the quadratic form is positive-definite; hence, all its eigenvalues are positive. If $\{\mu_k(\boldsymbol{\varepsilon})\}$ are the eigenvalues and $\{\mathbf{v}_k(\boldsymbol{\varepsilon})\}$ are the corresponding orthonormal eigenvectors of **A** for an error event $\boldsymbol{\varepsilon}$ then the quadratic form in (10-1-44) can be expressed as

$$\delta^2(\boldsymbol{\varepsilon}) = \sum_{k=1}^{L+1} \mu_k(\boldsymbol{\varepsilon})[\mathbf{f}'\mathbf{v}_k(\boldsymbol{\varepsilon})]^2 \tag{10-1-66}$$

In other words, $\delta^2(\boldsymbol{\varepsilon})$ is expressed as a linear combination of the squared projections of the channel vector **f** onto the eigenvectors of **A**. Each squared projection in the sum is weighted by the corresponding eigenvalue $\mu_k(\boldsymbol{\varepsilon})$, $k = 1, 2, \ldots, L+1$. Then

$$\delta^2_{\min} = \min_{\boldsymbol{\varepsilon}} \delta^2(\boldsymbol{\varepsilon}) \tag{10-1-67}$$

It is interesting to note that the worst channel characteristic of a given length $L+1$ can be obtained by finding the eigenvector corresponding to the minimum eigenvalue. Thus, if $\mu_{\min}(\boldsymbol{\varepsilon})$ is the minimum eigenvalue for a given error event $\boldsymbol{\varepsilon}$ and $\mathbf{v}_{\min}(\boldsymbol{\varepsilon})$ is the corresponding eigenvector then

$$\mu_{min} = \min_{\boldsymbol{\varepsilon}} \mu_{min}(\boldsymbol{\varepsilon})$$

$$\mathbf{f} = \min_{\boldsymbol{\varepsilon}} \mathbf{v}_{\min}(\boldsymbol{\varepsilon})$$

and

$$\delta^2_{\min} = \mu_{\min}$$

Example 10-1-5

Let us determine the worst time-dispersive channel of length 3 ($L = 2$) by finding the minimum eigenvalue of **A** for different error events. Thus,

$$F(z) = f_0 + f_1 z^{-1} + f_2 z^{-2}$$

where f_0, f_1, and f_2 are the components of the eigenvector of **A** corresponding to the minimum eigenvalue. An error event of the form

$$\varepsilon(z) = 1 - z^{-1}$$

results in a matrix

$$\mathbf{A} = \begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}$$

which has the eigenvalues $\mu_1 = 2$, $\mu_2 = 2 + \sqrt{2}$, $\mu_3 = 2 - \sqrt{2}$. The eigenvector corresponding to μ_3 is

$$\mathbf{v}_3' = [\tfrac{1}{2} \quad \sqrt{\tfrac{1}{2}} \quad \tfrac{1}{2}] \tag{10-1-68}$$

We may also consider the dual error event

$$\epsilon(z) = 1 + z^{-1}$$

which results in the matrix

$$\mathbf{A} = \begin{bmatrix} 2 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 1 & 2 \end{bmatrix}$$

This matrix has eigenvalues identical to those of the one for $\epsilon(z) = 1 - z^{-1}$ The corresponding eigenvector for $\mu_3 = 2 - \sqrt{2}$ is

$$\mathbf{v}_3' = [-\tfrac{1}{2} \quad \sqrt{\tfrac{1}{2}} \quad -\tfrac{1}{2}] \tag{10-1-69}$$

Any other error events lead to larger values for $\mu_{\min}$. Hence, $\mu_{\min} = 2 - \sqrt{2}$ and the worst-case channel is either

$$[\tfrac{1}{2} \quad \sqrt{\tfrac{1}{2}} \quad \tfrac{1}{2}] \quad \text{or} \quad [-\tfrac{1}{2} \quad \sqrt{\tfrac{1}{2}} \quad -\tfrac{1}{2}]$$

The loss in SNR from the channel is

$$-10 \log \delta_{\min}^2 = -10 \log \mu_{\min} = 2.3 \text{ dB}$$

Repetitions of the above computation for channels with $L = 3$, 4, and 5 yield the results given in Table 10-1-1.

10-2 LINEAR EQUALIZATION

The MLSE for a channel with ISI has a computational complexity that grows exponentially with the length of the channel time dispersion. If the size of the symbol alphabet is M and the number of interfering symbols contributing to ISI is L, the Viterbi algorithm computes M^{L+1} metrics for each new received symbol. In most channels of practical interest, such a large computational complexity is prohibitively expensive to implement.

In this and the following sections, we describe two suboptimum channel equalization approaches to compensate for the ISI. One approach employs a linear transversal filter, which is described in this section. These filter

TABLE 10-1-1 MAXIMUM PERFORMANCE LOSS AND CORRESPONDING CHANNEL CHARACTERISTICS

Channel length $L+1$	Performance loss $-10 \log \delta_{\min}^2$ (dB)	Minimum-distance channel
3	2.3	0.50, 0.71, 0.50
4	4.2	0.38, 0.60, 0.60, 0.38
5	5.7	0.29, 0.50, 0.58, 0.50, 0.29
6	7.0	0.23, 0.42, 0.52, 0.52, 0.42, 0.23

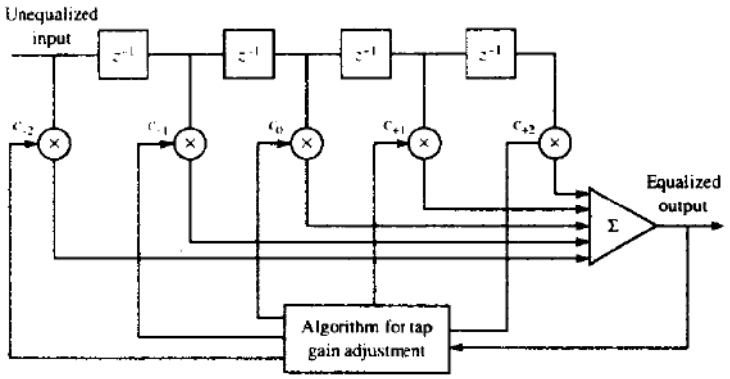


FIGURE 10-2-1 Linear transversal filter.

structures have a computational complexity that is a linear function of the channel dispersion length L.

The linear filter most often used for equalization is the transversal filter shown in Fig. 10-2-1. Its input is the sequence $\{v_k\}$ given in (10-1-16) and its output is the estimate of the information sequence $\{I_k\}$. The estimate of the k th symbol may be expressed as

$$\hat{I}_k = \sum_{j=-K}^{K} c_j v_{k-j} \tag{10-2-1}$$

where $\{c_j\}$ are the $2K + 1$ complex-valued tap weight coefficients of the filter. The estimate $\hat{I}_k$ is quantized to the nearest (in distance) information symbol to form the decision $\tilde{I}_k$. If $\tilde{I}_k$ is not identical to the transmitted information symbol I_k, an error has been made.

Considerable research has been performed on the criterion for optimizing the filter coefficients $\{c_k\}$. Since the most meaningful measure of performance for a digital communications system is the average probability of error, it is desirable to choose the coefficients to minimize this performance index. However, the probability of error is a highly nonlinear function of $\{c_j\}$. Consequently, the probability of error as a performance index for optimizing the tap weight coefficients of the equalizer is impractical.

Two criteria have found widespread use in optimizing the equalizer coefficients $\{c_j\}$. One is the peak distortion criterion and the other is the mean square error criterion.

10-2-1 Peak Distortion Criterion

The peak distortion is simply defined as the worst-case intersymbol interference at the output of the equalizer. The minimization of this performance index is called the *peak distortion criterion*. First we consider the minimization

of the peak distortion assuming that the equalizer has an infinite number of taps. Then we shall discuss the case in which the transversal equalizer spans a finite time duration.

We observe that the cascade of the discrete-time linear filter model having an impulse response $\{f_n\}$ and an equalizer having an impulse response $\{c_n\}$ can be represented by a single equivalent filter having the impulse response

$$q_n = \sum_{j=-\infty}^{\infty} c_j f_{n-j} \tag{10-2-2}$$

That is, $\{q_n\}$ is simply the convolution of $\{c_n\}$ and $\{f_n\}$. The equalizer is assumed to have an infinite number of taps. Its output at the k th sampling instant can be expressed in the form

$$\hat{I}_k = q_0 I_k + \sum_{n \neq k} I_n q_{k-n} + \sum_{j=-\infty}^{\infty} c_j \eta_{k-j} \tag{10-2-3}$$

The first term in (10-2-3) represents a scaled version of the desired symbol. For convenience, we normalize q_0 to unity. The second term is the intersymbol interference. The peak value of this interference, which is called the *peak distortion*, is

$$\begin{aligned} \mathcal{D}(\mathbf{c}) &= \sum_{\substack{n=-\infty \\ n \neq 0}}^{\infty} |q_n| \\ &= \sum_{\substack{n=-\infty \\ n \neq 0}}^{\infty} \left| \sum_{j=-\infty}^{\infty} c_j f_{n-j} \right| \end{aligned} \tag{10-2-4}$$

Thus, $\mathcal{D}(\mathbf{c})$ is a function of the equalizer tap weights.

With an equalizer having an infinite number of taps, it is possible to select the tap weights so that $\mathcal{D}(\mathbf{c}) = 0$, i.e., $q_n = 0$ for all n except $n = 0$. That is, the intersymbol interference can be completely eliminated. The values of the tap weights for accomplishing this goal are determined from the condition

$$q_n = \sum_{j=-\infty}^{\infty} c_j f_{n-j} = \begin{cases} 1 & (n = 0) \\ 0 & (n \neq 0) \end{cases} \tag{10-2-5}$$

By taking the z transform of (10-2-5), we obtain

$$Q(z) = C(z)F(z) = 1 \tag{10-2-6}$$

or, simply,

$$C(z) = \frac{1}{F(z)} \tag{10-2-7}$$

where $C(z)$ denotes the z transform of the $\{c_j\}$. Note that the equalizer, with transfer function $C(z)$, is simply the inverse filter to the linear filter model $F(z)$. In other words, complete elimination of the intersymbol interference requires the use of an inverse filter to $F(z)$. We call such a filter a *zero-forcing*

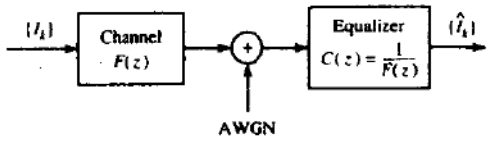


FIGURE 10-2-2 Block diagram of channel with zero-forcing equalizer.

filter. Figure 10-2-2 illustrates in block diagram the equivalent discrete-time channel and equalizer.

The cascade of the noise-whitening filter having the transfer function $1/F^*(z^{-1})$ and the zero-forcing equalizer having the transfer function $1/F(z)$ results in an equivalent zero-forcing equalizer having the transfer function

$$C'(z) = \frac{1}{F(z)F^*(z^{-1})} = \frac{1}{X(z)} \tag{10-2-8}$$

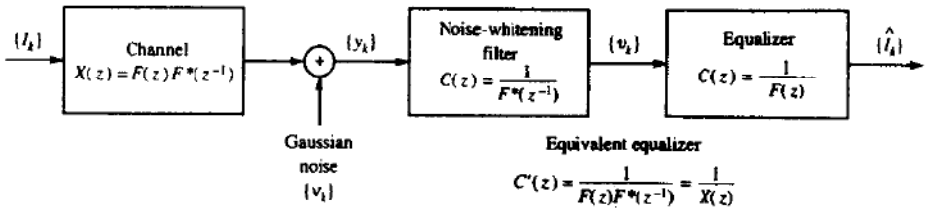
as shown in Fig. 10-2-3. This combined filter has as its input the sequence $\{y_k\}$ of samples from the matched filter, given by (10-1-10). Its output consists of the desired symbols corrupted only by additive zero-mean gaussian noise. The impulse response of the combined filter is

$$\begin{aligned} c_k' &= \frac{1}{2\pi j}\oint C'(z)z^{k-1}\,dz \\ &= \frac{1}{2\pi j}\oint \frac{z^{k-1}}{X(z)}\,dz \end{aligned} \tag{10-2-9}$$

where the integration is performed on a closed contour that lies within the region of convergence of $C'(z)$. Since $X(z)$ is a polynomial with $2L$ roots $(\rho_1, \rho_2, \ldots, \rho_L, 1/\rho_1^*, 1/\rho_2^*, \ldots, 1/\rho_L^*)$, it follows that $C'(z)$ must converge in an annular region in the z plane that includes the unit circle $(z = e^{j\theta})$. Consequently, the closed contour in the integral can be the unit circle.

The performance of the infinite-tap equalizer that completely eliminates the intersymbol interference can be expressed in terms of the signal-to-noise ratio (SNR) at its output. For mathematical convenience, we normalize the received

FIGURE 10-2-3 Block of channel with equivalent zero-forcing equalizer.



signal energy to unity.† This implies that $q_0 = 1$ and that the expected value of $|I_k|^2$ is also unity. Then the SNR is simply the reciprocal of the noise variance σ_n^2 at the output of the equalizer.

The value of σ_n^2 can be simply determined by observing that the noise sequence $\{v_k\}$ at the input to the equivalent zero-forcing equalizer $C'(z)$ has zero mean and a power spectral density

$$\Phi_{vv}(\omega) = N_0 X(e^{j\omega T}), \quad |\omega| \leq \frac{\pi}{T} \tag{10-2-10}$$

where $X(e^{j\omega T})$ is obtained from $X(z)$ by the substitution $z = e^{j\omega T}$. Since $C'(z) = 1/X(z)$, it follows that the noise sequence at the output of the equalizer has a power spectral density

$$\Phi_{nn}(\omega) = \frac{N_0}{X(e^{j\omega T})}, \quad |\omega| \leq \frac{\pi}{T} \tag{10-2-11}$$

Consequently, the variance of the noise variable at the output of the equalizer is

$$\begin{aligned} \sigma_n^2 &= \frac{T}{2\pi} \int_{-\pi/T}^{\pi/T} \Phi_{nn}(\omega)\, d\omega \\ &= \frac{TN_0}{2\pi} \int_{-\pi/T}^{\pi/T} \frac{d\omega}{X(e^{j\omega T})} \end{aligned} \tag{10-2-12}$$

and the SNR for the zero-forcing equalizer is

$$\begin{aligned} \gamma_\infty &= 1/\sigma_n^2 \\ &= \left[\frac{TN_0}{2\pi} \int_{-\pi/T}^{\pi/T} \frac{d\omega}{X(e^{j\omega T})} \right]^{-1} \end{aligned} \tag{10-2-13}$$

where the subscript on γ indicates that the equalizer has an infinite number of taps.

The spectral characteristics $X(e^{j\omega T})$ corresponding to the Fourier transform of the sampled sequence $\{x_n\}$ has an interesting relationship to the analog filter $H(\omega)$ used at the receiver. Since

$$x_k = \int_{-\infty}^{\infty} h^*(t) h(t + kT)\, dt$$

use of Parseval's theorem yields

$$x_k = \frac{1}{2\pi} \int_{-\infty}^{\infty} |H(\omega)|^2 e^{j\omega kT}\, d\omega \tag{10-2-14}$$

where $H(\omega)$ is the Fourier transform of $h(t)$. But the integral in (10-2-14) can be expressed in the form

$$x_k = \frac{1}{2\pi} \int_{-\pi/T}^{\pi/T} \left[\sum_{n=-\infty}^{\infty} \left| H\left(\omega + \frac{2\pi n}{T}\right) \right|^2 \right] e^{j\omega kT}\, d\omega \tag{10-2-15}$$

† This normalization is used throughout this chapter for mathematical convenience.

Now, the Fourier transform of $\{x_k\}$ is

$$X(e^{j\omega T}) = \sum_{k=-\infty}^{\infty} x_k e^{-j\omega kT} \tag{10-2-16}$$

and the inverse transform yields

$$x_k = \frac{T}{2\pi} \int_{-\pi/T}^{\pi/T} X(e^{j\omega T}) e^{j\omega kT}\, d\omega \tag{10-2-17}$$

From a comparison of (10-2-15) and (10-2-17), we obtain the desired relationship between $X(e^{j\omega T})$ and $H(\omega)$. That is,

$$X(e^{j\omega T}) = \frac{1}{T} \sum_{n=-\infty}^{\infty} \left| H\left(\omega + \frac{2\pi n}{T}\right)\right|^2, \quad |\omega| \leq \frac{\pi}{T} \tag{10-2-18}$$

where the right-hand side of (10-2-18) is called the *folded spectrum* of $|H(\omega)|^2$. We also observe that $|H(\omega)|^2 = X(\omega)$, where $X(\omega)$ is the Fourier transform of the waveform $x(t)$ and $x(t)$ is the response of the matched filter to the input $h(t)$. Therefore the right-hand side of (10-2-18) can also be expressed in terms of $X(\omega)$.

Substitution for $X(e^{j\omega T})$ in (10-2-13) using the result in (10-2-18) yields the desired expression for the SNR in the form

$$\gamma_\infty = \left[\frac{T^2 N_0}{2\pi} \int_{-\pi/T}^{\pi/T} \frac{d\omega}{\sum_{n=-\infty}^{\infty} |H(\omega + 2\pi n/T)|^2} \right]^{-1} \tag{10-2-19}$$

We observe that if the folded spectral characteristic of $H(\omega)$ possesses any zeros, the integrand becomes infinite and the SNR goes to zero. In other words, the performance of the equalizer is poor whenever the folded spectral characteristic possesses nulls or takes on small values. This behavior occurs primarily because the equalizer, in eliminating the intersymbol interference, enhances the additive noise. For example, if the channel contains a spectral null in its frequency response, the linear zero-forcing equalizer attempts to compensate for this by introducing an infinite gain at that frequency. But this compensates for the channel distortion at the expense of enhancing the additive noise. On the other hand, an ideal channel coupled with an appropriate signal design that results in no intersymbol interference will have a folded spectrum that satisfies the condition

$$\sum_{n=-\infty}^{\infty} \left| H\left(\omega + \frac{2\pi n}{T}\right)\right|^2 = T, \quad |\omega| \leq \frac{\pi}{T} \tag{10-2-20}$$

In this case, the SNR achieves its maximum value, namely,

$$\gamma_\infty = \frac{1}{N_0} \tag{10-2-21}$$

Finite-Length Equalizer Let us now turn our attention to an equalizer having $2K+1$ taps. Since $c_j = 0$ for $|j| > K$, the convolution of $\{f_n\}$ with $\{c_n\}$ is zero outside the range $-K \leq n \leq K+L-1$. That is, $q_n = 0$ for $n < -K$ and $n > K+L-1$. With q_0 normalized to unity, the peak distortion is

$$\mathcal{D}(\mathbf{c}) = \sum_{\substack{n=-K \\ n\neq 0}}^{K+L-1} |q_n| = \sum_{\substack{n=-K \\ n\neq 0}}^{K+L-1} \left| \sum_j c_j f_{n-j} \right| \tag{10-2-22}$$

Although the equalizer has $2K+1$ adjustable parameters, there are $2K+L$ nonzero values in the response $\{q_n\}$. Therefore, it is generally impossible to completely eliminate the intersymbol interference at the output of the equalizer. There is always some residual interference when the optimum coefficients are used. The problem is to minimize $\mathcal{D}(\mathbf{c})$ with respect to the coefficients $\{c_j\}$.

The peak distortion given by (10-2-22) has been shown by Lucky (1965) to be a convex function of the coefficients $\{c_j\}$. That is, it possesses a global minimum and no relative minima. Its minimization can be carried out numerically using, for example, the method of steepest descent. Little more can be said for the general solution to this minimization problem. However, for one special but important case, the solution for the minimization of $\mathcal{D}(\mathbf{c})$ is known. This is the case in which the distortion at the input to the equalizer, defined as

$$D_0 = \frac{1}{|f_0|} \sum_{n=1}^{L} |f_n| \tag{10-2-23}$$

is less than unity. This condition is equivalent to having the eye open prior to equalization. That is, the intersymbol interference is not severe enough to close the eye. Under this condition, the peak distortion $\mathcal{D}(\mathbf{c})$ is minimized by selecting the equalizer coefficients to force $q_n = 0$ for $1 \leq |n| \leq K$ and $q_0 = 1$. In other words, the general solution to the minimization of $\mathcal{D}(\mathbf{c})$, when $D_0 < 1$, is the zero-forcing solution for $\{q_n\}$ in the range $1 \leq |n| \leq K$. However, the values of $\{q_n\}$ for $K+1 \leq n \leq K+L-1$ are nonzero, in general. These nonzero values constitute the residual intersymbol interference at the output of the equalizer.

10-2-2 Mean Square Error (MSE) Criterion

In the MSE criterion, the tap weight coefficients $\{c_j\}$ of the equalizer are adjusted to minimize the mean square value of the error

$$\varepsilon_k = I_k - \hat{I}_k \tag{10-2-24}$$

where I_k is the information symbol transmitted in the k th signaling interval and $\hat{I}_k$ is the estimate of that symbol at the output of the equalizer, defined in

(10-2-1). When the information symbols $\{I_k\}$ are complex-valued, the performance index for the MSE criterion, denoted by J, is defined as

$$\begin{aligned} J &= E\,|\varepsilon_k|^2 \\ &= E\,|I_k - \hat{I}_k|^2 \end{aligned} \tag{10-2-25}$$

On the other hand, when the information symbols are real-valued, the performance index is simply the square of the real part of ε_k. In either case, J is a quadratic function of the equalizer coefficients $\{c_j\}$. In the following discussion, we consider the minimization of the complex-valued form given in (10-2-25).

Infinite-Length Equalizer First, we shall derive the tap weight coefficients that minimize J when the equalizer has an infinite number of taps. In this case, the estimate $\hat{I}_k$ is expressed as

$$\hat{I}_k = \sum_{j=-\infty}^{\infty} c_j v_{k-j} \tag{10-2-26}$$

Substitution of (10-2-26) into the expression for J given in (10-2-25) and expansion of the result yields a quadratic function of the coefficients $\{c_j\}$. This function can be easily minimized with respect to the $\{c_j\}$ to yield a set (infinite in number) of linear equations for the $\{c_j\}$. Alternatively, the set of linear equations can be obtained by invoking the orthogonality principle in mean square estimation. That is, we select the coefficients $\{c_j\}$ to render the error ε_k orthogonal to the signal sequence $\{v_{k-l}^*\}$ for $-\infty < l < \infty$. Thus,

$$E(\varepsilon_k v_{k-l}^*) = 0, \qquad -\infty < l < \infty \tag{10-2-27}$$

Substitution for ε_k in (10-2-27) yields

$$E\left[\left(I_k - \sum_{j=-\infty}^{\infty} c_j v_{k-j}\right) v_{k-l}^*\right] = 0$$

or, equivalently,

$$\sum_{j=-\infty}^{\infty} c_j E(v_{k-j} v_{k-l}^*) = E(I_k v_{k-l}^*), \qquad -\infty < l < \infty \tag{10-2-28}$$

To evaluate the moments in (10-2-28), we use the expression for v_k given in (10-1-16). Thus, we obtain

$$\begin{aligned} E(v_{k-j} v_{k-l}^*) &= \sum_{n=0}^{L} f_n^* f_{n+l-j} + N_0 \delta_{lj} \\ &= \begin{cases} x_{l-j} + N_0 \delta_{lj} & (|l-j| \le L) \\ 0 & (\text{otherwise}) \end{cases} \end{aligned} \tag{10-2-29}$$

and

$$E(I_k v_{k-l}^*) = \begin{cases} f_{-l}^* & (-L \le l \le 0) \\ 0 & (\text{otherwise}) \end{cases} \tag{10-2-30}$$

Now, if we substitute (10-2-29) and (10-2-30) into (10-2-28) and take the z transform of both sides of the resulting equation, we obtain

$$C(z)[F(z)F^*(z^{-1}) + N_0] = F^*(z^{-1}) \tag{10-2-31}$$

Therefore, the transfer function of the equalizer based on the MSE criterion is

$$C(z) = \frac{F^*(z^{-1})}{F(z)F^*(z^{-1}) + N_0} \tag{10-2-32}$$

When the noise-whitening filter is incorporated into $C(z)$, we obtain an equivalent equalizer having the transfer function

$$\begin{aligned} C'(z) &= \frac{1}{F(z)F^*(z^{-1}) + N_0} \\ &= \frac{1}{X(z) + N_0} \end{aligned} \tag{10-2-33}$$

We observe that the only difference between this expression for $C'(z)$ and the one based on the peak distortion criterion is the noise spectral density factor N_0 that appears in (10-2-33). When N_0 is very small in comparison with the signal, the coefficients that minimize the peak distortion $\mathscr{D}(\mathbf{c})$ are approximately equal to the coefficients that minimize the MSE performance index J. That is, in the limit as $N_0 \to 0$, the two criteria yield the same solution for the tap weights. Consequently, when $N_0 = 0$, the minimization of the MSE results in complete elimination of the intersymbol interference. On the other hand, that is not the case when $N_0 \neq 0$. In general, when $N_0 \neq 0$, there is both residual intersymbol interference and additive noise at the output of the equalizer.

A measure of the residual intersymbol interference and additive noise is obtained by evaluating the minimum value of J, denoted by $J_{\min}$, when the transfer function $C(z)$ of the equalizer is given by (10-2-32). Since $J = E|\varepsilon_k|^2 = E(\varepsilon_k I_k^*) - E(\varepsilon_k \hat{I}_k^*)$, and since $E(\varepsilon_k \hat{I}_k^*) = 0$ by virtue of the orthogonality conditions given in (10-2-27), it follows that

$$\begin{aligned} J_{\min} &= E(\varepsilon_k I_k^*) \\ &= E|I_k|^2 - \sum_{j=-\infty}^{\infty} c_j E(v_{k-j} I_k^*) \\ &= 1 - \sum_{j=-\infty}^{\infty} c_j f_{-j} \end{aligned} \tag{10-2-34}$$

This particular form for $J_{\min}$ is not very informative. More insight on the performance of the equalizer as a function of the channel characteristics is obtained when the summation in (10-2-34) is transformed into the frequency domain. This can be accomplished by first noting that the summation in (10-2-34) is the convolution of $\{c_j\}$ with $\{f_j\}$, evaluated at a shift of zero. Thus,

if $\{b_k\}$ denotes the convolution of these two sequences, the summation in (10-2-34) is simply equal to b_0. Since the z transform of the sequence $\{b_k\}$ is

$$\begin{aligned} B(z) &= C(z)F(z) \\ &= \frac{F(z)F^*(z^{-1})}{F(z)F^*(z^{-1}) + N_0} \\ &= \frac{X(z)}{X(z) + N_0} \end{aligned} \tag{10-2-35}$$

the term b_0 is

$$\begin{aligned} b_0 &= \frac{1}{2\pi j} \oint \frac{B(z)}{z} dz \\ &= \frac{1}{2\pi j} \oint \frac{X(z)}{z[X(z) + N_0]} dz \end{aligned} \tag{10-2-36}$$

The contour integral in (10-2-36) can be transformed into an equivalent line integral by the change of variable $z = e^{j\omega T}$. The result of this change of variable is

$$b_0 = \frac{T}{2\pi} \int_{-\pi/T}^{\pi/T} \frac{X(e^{j\omega T})}{X(e^{j\omega T}) + N_0} d\omega \tag{10-2-37}$$

Finally, substitution of the result in (10-2-37) for the summation in (10-2-34) yields the desired expression for the minimum MSE in the form

$$\begin{aligned} J_{\min} &= 1 - \frac{T}{2\pi} \int_{-\pi/T}^{\pi/T} \frac{X(e^{j\omega T})}{X(e^{j\omega T}) + N_0} d\omega \\ &= \frac{T}{2\pi} \int_{-\pi/T}^{\pi/T} \frac{N_0}{X(e^{j\omega T}) + N_0} d\omega \\ &= \frac{T}{2\pi} \int_{-\pi/T}^{\pi/T} \frac{N_0}{T^{-1} \sum_{n=-\infty}^{\infty} |H(\omega + 2\pi n/T)|^2 + N_0} d\omega \end{aligned} \tag{10-2-38}$$

In the absence of intersymbol interference, $X(e^{j\omega T}) = 1$ and, hence,

$$J_{\min} = \frac{N_0}{1 + N_0} \tag{10-2-39}$$

We observe that $0 \leq J_{\min} \leq 1$. Furthermore, the relationship between the output (normalized by the signal energy) SNR γ_∞ and $J_{\min}$ must be

$$\gamma_\infty = \frac{1 - J_{\min}}{J_{\min}} \tag{10-2-40}$$

More importantly, this relation between γ_∞ and $J_{\min}$ also holds when there is residual intersymbol interference in addition to the noise.

Finite-Length Equalizer Let us now turn our attention to the case in which the transversal equalizer spans a finite time duration. The output of the equalizer in the k th signaling interval is

$$\hat{I}_k = \sum_{j=-K}^{K} c_j v_{k-j} \tag{10-2-41}$$

The MSE for the equalizer having $2K + 1$ taps, denoted by $J(K)$, is

$$J(K) = E\,|I_k - \hat{I}_k|^2 = E\left|I_k - \sum_{j=-K}^{K} c_j v_{k-j}\right|^2 \tag{10-2-42}$$

Minimization of $J(K)$ with respect to the tap weights $\{c_j\}$ or, equivalently, forcing the error $\varepsilon_k = I_k - \hat{I}_k$ to be orthogonal to the signal samples v_{j-l}^*, $|l| \leq K$, yields the following set of simultaneous equations:

$$\sum_{j=-K}^{K} c_j \Gamma_{lj} = \xi_l, \qquad l = -K, \ldots, -1, 0, 1, \ldots, K \tag{10-2-43}$$

where

$$\Gamma_{lj} = \begin{cases} x_{l-j} + N_0 \delta_{lj} & (|l-j| \leq L) \\ 0 & (\text{otherwise}) \end{cases} \tag{10-2-44}$$

and

$$\xi_l = \begin{cases} f_{-l}^* & (-L \leq l \leq 0) \\ 0 & (\text{otherwise}) \end{cases} \tag{10-2-45}$$

It is convenient to express the set of linear equations in matrix form. Thus,

$$\mathbf{\Gamma C} = \boldsymbol{\xi} \tag{10-2-46}$$

where $\mathbf{C}$ denotes the column vector of $2K + 1$ tap weight coefficients, $\mathbf{\Gamma}$ denotes the $(2K + 1) \times (2K + 1)$ Hermitian covariance matrix with elements Γ_{ij}, and $\boldsymbol{\xi}$ is a $(2K + 1)$-dimensional column vector with elements ξ_l. The solution of (10-2-46) is

$$\mathbf{C}_{\text{opt}} = \mathbf{\Gamma}^{-1}\boldsymbol{\xi} \tag{10-2-47}$$

Thus, the solution for $\mathbf{C}_{\text{opt}}$ involves inverting the matrix $\mathbf{\Gamma}$. The optimum tap weight coefficients given by (10-2-47) minimize the performance index $J(K)$, with the result that the minimum value of $J(K)$ is

$$\begin{aligned} J_{\min}(K) &= 1 - \sum_{j=-K}^{0} c_j f_{-j} \\ &= 1 - \boldsymbol{\xi}'^* \mathbf{\Gamma}^{-1} \boldsymbol{\xi} \end{aligned} \tag{10-2-48}$$

where $\boldsymbol{\xi}'$ represents the transpose of the column vector $\boldsymbol{\xi}$. $J_{\min}(K)$ may be used

in (10-2-40) to compute the output SNR for the linear equalizer with $2K + 1$ tap coefficients.

10-2-3 Performance Characteristics of the MSE Equalizer

In this section, we consider the performance characteristics of the linear equalizer that is optimized by using the MSE criterion. Both the minimum MSE and the probability of error are considered as performance measures for some specific channels. We begin by evaluating the minimum MSE $J_{\min}$ and the output SNR γ_∞ for two specific channels. Then, we consider the evaluation of the probability of error.

Example 10-2-1

First, we consider an equivalent discrete-time channel model consisting of two components f_0 and f_1, which are normalized to $|f_0|^2 + |f_1|^2 = 1$. Then

$$F(z) = f_0 + f_1 z^{-1} \tag{10-2-49}$$

and

$$X(z) = f_0 f_1^* z + 1 + f_0^* f_1 z^{-1} \tag{10-2-50}$$

The corresponding frequency response is

$$\begin{aligned} X(e^{j\omega T}) &= f_0 f_1^* e^{j\omega T} + 1 + f_0^* f_1 e^{-j\omega T} \\ &= 1 + 2|f_0||f_1|\cos(\omega T + \theta) \end{aligned} \tag{10-2-51}$$

where θ is the angle of $f_0 f_1^*$. We note that this channel characteristic possesses a null at $\omega = \pi/T$ when $f_0 = f_1 = \sqrt{\frac{1}{2}}$.

A linear equalizer with an infinite number of taps, adjusted on the basis of the MSE criterion, will have the minimum MSE given by (10-2-38). Evaluation of the integral in (10-2-38) for the $X(e^{j\omega T})$ given in (10-2-51) yields the result

$$\begin{aligned} J_{\min} &= \frac{N_0}{\sqrt{N_0^2 + 2N_0(|f_0|^2 + |f_1|^2) + (|f_0|^2 - |f_1|^2)^2}} \\ &= \frac{N_0}{\sqrt{N_0^2 + 2N_0 + (|f_0|^2 - |f_1|^2)^2}} \end{aligned} \tag{10-2-52}$$

Let us consider the special case in which $f_0 = f_1 = \sqrt{\frac{1}{2}}$. The minimum MSE is $J_{\min} = N_0/\sqrt{N_0^2 + 2N_0}$ and the corresponding output SNR is

$$\begin{aligned} \gamma_\infty &= \sqrt{1 + \frac{2}{N_0}} - 1 \\ &\approx \left(\frac{2}{N_0}\right)^{1/2}, \quad N_0 \ll 1 \end{aligned} \tag{10-2-53}$$

This result should be compared with the output SNR of $1/N_0$ obtained in

the case of no intersymbol interference. A significant loss in SNR occurs from this channel.

Example 10-2-2

As a second example, we consider an exponentially decaying characteristic of the form

$$f_k = \sqrt{1-a^2}\,a^k, \qquad k = 0, 1, \ldots$$

where $a < 1$. The Fourier transform of this sequence is

$$X(e^{j\omega T}) = \frac{1-a^2}{1+a^2-2a\cos\omega T} \tag{10-2-54}$$

which is a function that contains a minimum at $\omega = \pi/T$.

The output SNR for this channel is

$$\gamma_\infty = \left(\sqrt{1+2N_0\frac{1+a^2}{1-a^2}+N_0^2}-1\right)^{-1}$$

$$\approx \frac{1-a^2}{(1+a^2)N_0}, \qquad N_0 \ll 1 \tag{10-2-55}$$

Therefore, the loss in SNR due to the presence of the interference is

$$10\log_{10}\left(\frac{1-a^2}{1+a^2}\right)$$

Probability of Error Performance of Linear MSE Equalizer Above, we discussed the performance of the linear equalizer in terms of the minimum achievable MSE $J_{\min}$ and the output SNR γ that is related to $J_{\min}$ through the formula in (10-2-40). Unfortunately, there is no simple relationship between these quantities and the probability of error. The reason is that the linear MSE equalizer contains some residual intersymbol interference at its output. This situation is unlike that of the infinitely long zero-forcing equalizer, for which there is no residual interference, but only gaussian noise. The residual interference at the output of the MSE equalizer is not well characterized as an additional gaussian noise term, and, hence, the output SNR does not translate easily into an equivalent error probability.

One approach to computing the error probability is a brute force method that yields an exact result. To illustrate this method, let us consider a PAM signal in which the information symbols are selected from the set of values $2n - M - 1$, $n = 1, 2, \ldots, M$, with equal probability. Now consider the decision on the symbol I_n. The estimate of I_n is

$$\hat{I}_n = q_0 I_n + \sum_{k\neq n} I_k q_{n-k} + \sum_{j=-K}^{K} c_j \eta_{n-j} \tag{10-2-56}$$

where $\{q_n\}$ represent the convolution of the impulse response of the equalizer and equivalent channel, i.e.,

$$q_n = \sum_{k=-K}^{K} c_k f_{n-k} \tag{10-2-57}$$

and the input signal to the equalizer is

$$v_k = \sum_{j=0}^{L} f_j I_{k-j} + \eta_k \tag{10-2-58}$$

The first term in the right-hand side of (10-2-56) is the desired symbol, the middle term is the intersymbol interference, and the last term is the gaussian noise. The variance of the noise is

$$\sigma_n^2 = N_0 \sum_{j=-K}^{K} c_j^2 \tag{10-2-59}$$

For an equalizer with $2K + 1$ taps and a channel response that spans $L + 1$ symbols, the number of symbols involved in the intersymbol interference is $2K + L$.

Define

$$\mathscr{D} = \sum_{k \neq n} I_k q_{n-k} \tag{10-2-60}$$

For a particular sequence of $2K + L$ information symbols, say the sequence $\mathbf{I}_J$, the intersymbol interference term $\mathscr{D} \equiv D_J$ is fixed. The probability of error for a fixed D_J is

$$\begin{aligned} P_M(D_J) &= 2\frac{(M-1)}{M} P(N + D_J > q_0) \\ &= \frac{2(M-1)}{M} Q\left(\sqrt{\frac{(q_0 - D_J)^2}{\sigma_n^2}}\right) \end{aligned} \tag{10-2-61}$$

where N denotes the additive noise term. The average probability of error is obtained by averaging $P_M(D_J)$ over all possible sequences $\mathbf{I}_J$. That is,

$$\begin{aligned} P_M &= \sum_{\mathbf{I}_J} P_M(D_J) P(\mathbf{I}_J) \\ &= \frac{2(M-1)}{M} \sum_{\mathbf{I}_J} Q\left(\sqrt{\frac{(q_0 - D_J)^2}{\sigma_n^2}}\right) P(\mathbf{I}_J) \end{aligned} \tag{10-2-62}$$

When all the sequences are equally likely,

$$P(\mathbf{I}_J) = \frac{1}{M^{2K+L}} \tag{10-2-63}$$

The conditional error probability terms $P_M(D_J)$ are dominated by the sequence that yields the largest value of D_J. This occurs when $I_n = \pm(M-1)$

and the signs of the information symbols match the signs of the corresponding $\{q_n\}$. Then,

$$D_f^* = (M-1)\sum_{k\neq 0}|q_k|$$

and

$$P_M(D_f^*) = \frac{2(M-1)}{M}\,Q\left(\sqrt{\frac{q_0^2}{\sigma_n^2}\left(1-\frac{M-1}{q_0}\sum_{k\neq 0}|q_k|\right)^2}\right) \qquad (10\text{-}2\text{-}64)$$

Thus, an upper bound on the average probability of error for equally likely symbol sequences is

$$P_M \leq P_M(D_f^*) \qquad (10\text{-}2\text{-}65)$$

If the computation of the exact error probability in (10-2-62) proves to be too cumbersome and too time consuming because of the large number of terms in the sum and if the upper bound is too loose, one can resort to one of a number of different approximate methods that have been devised, which are known to yield tight bounds on P_M. A discussion of these different approaches would take us too far afield. The interested reader is referred to the papers by Saltzberg (1968), Lugannani (1969), Ho and Yeh (1970), Shimbo and Celebiler (1971), Glave (1972), Yao (1972), and Yao and Tobin (1976).

As an illustration of the performance limitations of a linear equalizer in the presence of severe intersymbol interference, we show in Fig. 10-2-4 the probability of error for binary (antipodal) signaling, as measured by Monte Carlo simulation, for the three discrete-time channel characteristic shown in

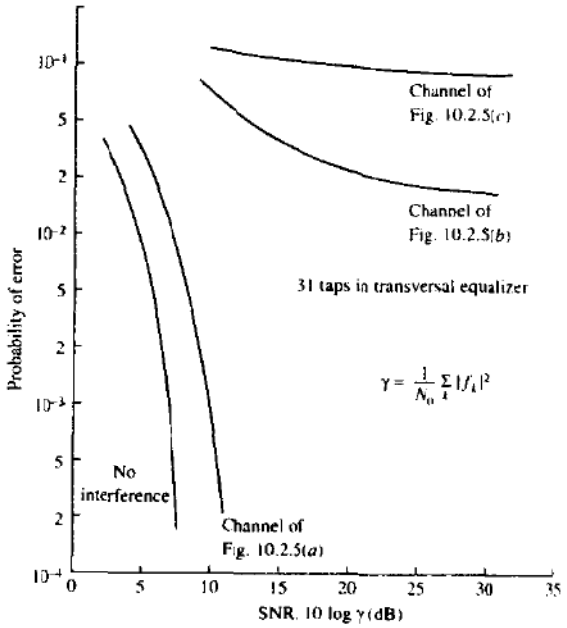


FIGURE 10-2-4 Error rate performance of linear MSE equalizer.

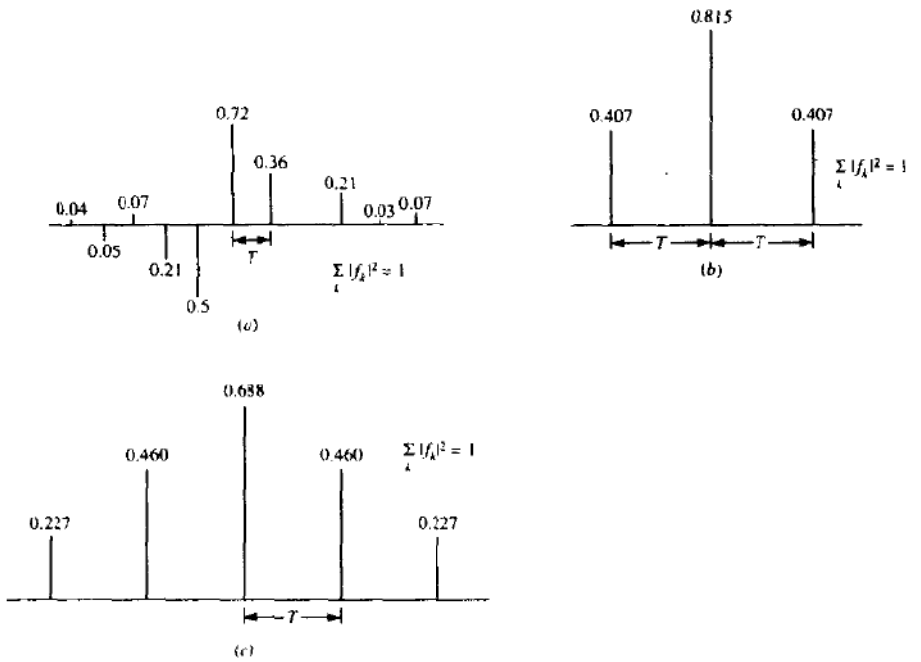


FIGURE 10-2-5 Three discrete-time channel characteristics.

Fig. 10-2-5. For purposes of comparison, the performance obtained for a channel with no intersymbol interference is also illustrated in Fig. 10-2-4. The equivalent discrete-time channel shown in Fig. 10-2-5(a) is typical of the response of a good quality telephone channel. In contrast, the equivalent discrete-time channel characteristics shown in Fig. 10-2-5(b) and (c) result in severe intersymbol interference. The spectral characteristics $|X(e^{j\omega})|$ for the three channels, illustrated in Fig. 10-2-6, clearly show that the channel in Fig. 10-2-5(c) has the worst spectral characteristic. Hence the performance of the linear equalizer for this channel is the poorest of the three cases. Next in performance is the channel shown in Fig. 10-2-5(b), and finally, the best performance is obtained with the channel shown in Fig. 10-2-5(a). In fact, the error rate of the latter is within 3 dB of the error rate achieved with no interference.

One conclusion reached from the results on output SNR γ_c and the limited probability of error results illustrated in Fig. 10-2-4 is that a linear equalizer yields good performance on channels such as telephone lines, where the spectral characteristics of the channels are well behaved and do not exhibit spectral nulls. On the other hand, a linear equalizer is inadequate as a compensator for the intersymbol interference on channels with spectral nulls, which may be encountered in radio transmission.

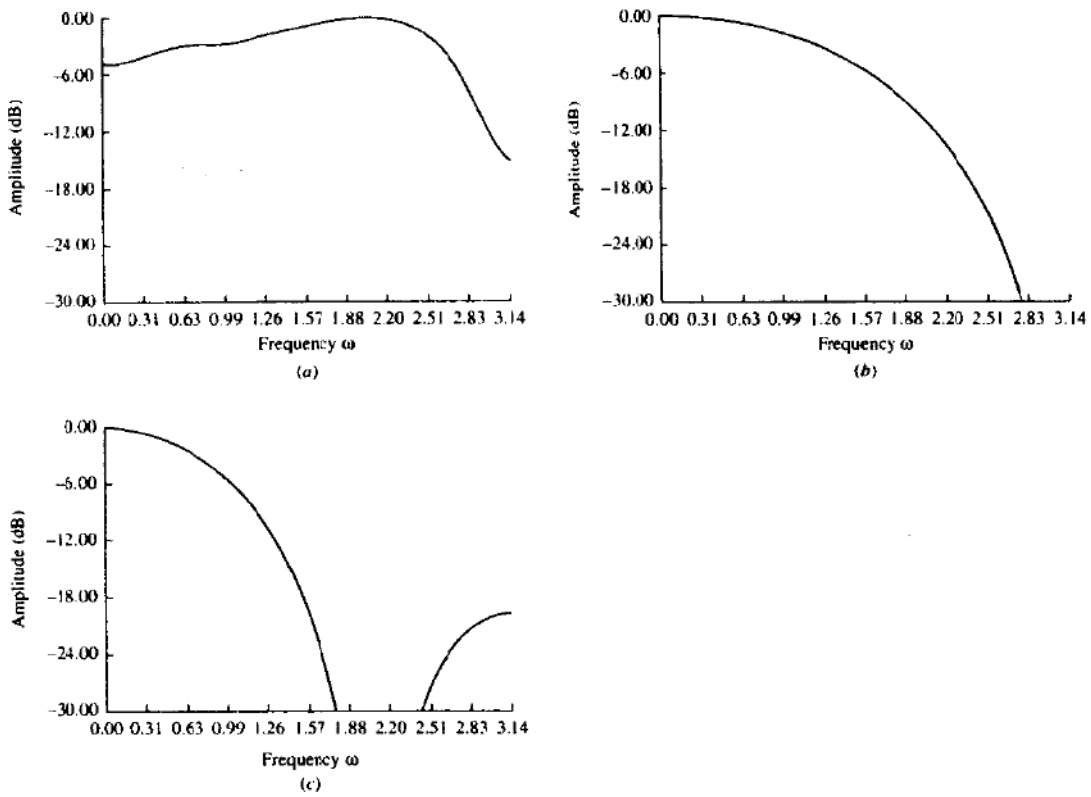


FIGURE 10-2-6 Amplitude spectra for the channels shown in Figs 10-2-5(*a*), (*b*), and (*c*), respectively.

The basic limitation of the linear equalizer to cope with severe ISI has motivated a considerable amount of research into nonlinear equalizers with low computational complexity. The decision-feedback equalizer described in Section 10-3 is shown to be an effective solution to this problem.

10-2-4 Fractionally Spaced Equalizers

In the linear equalizer structures that we have described in the previous section, the equalizer taps are spaced at the reciprocal of the symbol rate, i.e., at the reciprocal of the signaling rate $1/T$. This tap spacing is optimum if the equalizer is preceded by a filter matched to the channel distorted transmitted pulse. When the channel characteristics are unknown, the receiver filter is usually matched to the transmitted signal pulse and the sampling time is optimized for this suboptimum filter. In general, this approach leads to an equalizer performance that is very sensitive to the choice of sampling time.

The limitations of the symbol rate equalizer are most easily evident in the

frequency domain. From (9-2-5), the spectrum of the signal at the input to the equalizer may be expressed as

$$Y_T(f) = \frac{1}{T}\sum_n X\left(f - \frac{n}{T}\right)e^{j2\pi(f - n/T)\tau_0} \tag{10-2-66}$$

where $Y_T(f)$ is the folded or aliased spectrum, where the folding frequency is $1/2T$. Note that the received signal spectrum is dependent on the choice of the sampling delay τ_0. The signal spectrum at the output of the equalizer is $C_T(f)Y_T(f)$, where

$$C_T(f) = \sum_{k=-K}^{K} c_k e^{-j2\pi fkT} \tag{10-2-67}$$

It is clear from these relationships that the symbol rate equalizer can only compensate for the frequency response characteristics of the aliased received signal. It cannot compensate for the channel distortion inherent in $X(f)e^{j2\pi f\tau_0}$.

In contrast to the symbol rate equalizer, a *fractionally spaced equalizer* (FSE) is based on sampling the incoming signal at least as fast as the Nyquist rate. For example, if the transmitted signal consists of pulses having a raised cosine spectrum with a roll-off factor β, its spectrum extends to $F_{\max} = (1 + \beta)/2T$. This signal can be sampled at the receiver at a rate

$$2F_{\max} = \frac{1 + \beta}{T} \tag{10-2-68}$$

and then passed through an equalizer with tap spacing of $T/(1 + \beta)$. For example, if $\beta = 1$, we would have a $\frac{1}{2}T$-spaced equalizer. If $\beta = 0.5$, we would have a $\frac{2}{3}T$-spaced equalizer, and so forth. In general, then, a digitally implemented fractionally spaced equalizer has tap spacing of MT/N where M and N are integers and $N > M$. Usually, a $\frac{1}{2}T$-spaced equalizer is used in many applications.

Since the frequency response of the FSE is

$$C_{T'}(f) = \sum_{k=-K}^{K} c_k e^{-j2\pi fkT'} \tag{10-2-69}$$

where $T' = MT/N$, it follows that $C_{T'}(f)$ can equalize the received signal spectrum beyond the Nyquist frequency $f = 1/2T$ to $f = (1 + \beta)/T = N/MT$. The equalized spectrum is

$$\begin{aligned} C_{T'}(f)Y_{T'}(f) &= C_{T'}(f)\sum_n X\left(f - \frac{n}{T'}\right)e^{j2\pi(f - n/T')\tau_0} \\ &= C_{T'}(f)\sum_n X\left(f - \frac{nN}{MT}\right)e^{j2\pi(f - nN/MT)\tau_0} \end{aligned} \tag{10-2-70}$$

Since $X(f) = 0$ for $|f| > N/MT$, (10-2-70) may be expressed as

$$C_{T'}(f)Y_{T'}(f) = C_{T'}(f)X(f)e^{j2\pi f\tau_0}, \quad |f| \leq \frac{1}{2T'} \tag{10-2-71}$$

Thus, we observe that the FSE compensates for the channel distortion in the received signal before the aliasing effects due to symbol rate sampling. In other words, $C_T(f)$ can compensate for any arbitrary timing phase.

The FSE output is sampled at the symbol rate $1/T$ and has the spectrum

$$\sum_k C_T\left(f - \frac{k}{T}\right) X\left(f - \frac{k}{T}\right) e^{j2\pi(f - k/T)\tau_0} \tag{10-2-72}$$

In effect, the optimum FSE is equivalent to the optimum linear receiver consisting of the matched filter followed by a symbol rate equalizer.

Let us now consider the adjustment of the tap coefficients in the FSE. The input to the FSE may be expressed as

$$y\left(\frac{kMT}{N}\right) = \sum_n I_n x\left(\frac{kMT}{N} - nT\right) + v\left(\frac{kMT}{N}\right) \tag{10-2-73}$$

In each symbol interval, the FSE produces an output of the form

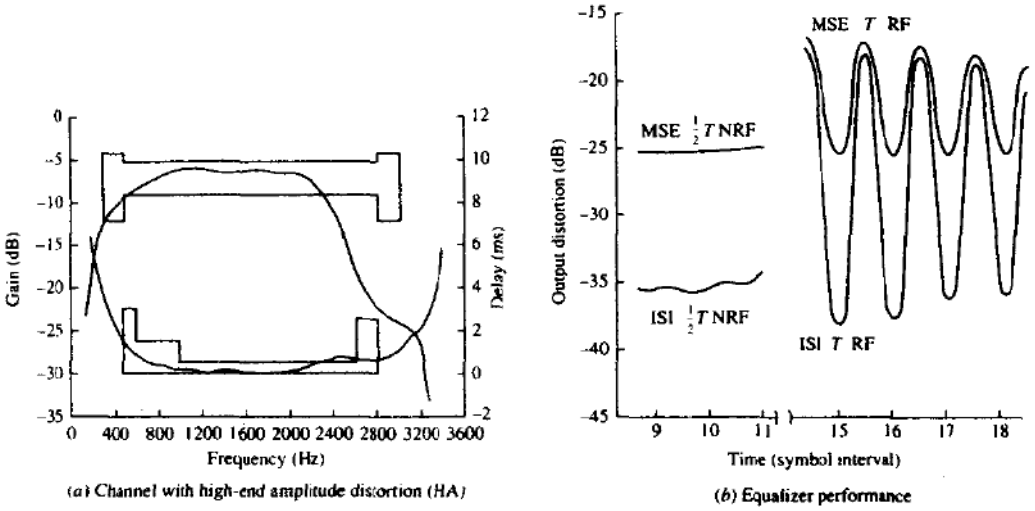
$$\hat{I}_k = \sum_{n=-K}^{K} c_n y\left(kT - \frac{nMT}{N}\right) \tag{10-2-74}$$

where the coefficients of the equalizer are selected to minimize the MSE. This optimization leads to a set of linear equations for the equalizer coefficients that have the solution

$$\mathbf{C}_{\text{opt}} = \mathbf{A}^{-1}\boldsymbol{\alpha} \tag{10-2-75}$$

where $\mathbf{A}$ is the covariance matrix of the input data and $\boldsymbol{\alpha}$ is the vector of cross-correlations. These equations are identical in form to those for the symbol rate equalizer, but there are some subtle differences. One is that $\mathbf{A}$ is Hermitian, but not Toeplitz. In addition, $\mathbf{A}$ exhibits periodicities that are inherent in a cyclostationary process, as shown by Qureshi (1985). As a result of the fractional spacing, some of the eigenvalues of $\mathbf{A}$ are nearly zero. Attempts have been made by Long *et al.* (1988a, b) to exploit this property in the coefficient adjustment.

An analysis of the performance of fractionally spaced equalizers, including their convergence properties, is given in a paper by Ungerboeck (1976). Simulation results demonstrating the effectiveness of the FSE over a symbol rate equalizer have also been given in the papers by Qureshi and Forney (1977) and Gitlin and Weinstein (1981). We cite two examples from these papers. First, Fig. 10-2-7 illustrates the performance of the symbol rate equalizer and a $\frac{1}{2}T$-FSE for a channel with high-end amplitude distortion, whose characteristics are also shown in this figure. The symbol-spaced equalizer was preceded with a filter matched to the transmitted pulse that had a (square-root) raised cosine spectrum with a 20% roll-off ($\beta = 0.2$). The FSE did not have any filter preceding it. The symbol rate was 2400 symbols/s and the modulation was QAM. The received SNR was 30 dB. Both equalizers had 31 taps; hence, the $\frac{1}{2}T$-FSE spanned one-half of the time interval of the



(*a*) Channel with high-end amplitude distortion (*HA*)

(*b*) Equalizer performance

FIGURE 10-2-7 T and $\frac{1}{2}T$ equalizer performance as a function of timing phase for 2400 symbols per second. (NRF indicates no receiver filter.) [*From Qureshi and Forney (1977). © 1977 IEEE.*]

symbol rate equalizer. Nevertheless, the FSE outperformed the symbol rate equalizer when the latter was optimized at the best sampling time. Furthermore, the FSE did not exhibit any sensitivity to timing phase, as illustrated in Fig. 10-2-7.

Similar results were obtained by Gitlin and Weinstein. For a channel with poor envelope delay characteristics, the SNR performance of the symbol rate equalizer and a $\frac{1}{2}T$-FSE are illustrated in Fig. 10-2-8. In this case, both equalizers had the same time span. The T-spaced equalizer had 24 taps while the FSE had 48 taps. The symbol rate was 2400 symbols/s and the data rate was 9600 bits/s with 16-QAM modulation. The signal pulse had a raised cosine spectrum with $\beta = 0.12$. Note again that the FSE outperformed the T-spaced equalizer by several decibels, even when the latter was adjusted for optimum

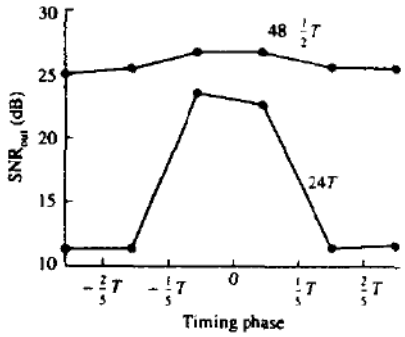


FIGURE 10-2-8 Performance of T and $\frac{1}{2}T$ equalizers as a function of timing phase for 2400 symbols/s 16-QAM on a channel with poor envelope delay. [*From Gitlin and Weinstein (1981). Reprinted with permission from* Bell System Technical Journal. *© 1981 AT & T.*]

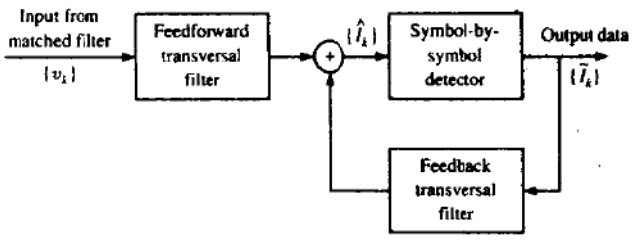


FIGURE 10-3-1 Structure of decision-feedback equalizer.

sampling. The results in these two papers clearly demonstrate the superior performance achieved with a fractionally spaced equalizer.

10-3 DECISION-FEEDBACK EQUALIZATION

The *decision-feedback equalizer* (DFE), depicted in Fig. 10-3-1, consists of two filters, a feedforward filter and a feedback filter. As shown, both have taps spaced at the symbol interval T. The input to the feedforward section is the received signal sequence $\{v_k\}$. In this respect, the feedforward filter is identical to the linear transversal equalizer described in Section 10-2. The feedback filter has as its input the sequence of decisions on previously detected symbols. Functionally, the feedback filter is used to remove that part of the intersymbol interference from the present estimate caused by previously detected symbols.

10-3-1 Coefficient Optimization

From the description given above, it follows that the equalizer output can be expressed as

$$\hat{I}_k = \sum_{j=-K_1}^{0} c_j v_{k-j} + \sum_{j=1}^{K_2} c_j \tilde{I}_{k-j} \tag{10-3-1}$$

where $\hat{I}_k$ is an estimate of the k th information symbol, $\{c_j\}$ are the tap coefficients of the filter, and $\{\tilde{I}_{k-1}, \ldots, \tilde{I}_{k-K_2}\}$ are previously detected symbols. The equalizer is assumed to have $(K_1 + 1)$ taps in its feedforward section and K_2 in its feedback section. It should be observed that this equalizer is nonlinear because the feedback filter contains previously detected symbols $\{\tilde{I}_k\}$.

Both the peak distortion criterion and the MSE criterion result in a mathematically tractable optimization of the equalizer coefficients, as can be concluded from the papers by George *et al.* (1971). Price (1972), Salz (1973), and Proakis (1975). Since the MSE criterion is more prevalent in practice, we focus our attention on it. Based on the assumption that previously detected symbols in the feedback filter are correct, the minimization of MSE

$$J(K_1, K_2) = E\,|I_k - \hat{I}_k|^2 \tag{10-3-2}$$

leads to the following set of linear equations for the coefficients of the feedforward filter:

$$\sum_{j=-K_1}^{0} \psi_{lj} c_j = f^*_{-l}, \qquad l = -K_1, \ldots, -1, 0 \tag{10-3-3}$$

where

$$\psi_{lj} = \sum_{m=0}^{-l} f^*_m f_{m+l-j} + N_0 \delta_{lj}, \qquad l, j = -K_1, \ldots, -1, 0 \tag{10-3-4}$$

The coefficients of the feedback filter of the equalizer are given in terms of the coefficients of the feedforward section by the following expression:

$$c_k = -\sum_{j=-K_1}^{0} c_j f_{k-j}, \qquad k = 1, 2, \ldots, K_2 \tag{10-3-5}$$

The values of the feedback coefficients result in complete elimination of intersymbol interference from previously detected symbols, provided that previous decisions are correct and that $K_2 \geq L$ (see Problem 10-9).

10-3-2 Performance Characteristics of DFE

We now turn our attention to the performance achieved with decision-feedback equalization. The exact evaluation of the performance is complicated to some extent by occasional incorrect decisions made by the detector, which then propagate down the feedback section. In the absence of decision errors, the minimum MSE is given as

$$J_{\min}(K_1) = 1 - \sum_{j=-K_1}^{0} c_j f_{-j} \tag{10-3-6}$$

By going to the limit ($K_1 \to \infty$) of an infinite number of taps in the feedforward filter, we obtain the smallest achievable MSE, denoted as $J_{\min}$. With some effort $J_{\min}$ can be expressed in terms of the spectral characteristics of the channel and additive noise, as shown by Salz (1973). This more desirable form for $J_{\min}$ is

$$J_{\min} = \exp\left\{\frac{T}{2\pi}\int_{-\pi/T}^{\pi/T} \ln\left[\frac{N_0}{X(e^{j\omega T}) + N_0}\right] d\omega\right\} \tag{10-3-7}$$

The corresponding output SNR is

$$\begin{aligned}\gamma_\infty &= \frac{1 - J_{\min}}{J_{\min}} \\ &= -1 + \exp\left\{\frac{T}{2\pi}\int_{-\pi/T}^{\pi/T} \ln\left[\frac{N_0 + X(e^{j\omega T})}{N_0}\right] d\omega\right\}\end{aligned} \tag{10-3-8}$$

We observe again that, in the absence of intersymbol interference, $X(e^{j\omega T}) = 1$ and, hence, $J_{\min} = N_0/(1 + N_0)$. The corresponding output SNR is $\gamma_\infty = 1/N_0$.

Example 10-3-1

It is interesting to compare the value of $J_{\min}$ for the decision-feedback equalizer with the value of $J_{\min}$ obtained with the linear MSE equalizer. For example, ler us consider the discrete-time equivalent channel consisting of two taps f_0 and f_1. The minimum MSE for this channel is

$$\begin{aligned} J_{\min} &= \exp\left\{\frac{T}{2\pi}\int_{-\pi/T}^{\pi/T} \ln\left[\frac{N_0}{1+N_0+2|f_0||f_1|\cos(\omega T+\theta)}\right]d\omega\right\} \\ &= N_0 \exp\left[-\frac{1}{2\pi}\int_{-\pi}^{\pi} \ln(1+N_0+2|f_0||f_1|\cos\omega)\,d\omega\right] \\ &= \frac{2N_0}{1+N_0+\sqrt{(1+N_0)^2-4|f_0f_1|^2}} \end{aligned} \tag{10-3-9}$$

Note that $J_{\min}$ is maximized when $|f_0| = |f_1| = \sqrt{\frac{1}{2}}$. Then

$$\begin{aligned} J_{\min} &= \frac{2N_0}{1+N_0+\sqrt{(1+N_0)^2-1}} \\ &\approx 2N_0, \qquad N_0 \ll 1 \end{aligned} \tag{10-3-10}$$

The corresponding output SNR is

$$\gamma_\infty \approx \frac{1}{2N_0}, \qquad N_0 \ll 1 \tag{10-3-11}$$

Therefore, there is a 3 dB degradation in output SNR due to the presence of intersymbol interference. In comparison, the performance loss for the linear equalizer is very severe. Its output SNR as given by (10-2-53) is $\gamma_\infty \approx (2/N_0)^{1/2}$ for $N_0 \ll 1$.

Example 10-3-2

Consider the exponentially decaying channel characteristic of the form

$$f_k = (1-a^2)^{1/2}a^k, \qquad k = 0, 1, 2, \ldots \tag{10-3-12}$$

where $a < 1$. The output SNR of the decision-feedback equalizer is

$$\begin{aligned} \gamma_\infty &= -1 + \exp\left\{\frac{1}{2\pi}\int_{-\pi}^{\pi} \ln\left[\frac{1+a^2+(1-a^2)/N_0-2a\cos\omega}{1+a^2-2a\cos\omega}\right]d\omega\right\} \\ &= -1 + \frac{1}{2N_0}\{1-a^2+N_0(1+a^2)+\sqrt{[1-a^2+N_0(1+a^2)]^2-4a^2N_0^2}\} \\ &\approx \frac{(1-a^2)[1+N_0(1+a^2)/(1-a^2)]-N_0}{N_0} \\ &\approx \frac{1-a^2}{N_0}, \qquad N_0 \ll 1 \end{aligned} \tag{10-3-13}$$

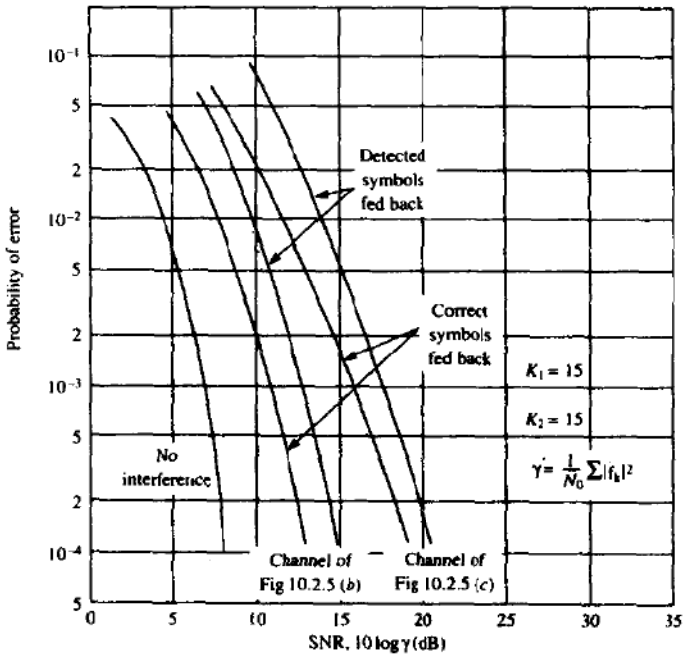
Thus, the loss in SNR is $10 \log_{10}(1 - a^2)$ dB. In comparison, the linear equalizer has a loss of $10 \log_{10}[(1 - a^2)/(1 + a^2)]$ dB.

These results illustrate the superiority of the decision-feedback equalizer over the linear equalizer when the effect of decision errors on performance is neglected. It is apparent that a considerable gain in performance can be achieved relative to the linear equalizer by the inclusion of the decision-feedback section, which eliminates the intersymbol interference from previously detected symbols.

One method of assessing the effect of decision errors on the error rate performance of the decision-feedback equalizer is Monte Carlo simulation on a digital computer. For purposes of illustration, we offer the following results for binary PAM signaling through the equivalent discrete-time channel models shown in Figs 10-2-5(b) and (c).

The results of the simulation are displayed in Fig. 10-3-2. First of all, a comparison of these results with those presented in Fig. 10-2-4 leads us to conclude that the decision-feedback equalizer yields a significant improvement in performance relative to the linear equalizer having the same number of taps. Second, these results indicate that there is still a significant degradation in performance of the decision-feedback equalizer due to the residual intersymbol interference, especially on channels with severe distortion such as the one

FIGURE 10-3-2 *Performance of decision-feedback equalizer with and without error propagation.*

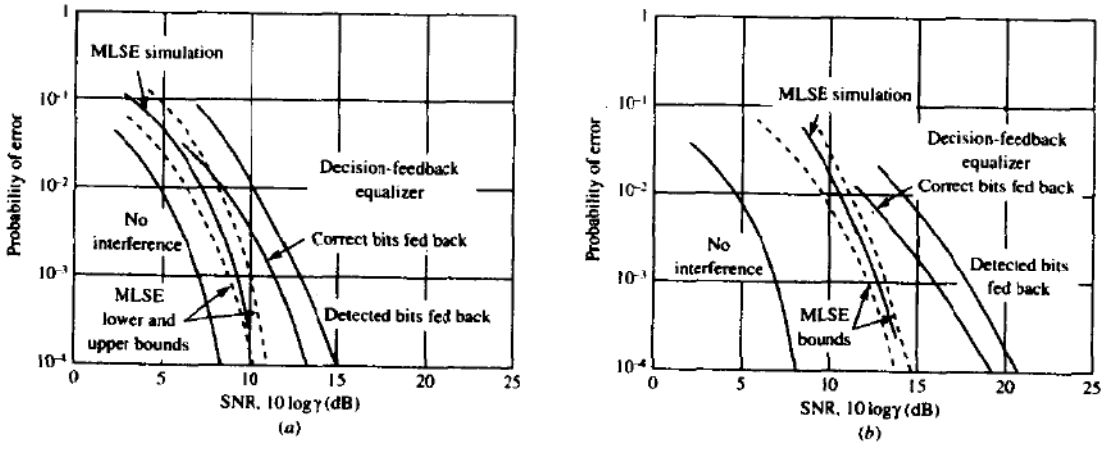


shown in Fig. 10-2-5(*c*). Finally, the performance loss due to incorrect decisions being fed back is 2 dB, approximately, for the channel responses under consideration. Additional results on the probability of error for a decision-feedback equalizer with error propagation may be found in the papers by Duttweiler *et al.* (1974) and Beaulieu (1992).

The structure of the DFE that is analyzed above employs a T-spaced filter for the feedforward section. The optimality of such a structure is based on the assumption that the analog filter preceding the DFE is matched to the channel-corrupted pulse response and its output is sampled at the optimum time instant. In practice, the channel response is not known a priori, so it is not possible to design an ideal matched filter. In view of this difficulty, it is customary in practical applications to use a fractionally spaced feedforward filter. Of course, the feedback filter tap spacing remains at T. The use of the FSE for the feedforward filter eliminates the system sensitivity to a timing error.

Performance Comparison with MLSE We conclude this subsection on the performance of the DFE by comparing its performance against that of MLSE. For the two-path channel with $f_0 = f_1 = \sqrt{\frac{1}{2}}$, we have shown that MLSE suffers no SNR loss while the decision-feedback equalizer suffers a 3 dB loss. On channels with more distortion, the SNR advantage of MLSE over decision-feedback equalization is even greater. Figure 10-3-3 illustrates a comparison of the error rate performance of these two equalization techniques, obtained via Monte Carlo simulation, for binary PAM and the channel characteristics shown in Figs 10-2-5(*b*) and (*c*). The error rate curves for the two methods have different slopes; hence the difference in SNR increases as the error

FIGURE 10-3-3 Comparison of performance between MLSE and decision-feedback equalization for channel characteristics shown (*a*) in Fig. 10-2-5(*b*) and (*b*) in Fig. 10-2-5(*c*).



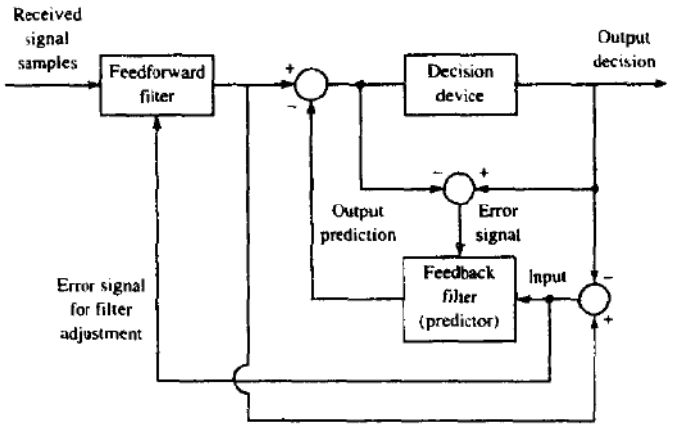


FIGURE 10-3-4 Block diagram of predictive DFE.

probability decreases. As a benchmark, the error rate for the AWGN channel with no intersymbol interference is also shown in Fig. 10-3-3.

10-3-3 Predictive Decision-Feedback Equalizer

Belfiore and Park (1979) proposed another DFE structure that is equivalent to the one shown in Fig. 10-3-1 under the condition that the feedforward filter has an infinite number of taps. This structure consists of a FSE as a feedforward filter and a linear predictor as a feedback filter, as shown in the configuration given in Fig. 10-3-4. Let us briefly consider the performance characteristics of this equalizer.

First of all, the noise at the output of the infinite length feedforward filter has the power spectral density

$$\frac{N_0 X(e^{j\omega T})}{|N_0 + X(e^{j\omega T})|^2}, \qquad |\omega| \leq \frac{\pi}{T} \tag{10-3-14}$$

The residual intersymbol interference has the power spectral density

$$\left|1 - \frac{X(e^{j\omega T})}{N_0 + X(e^{j\omega T})}\right|^2 = \frac{N_0^2}{|N_0 + X(e^{j\omega T})|^2}, \qquad |\omega| \leq \frac{\pi}{T} \tag{10-3-15}$$

The sum of these two spectra represents the power spectral density of the total noise and intersymbol interference at the output of the feedforward filter. Thus, on adding (10-3-14) and (10-3-15), we obtain

$$E(\omega) = \frac{N_0}{N_0 + X(e^{j\omega T})}, \qquad |\omega| \leq \frac{\pi}{T} \tag{10-3-16}$$

As we have observed previously, if $X(e^{j\omega T}) = 1$, the channel is ideal and,

hence, it is not possible to reduce the MSE any further. On the other hand, if there is channel distortion, the power in the error sequence at the output of the feedforward filter can be reduced by means of linear prediction based on past values of the error sequence.

If $\mathcal{B}(\omega)$ represents the frequency response of the infinite length feedback predictor, i.e.,

$$\mathcal{B}(\omega) = \sum_{n=1}^{\infty} b_n e^{-j\omega nT} \tag{10-3-17}$$

then the error at the output of the predictor is

$$E(\omega) - E(\omega)\mathcal{B}(\omega) = E(\omega)[1 - \mathcal{B}(\omega)] \tag{10-3-18}$$

The minimization of the mean square value of this error, i.e.,

$$J = \frac{1}{2\pi}\int_{-\pi/T}^{\pi/T} |1 - \mathcal{B}(\omega)|^2 |E(\omega)|^2 \, d\omega \tag{10-3-19}$$

over the predictor coefficients $\{b_n\}$ yields the optimum predictor in the form

$$\mathcal{B}(\omega) = 1 - \frac{G(\omega)}{g_0} \tag{10-3-20}$$

where $G(\omega)$ is the solution to the spectral factorization

$$G(\omega)G^*(-\omega) = \frac{1}{|E(\omega)|^2} \tag{10-3-21}$$

and

$$G(\omega) = \sum_{n=0}^{\infty} g_n e^{-j\omega nT} \tag{10-3-22}$$

The output of the infinite length linear predictor is a white noise sequence with power spectral density $1/g_0^2$ and the corresponding minimum MSE is given by (10-3-7). Therefore, the MSE performance of the infinite-length predictive DFE is identical to the conventional DFE.

Although these two DFE structures result in equivalent performance if their lengths are infinite, the predictive DFE is suboptimum if the lengths of the two filters are finite. The reason for the optimality of the conventional DFE is relatively simple. The optimization of its tap coefficients in the feedforward and feedback filters is done jointly. Hence, it yields the minimum MSE. On the other hand, the optimizations of the feedforward filter and the feedback predictor in the predictive DFE are done separately. Hence, its MSE is at least as large as that of the conventional DFE. In spite of this suboptimality of the predictive DFE, it is suitable as an equalizer for trellis-coded signals, where the conventional DFE is not as suitable, as described in the next chapter.

10-4 BIBLIOGRAPHICAL NOTES AND REFERENCES

Channel equalization for digital communications was developed by Lucky (1965, 1966), who focused on linear equalizers that were optimized using the peak distortion criterion. The mean square error criterion for optimization of the equalizer coefficients was proposed by Widrow (1966).

Decision-feedback equalization was proposed and analyzed by Austin (1967). Analyses of the performance of the DFE can be found in the papers by Monsen (1971), George *et al.* (1971), Price (1972), Salz (1973), Duttweiler *et al.* (1974), and Altekar and Beaulieu (1993).

The use of the Viterbi algorithm as the optimal maximum-likelihood sequence estimator for symbols corrupted by ISI was proposed and analyzed by Forney (1972) and Omura (1971). Its use for carrier-modulated signals was considered by Ungerboeck (1974) and MacKenchnie (1973).

PROBLEMS

10-1 In a binary PAM system, the input to the detector is

$$y_m = a_m + n_m + i_m$$

where $a_m = \pm 1$ is the desired signal, n_m is a zero-mean Gaussian random variable with variance σ_n^2, and i_m represents the ISI due to channel distortion. The ISI term is a random variable that takes the values $-\frac{1}{2}$, 0, and $\frac{1}{2}$ with probabilities $\frac{1}{4}$, $\frac{1}{2}$, and $\frac{1}{4}$, respectively. Determine the average probability of error as a function of σ_n^2.

10-2 In a binary PAM system, the clock that specifies the sampling of the correlator output is offset from the optimum sampling time by 10%.

a If the signal pulse used is rectangular, determine the loss in SNR due to the mistiming.

b Determine the amount of ISI introduced by the mistiming and determine its effect on performance.

10-3 The frequency response characteristic of a lowpass channel can be approximated by

$$H(f) = \begin{cases} 1 + \alpha \cos 2\pi f t_0 & (|\alpha| < 1,\ |f| \le W) \\ 0 & (\text{otherwise}) \end{cases}$$

where W is the channel bandwidth. An input signal $s(t)$ whose spectrum is bandlimited to W Hz is passed through the channel.

a Show that

$$y(t) = s(t) + \tfrac{1}{2}\alpha[s(t - t_0) + s(t + t_0)]$$

Thus, the channel produces a pair of echoes.

b Suppose that the received signal $y(t)$ is passed through a filter matched to $s(t)$. Determine the output of the matched filter at $t = kT$, $k = 0, \pm 1, \pm 2, \ldots$, where T is the symbol duration.

c What is the ISI pattern resulting from the channel if $t_0 = T$?

10-4 A wireline channel of length 1000 km is used to transmit data by means of binary

PAM. Regenerative repeaters are spaced 50 km apart along the system. Each segment of the channel has an ideal (constant) frequency response over the frequency band $0 \leq f \leq 1200$ Hz and an attenuation of 1 dB/km. The channel noise is AWGN.

a What is the highest bit rate that can be transmitted without ISI?

b Determine the required $\mathscr{E}_b/N_0$ to achieve a bit error of $P_2 = 10^{-7}$ for each repeater.

c Determine the transmitted power at each repeater to achieve the desired $\mathscr{E}_b/N_0$, where $N_0 = 4.1 \times 10^{-21}$ W/Hz.

10-5 Prove the relationship in (10-1-13) for the autocorrelation of the noise at the output of the matched filter.

10-6 In the case of PAM with correlated noise, the correlation metrics in the Viterbi algorithm may be expressed in general as (Ungerboeck, 1974)

$$CM(\mathbf{I}) = 2\sum_n I_n r_n - \sum_n \sum_m I_n I_m x_{n-m}$$

where $x_n = x(nT)$ is the sampled signal output of the matched filter, $\{I_n\}$ is the data sequence, and $\{r_n\}$ is the received signal sequence at the output of the matched filter. Determine the metric for the duobinary signal.

10-7 Consider the use of a (square-root) raised cosine signal pulse with a roll-off factor of unity for transmission of binary PAM over an ideal bandlimited channel that passes the pulse without distortion. Thus, the transmitted signal is

$$v(t) = \sum_{k=-\infty}^{\infty} I_k g_T(t - kT_b)$$

where the signal interval $T_b = \frac{1}{2}T$. Thus, the symbol rate is double of that for no ISI.

a Determine the ISI values at the output of a matched filter demodulator.

b Sketch the trellis for the maximum-likelihood sequence detector and label the states.

10-8 A binary antipodal signal is transmitted over a nonideal band-limited channel, which introduces ISI over two adjacent symbols. For an isolated transmitted signal pulse $s(t)$, the (noise-free) output of the demodulator is $\sqrt{\mathscr{E}_b}$ at $t = T$, $\sqrt{\mathscr{E}_b}/4$ at $t = 2T$, and zero for $t = kT$, $k > 2$, where $\mathscr{E}_b$ is the signal energy and T is the signaling interval.

a Determine the average probability of error, assuming that the two signals are equally probable and the additive noise is white and gaussian.

b By plotting the error probability obtained in (a) and that for the case of no ISI, determine the relative difference in SNR of the error probability of 10^{-6}.

10-9 Derive the expression in (10-3-5) for the coefficients in the feedback filter of the DFE.

10-10 Binary PAM is used to transmit information over an unequalized linear filter channel. When $a = 1$ is transmitted, the noise-free output of the demodulator is

$$x_m = \begin{cases} 0.3 & (m = 1) \\ 0.9 & (m = 0) \\ 0.3 & (m = -1) \\ 0 & \text{(otherwise)} \end{cases}$$

a Design a three-tap zero-forcing linear equalizer so that the output is

$$q_m = \begin{cases} 1 & (m = 0) \\ 0 & (m = \pm 1) \end{cases}$$

b Determine q_m for $m = \pm 2, \pm 3$, by convolving the impulse response of the equalizer with the channel response.

10-11 The transmission of a signal pulse with a raised cosine spectrum through a channel results in the following (noise-free) sampled output from the demodulator:

$$x_k = \begin{cases} -0.5 & (k = -2) \\ 0.1 & (k = -1) \\ 1 & (k = 0) \\ -0.2 & (k = 1) \\ 0.05 & (k = 2) \\ 0 & (\text{otherwise}) \end{cases}$$

a Determine the tap coefficients of a three-tap linear equalizer based on the zero-forcing criterion.

b For the coefficients determined in (a), determine the output of the equalizer for the case of the isolated pulse. Thus, determine the residual ISI and its span in time.

10-12 A nonideal band-limited channel introduces ISI over three successive symbols. The (noise-free) response of the matched filter demodulator sampled at the sampling time kT is

$$\int_{-\infty}^{\infty} s(t)s(t - kT)\,dt = \begin{cases} \mathscr{E}_b & (k = 0) \\ 0.9\mathscr{E}_b & (k = \pm 1) \\ 0.1\mathscr{E}_b & (k = \pm 2) \\ 0 & (\text{otherwise}) \end{cases}$$

a Determine the tap coefficients of a three-tap linear equalizer that equalizes the channel (received signal) response to an equivalent partial response (duobinary) signal

$$y_k = \begin{cases} \mathscr{E}_b & (k = 0, 1) \\ 0 & (\text{otherwise}) \end{cases}$$

b Suppose that the linear equalizer in (a) is followed by a Viterbi sequence detector for the partial signal. Give an estimate of the error probability if the additive noise is white and gaussian, with power spectral density $\frac{1}{2}N_0$ W/Hz.

***10-13** Determine the tap weight coefficients of a three-tap zero-forcing equalizer if the ISI spans three symbols and is characterized by the values $x(0) = 1, x(-1) = 0.3$, $x(1) = 0.2$. Also determine the residual ISI at the output of the equalizer for the optimum tap coefficients.

10-14 In line-of-sight microwave radio transmission, the signal arrives at the receiver via two propagation paths: the direct path and a delayed path that occurs due to signal reflection from surrounding terrain. Suppose that the received signal has the form

$$r(t) = s(t) + \alpha s(t - T) + n(t)$$

where $s(t)$ is the transmitted signal, α is the attenuation ($\alpha < 1$) of the secondary path and $n(t)$ is AWGN.

a Determine the output of the demodulator at $t = T$ and $t = 2T$ that employs a filter matched to $s(t)$.

b Determine the probability of error for a symbol-by-symbol detector if the transmitted signal is binary antipodal and the detector ignores the ISI.

c What is the error-rate performance of a simple (one-tap) DFE that estimates α and removes the ISI? Sketch the detector structure that employs a DFE.

10-15 Repeat Problem 10-10 using the MMSE as the criterion for optimizing the tap coefficients. Assume that the noise power spectral density is 0.1 W/Hz.

10-16 In a magnetic recording channel, where the readback pulse resulting from a positive transition in the write current has the form

$$p(t) = \left[1 + \left(\frac{2t}{T_{50}}\right)^2\right]^{-1}$$

a linear equalizer is used to equalize the pulse to a partial response. The parameter T_{50} is defined as the width of the pulse at the 50% amplitude level. The bit rate is $1/T_b$ and the ratio of $T_{50}/T_b = \Delta$ is the normalized density of the recording. Suppose the pulse is equalized to the partial-response values

$$x(nT) = \begin{cases} 1 & (n = -1, 1) \\ 2 & (n = 0) \\ 0 & (\text{otherwise}) \end{cases}$$

where $x(t)$ represents the equalized pulse shape.

a Determine the spectrum $X(f)$ of the band-limited equalized pulse.

b Determine the possible output levels at the detector, assuming that successive transitions can occur at the rate $1/T_b$.

c Determine the error rate performance of the symbol-by-symbol detector for this signal, assuming that the additive noise is zero-mean gaussian with variance σ^2.

10-17 Sketch the trellis for the Viterbi detector of the equalized signal in Problem 10-16 and label all the states. Also, determine the minimum euclidean distance between merging paths.

10-18 Consider the problem of equalizing the discrete-time equivalent channel shown in Fig. P10-18. The information sequence $\{I_n\}$ is binary (± 1) and uncorrelated.

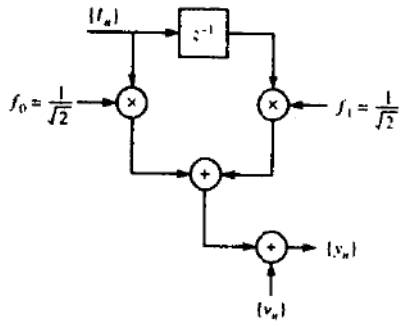


FIGURE P10-18

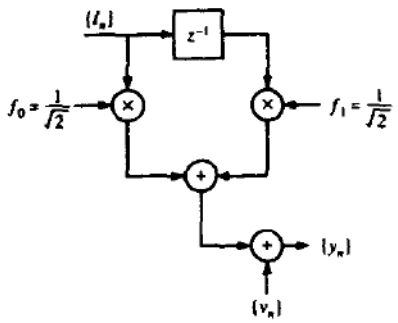


FIGURE P10-21

The additive noise $\{v_n\}$ is white and real-valued, with variance N_0. The received sequence $\{y_n\}$ is processed by a linear three-tap equalizer that is optimized on the basis of the MSE criterion.

a Determine the optimum coefficients of the equalizer as a function of N_0.

b Determine the three eigenvalues λ_1, λ_2, and λ_3 of the covariance matrix Γ and the corresponding (normalized to unit length) eigenvectors $\mathbf{v}_1$, $\mathbf{v}_2$, $\mathbf{v}_3$.

c Determine the minimum MSE for the three-tap equalizer as a function of N_0.

d Determine the output SNR for the three-tap equalizer as a function of N_0. How does this compare with the output SNR for the infinite-tap equalizer? For example, evaluate the output SNR for these two equalizers when $N_0 = 0.1$.

10-19 Use the orthogonality principle to derive the equations for the coefficients in a decision-feedback equalizer based on the MSE criterion and given by (10-3-3) and (10-3-5).

10-20 Suppose that the discrete-time model for the intersymbol interference is characterized by the tap coefficients $f_0, f_1, \ldots, f_L$. From the equations for the tap coefficients of a decision-feedback equalizer (DFE), show that only L taps are needed in the feedback filter of the DFE. That is, if $\{c_k\}$ are the coefficients of the feedback filter then $c_k = 0$ for $k \geq L + 1$.

10-21 Consider the channel model shown in Fig. P10-21. $\{v_n\}$ is a real-valued white-noise sequence with zero mean and variance N_0. Suppose the channel is to be equalized by DFE having a two-tap feedforward filter (c_0, c_{-1}) and a one-tap feedback filter (c_1). The $\{c_i\}$ are optimized using the MSE criterion.

a Determine the optimum coefficients and their approximate values for $N_0 \ll 1$.

b Determine the exact value of the minimum MSE and a first-order approximation appropriate to the case $N_0 \ll 1$.

c Determine the exact value of the output SNR for the three-tap equalizer as a function of N_0 and a first-order approximation appropriate to the case $N_0 \ll 1$.

d Compare the results in (b) and (c) with the performance of the infinite-tap DFE.

e Evaluate and compare the exact values of the output SNR for the three-tap and infinite-tap DFE in the special cases where $N_0 = 0.1$ and 0.01. Comment on how well the three-tap equalizer performs relative to the infinite-tap equalizer.

10-22 A pulse and its (raised-cosine) spectral characteristic are shown in Fig. P10-22. This pulse is used for transmitting digital information over a band-limited channel at a rate $1/T$ symbols/s.

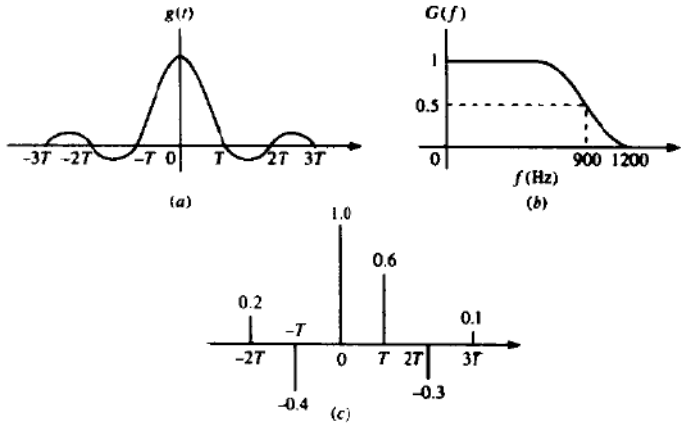


FIGURE P10-22

a What is the roll-off factor β?

b What is the pulse rate?

c The channel distorts the signal pulses. Suppose the sampled values of the filtered received pulse $x(t)$ are as shown in Fig. P10-22(c) It is obvious that there are five interfering signal components. Give the sequence of +1s and −1s that will cause the largest (destructive or constructive) interference and the corresponding value of the interference (the peak distortion).

d What is the probability of occurrence of the worst sequence obtained in (c), assuming that all binary digits are equally probable and independent?

10-23 A time-dispersive channel having an impulse response $h(t)$ is used to transmit four-phase PSK at a rate $R = 1/T$ symbols/s. The equivalent discrete-time channel is shown in Fig. P10-23. The sequence $\{\eta_k\}$ is a white noise sequence having zero mean and variance $\sigma^2 = N_0$.

a What is the sampled autocorrelation function sequence $\{x_k\}$ defined by

$$x_k = \int_{-\infty}^{\infty} h^*(t)h(t + kT)\,dt$$

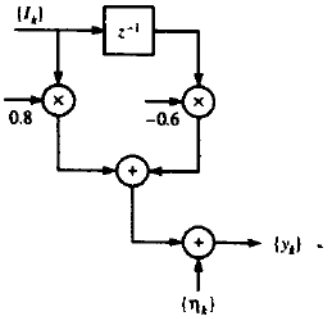


FIGURE P10-23

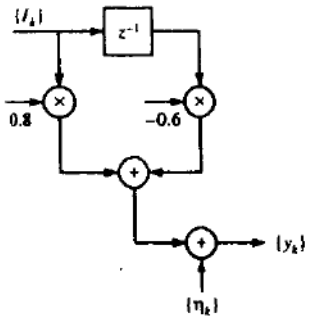


FIGURE P10-24

for this channel?

b The minimum MSE performance of a linear equalizer and a decision-feedback equalizer having an infinite number of taps depends on the *folded spectrum of the channel*

$$\frac{1}{T}\sum_{n=-\infty}^{\infty}\left|H\left(\omega+\frac{2\pi n}{T}\right)\right|^2$$

where $H(\omega)$ is the Fourier transform of $h(t)$. Determine the folded spectrum of the channel given above.

c Use your answer in (b) to express the minimum MSE of a linear equalizer in terms of the folded spectrum of the channel. (You may leave your answer in integral form.)

d Repeat (c) for an infinite-tap decision-feedback equalizer.

10-24 Consider a four-level PAM system with possible transmitted levels, 3, 1, −1, and −3. The channel through which the data are transmitted introduces intersymbol interference over two successive symbols. The equivalent discrete-time channel model is shown in Fig. P10-24. $\{\eta_k\}$ is a sequence of real-valued independent zero-mean gaussian noise variables with variance $\sigma^2 = N_0$. The received sequence is

$$\begin{aligned} y_1 &= 0.8I_1 + n_1 \\ y_2 &= 0.8I_2 - 0.6I_1 + n_2 \\ y_3 &= 0.8I_3 - 0.6I_2 + n_3 \\ &\vdots \\ y_k &= 0.8I_k - 0.6I_{k-1} + n_k \end{aligned}$$

a Sketch the tree structure, showing the possible signal sequences for the received signals y_1, y_2 and y_3.

b Suppose the Viterbi algorithm is used to detect the information sequence. How many probabilities must be computed at each stage of the algorithm?

c How many surviving sequences are there in the Viterbi algorithm for this channel?

d Suppose that the received signals are

$$y_1 = 0.5, \quad y_2 = 2.0, \quad y_3 = -1.0$$

Determine the surviving sequences through stage y_3 and the corresponding metrics.

e Give a tight upper bound for the probability of error for four-level PAM transmitted over this channel.

10-25 A transversal equalizer with K taps has an impulse response

$$e(t) = \sum_{k=0}^{K-1} c_k \delta(t - kT)$$

where T is the delay between adjacent taps, and a transfer function

$$E(z) = \sum_{k=0}^{K-1} c_k z^{-k}$$

The *discrete Fourier transform* (DFT) of the equalizer coefficients $\{c_k\}$ is defined as

$$E_n \equiv E(z)|_{z=e^{j2\pi n/K}} = \sum_{k=0}^{K-1} c_k e^{-j2\pi kn/K}, \quad n = 0, 1, \ldots, K-1$$

The *inverse DFT* is defined as

$$b_k = \frac{1}{K} \sum_{n=0}^{K-1} E_n e^{j2\pi nk/K}, \quad k = 0, 1, \ldots, K-1$$

a Show that $b_k = c_k$, by substituting for E_n in the above expression.

b From the relations given above, derive an equivalent filter structure having the z transform

$$E(z) = \underbrace{\frac{1 - z^{-K}}{K}}_{E_1(z)} \underbrace{\sum_{n=0}^{K-1} \frac{E_n}{1 - e^{j2\pi n/K} z^{-1}}}_{E_2(z)}$$

c If $E(z)$ is considered as two separate filters $E_1(z)$ and $E_2(z)$ in cascade, sketch a block diagram for each of the filters, using z^{-1} to denote a unit of delay.

d In the transversal equalizer, the adjustable parameters are the equalizer coefficients $\{c_k\}$. What are the adjustable parameters of the equivalent equalizer in (b), and how are they related to $\{c_k\}$?

11

ADAPTIVE EQUALIZATION

In Chapter 10, we introduced both optimum and suboptimum receivers that compensate for ISI in the transmission of digital information through band-limited, nonideal channels. The optimum receiver employed maximum-likelihood sequence estimation for detecting the information sequence from the samples of the demodulation filter. The suboptimum receivers employed either a linear equalizer or a decision-feedback equalizer.

In the development of the three equalization methods, we implicitly assumed that the channel characteristics, either the impulse response or the frequency response, were known at the receiver. However, in most communication systems that employ equalizers, the channel characteristics are unknown a priori and, in many cases, the channel response is time-variant. In such a case, the equalizers are designed to be adjustable to the channel response and, for time-variant channels, to be adaptive to the time variations in the channel response.

In this chapter, we present algorithms for automatically adjusting the equalizer coefficients to optimize a specified performance index and to adaptively compensate for time variations in the channel characteristics. We also analyze the performance characteristics of the algorithm, including their rate of convergence and their computational complexity.

11-1 ADAPTIVE LINEAR EQUALIZER

In the case of the linear equalizer, recall that we considered two different criteria for determining the values of the equalizer coefficients $\{c_k\}$. One criterion was based on the minimization of the peak distortion at the output of

the equalizer, which is defined by (10-2-4). The other criterion was based on the minimization of the mean-square error at the output of the equalizer, which is defined by (10-2-25). Below, we describe two algorithms for performing the optimization automatically and adaptively.

11-1-1 The Zero-Forcing Algorithm

In the peak-distortion criterion, the peak distortion $\mathscr{D}(\mathbf{c})$, given by (10-2-22), is minimized by selecting the equalizer coefficients $\{c_k\}$. In general, there is no simple computational algorithm for performing this optimization, except in the special case where the peak distortion at the input to the equalizer, defined as $\mathscr{D}_0$ in (10-2-23), is less than unity. When $\mathscr{D}_0 < 1$, the distortion $\mathscr{D}(\mathbf{c})$ at the output of the equalizer is minimized by forcing the equalizer response $q_n = 0$, for $1 \leq |n| \leq K$, and $q_0 = 1$. In this case, there is a simple computational algorithm, called the zero-forcing algorithm, that achieves these conditions.

The zero-forcing solution is achieved by forcing the cross-correlation between the error sequence $\varepsilon_k = I_k - \hat{I}_k$ and the desired information sequence $\{I_k\}$ to be zero for shifts in the range $0 \leq |n| \leq K$. The demonstration that this leads to the desired solution is quite simple. We have

$$\begin{aligned} E(\varepsilon_k I_{k-j}^*) &= E[(I_k - \hat{I}_k) I_{k-j}^*] \\ &= E(I_k I_{k-j}^*) - E(\hat{I}_k I_{k-j}^*), \quad j = -K, \ldots, K \end{aligned} \tag{11-1-1}$$

We assume that the information symbols are uncorrelated, i.e., $E(I_k I_j^*) = \delta_{kj}$, and that the information sequence $\{I_k\}$ is uncorrelated with the additive noise sequence $\{\eta_k\}$. For $\hat{I}_k$, we use the expression given in (10-2-41). Then, after taking the expected values in (11-1-1), we obtain

$$E(\varepsilon_k I_{k-j}^*) = \delta_{j0} - q_j, \quad j = -K, \ldots, K \tag{11-1-2}$$

Therefore, the conditions

$$E(\varepsilon_k I_{k-j}^*) = 0, \quad j = -K, \ldots, K \tag{11-1-3}$$

are fulfilled when $q_0 = 1$ and $q_n = 0$, $1 \leq |n| \leq K$.

When the channel response is unknown, the cross-correlations given by (11-1-1) are also unknown. This difficulty can be circumvented by transmitting a known training sequence $\{I_k\}$ to the receiver, which can be used to estimate the cross-correlation by substituting time averages for the ensemble averages given in (11-1-1). After the initial training, which will require the transmission of a training sequence of some predetermined length that equals or exceeds the equalizer length, the equalizer coefficients that satisfy (11-1-3) can be determined.

A simple recursive algorithm for adjusting the equalizer coefficients is

$$c_j^{(k+1)} = c_j^{(k)} + \Delta \varepsilon_k I_{k-j}^*, \quad j = -K, \ldots, -1, 0, 1, \ldots, K \qquad (11\text{-}1\text{-}4)$$

where $c_j^{(k)}$ is the value of the jth coefficient at time $t = kT$, $\varepsilon_k = I_k - \hat{I}_k$ is the error signal at time $t = kT$, and Δ is a scale factor that controls the rate of adjustment, as will be explained later in this section. This is the *zero-forcing algorithm.* The term $\varepsilon_k I_{k-j}^*$ is an estimate of the cross-correlation (ensemble average) $E(\varepsilon_k I_{k-j}^*)$. The averaging operation of the cross-correlation is accomplished by means of the recursive first-order difference equation algorithm in (11-1-4), which represents a simple discrete-time integrator.

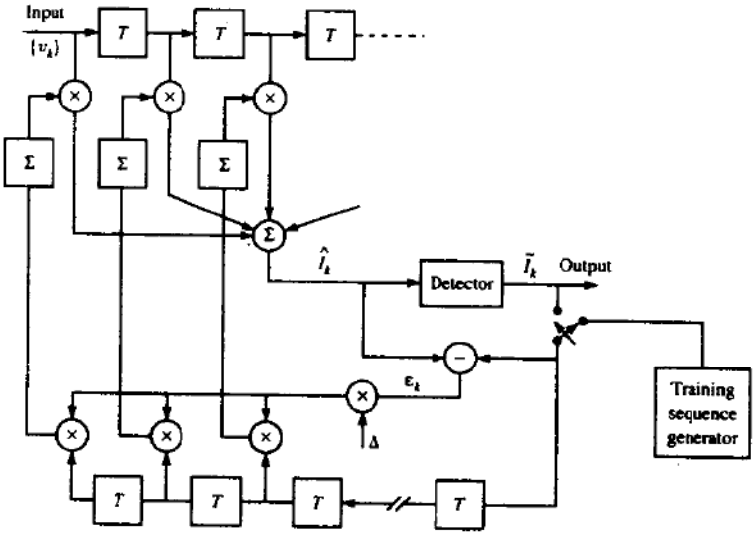
Following the training period, after which the equalizer coefficients have converged to their optimum values, the decisions at the output of the detector are generally sufficiently reliable so that they may be used to continue the coefficient adaptation process. This is called a *decision-directed mode* of adaptation. In such a case, the cross-correlations in (11-1-4) involve the error signal $\tilde{\varepsilon}_k = \tilde{I}_k - \hat{I}_k$ and the detected output sequence $\tilde{I}_{k-j}$, $j = -K, \ldots, K$. Thus, in the adaptive mode, (11-1-4) becomes

$$c_j^{(k+1)} = c_j^{(k)} + \Delta \tilde{\varepsilon}_k \tilde{I}_{k-j}^* \qquad (11\text{-}1\text{-}5)$$

Figure 11-1-1 illustrates the zero-forcing equalizer in the training mode and the adaptive mode of operation.

The characteristics of the zero-forcing algorithm are similar to those of the LMS algorithm, which minimizes the MSE and which is described in detail in the following section.

FIGURE 11-1-1 An adaptive zero-forcing equalizer.



11-1-2 The LMS Algorithm

In the minimization of the MSE, treated in Section 10-2-2, we found that the optimum equalizer coefficients are determined from the solution of the set of linear equations, expressed in matrix form as

$$\mathbf{\Gamma C} = \boldsymbol{\xi} \tag{11-1-6}$$

where $\mathbf{\Gamma}$ is the $(2K+1)\times(2K+1)$ covariance matrix of the signal samples $\{v_k\}$, $\mathbf{C}$ is the column vector of $(2K+1)$ equalizer coefficients, and $\boldsymbol{\xi}$ is a $(2K+1)$-dimensional column vector of channel filter coefficients. The solution for the optimum equalizer coefficients vector $\mathbf{C}_{opt}$ can be determined by inverting the covariance matrix $\mathbf{\Gamma}$, which can be efficiently performed by use of the Levinson–Durbin algorithm described in Appendix A.

Alternatively, an iterative procedure that avoids the direct matrix inversion may be used to compute $\mathbf{C}_{opt}$. Probably the simplest iterative procedure is the method of steepest descent, in which one begins by arbitrarily choosing the vector $\mathbf{C}$, say as $\mathbf{C}_0$. This initial choice of coefficients corresponds to some point on the quadratic MSE surface in the $(2K+1)$-dimensional space of coefficients. The gradient vector $\mathbf{G}_0$, having the $2K+1$ gradient components $\frac{1}{2}\partial J/\partial c_{0k}$, $k=-K,\ldots,-1,0,1,\ldots,K$, is then computed at this point on the MSE surface, and each tap weight is changed in the direction opposite to its corresponding gradient component. The change in the jth tap weight is proportional to the size of the jth gradient component. Thus, succeeding values of the coefficient vector $\mathbf{C}$ are obtained according to the relation

$$\mathbf{C}_{k+1} = \mathbf{C}_k - \Delta\mathbf{G}_k, \qquad k = 0, 1, 2, \ldots \tag{11-1-7}$$

where the gradient vector $\mathbf{G}_k$ is

$$\mathbf{G}_k = \frac{1}{2}\frac{dJ}{d\mathbf{C}_k} = \mathbf{\Gamma C}_k - \boldsymbol{\xi} = -E(\varepsilon_k \mathbf{V}_k^*) \tag{11-1-8}$$

The vector $\mathbf{C}_k$ represents the set of coefficients at the kth iteration, $\varepsilon_k = I_k - \hat{I}_k$ is the error signal at the kth iteration, $\mathbf{V}_k$ is the vector of received signal samples that make up the estimate $\hat{I}_k$, i.e., $\mathbf{V}_k = [v_{k+K} \cdots v_k \cdots v_{k-K}]'$, and Δ is a positive number chosen small enough to ensure convergence of the iterative procedure. If the minimum MSE is reached for some $k = k_0$, then $\mathbf{G}_k = \mathbf{0}$, so that no further change occurs in the tap weights. In general, $J_{\min}(K)$ cannot be attained for a finite value of k_0 with the steepest-descent method. It can, however, be approached as closely as desired for some finite value of k_0.

The basic difficulty with the method of steepest descent for determining the optimum tap weights is the lack of knowledge of the gradient vector $\mathbf{G}_k$, which depends on both the covariance matrix $\mathbf{\Gamma}$ and the vector $\boldsymbol{\xi}$ of cross-correlations. In turn, these quantities depend on the coefficients $\{f_k\}$ of the equivalent discrete-time channel model and on the covariance of the information sequence and the additive noise, all of which may be unknown at the receiver

in general. To overcome the difficulty, estimates of the gradient vector may be used. That is, the algorithm for adjusting the tap weight coefficients may be expressed in the form

$$\hat{\mathbf{C}}_{k+1} = \hat{\mathbf{C}}_k - \Delta \hat{\mathbf{G}}_k \tag{11-1-9}$$

where $\hat{\mathbf{G}}_k$ denotes an estimate of the gradient vector $\mathbf{G}_k$ and $\hat{\mathbf{C}}_k$ denotes the estimate of the vector of coefficients.

From (11-1-8) we note that $\mathbf{G}_k$ is the negative of the expected value of the $\varepsilon_k \mathbf{V}_k^*$. Consequently, an estimate of $\mathbf{G}_k$ is

$$\hat{\mathbf{G}}_k = -\varepsilon_k \mathbf{V}_k^* \tag{11-1-10}$$

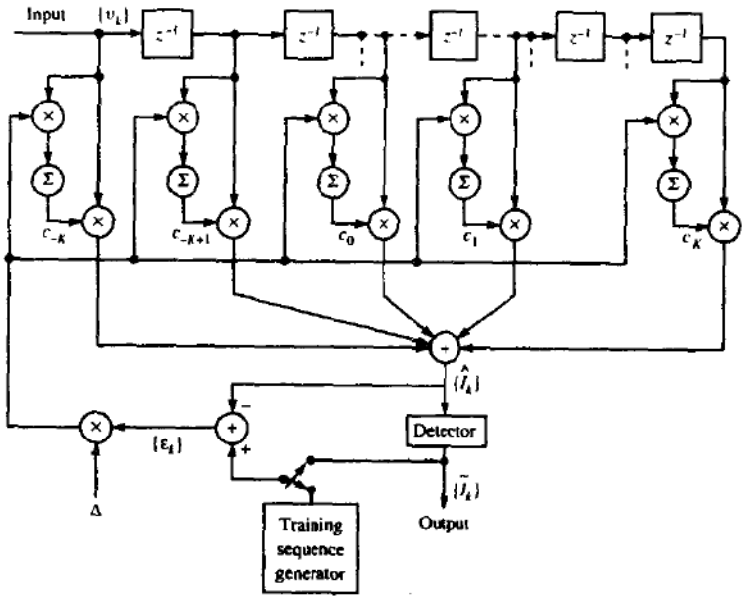
Since $E(\hat{\mathbf{G}}_k) = \mathbf{G}_k$, the estimate $\hat{\mathbf{G}}_k$ is an unbiased estimate of the true gradient vector $\mathbf{G}_k$. Incorporation of (11-1-10) into (11-1-9) yields the algorithm

$$\hat{\mathbf{C}}_{k+1} = \hat{\mathbf{C}}_k + \Delta \varepsilon_k \mathbf{V}_k^* \tag{11-1-11}$$

This is the basic LMS (least-mean-square) algorithm for recursively adjusting the tap weight coefficients of the equalizer first proposed by Widrow and Hoff (1960). It is illustrated in the equalizer shown in Fig. 11-1-2.

The basic algorithm given by (11-1-11) and some of its possible variations have been incorporated into many commercial adaptive equalizers that are

FIGURE 11-1-2 Linear adaptive equalizer based on MSE criterion.



used in high-speed modems. Three variations of the basic algorithm are obtained by using only sign information contained in the error signal ε_k and/or in the components of $\mathbf{V}_k$. Hence, the three possible variations are

$$c_{(k+1)j} = c_{kj} + \Delta \operatorname{csgn}(\varepsilon_k) v^*_{k-j}, \quad j = -K, \ldots, -1, 0, 1, \ldots, K \tag{11-1-12}$$

$$c_{(k+1)j} = c_{kj} + \Delta \varepsilon_k \operatorname{csgn}(v^*_{k-j}), \quad j = -K, \ldots, -1, 0, 1, \ldots, K \tag{11-1-13}$$

$$c_{(k+1)j} = c_{kj} + \Delta \operatorname{csgn}(\varepsilon_k) \operatorname{csgn}(v^*_{k-j}), \quad j = -K, \ldots, -1, 0, 1, \ldots, K \tag{11-1-14}$$

where csgn (x) is defined as

$$\operatorname{csgn}(x) = \begin{cases} 1+j & (\operatorname{Re}(x) > 0, \operatorname{Im}(x) > 0) \\ 1-j & (\operatorname{Re}(x) > 0, \operatorname{Im}(x) < 0) \\ -1+j & (\operatorname{Re}(x) < 0, \operatorname{Im}(x) > 0) \\ -1-j & (\operatorname{Re}(x) < 0, \operatorname{Im}(x) < 0) \end{cases} \tag{11-1-15}$$

(Note that in (11-1-15), $j \equiv \sqrt{-1}$, as distinct from the index j in (11-1-12)–(11-1-14).) Clearly, the algorithm in (11-1-14) is the most easily implemented, but it gives the slowest rate of convergence to the others.

Several other variations of the LMS algorithm are obtained by averaging or filtering the gradient vectors over several iterations prior to making adjustments of the equalizer coefficients. For example, the average over N gradient vectors is

$$\bar{\mathbf{G}}_{mN} = -\frac{1}{N} \sum_{n=0}^{N-1} \varepsilon_{mN+n} \mathbf{V}^*_{mN+n} \tag{11-1-16}$$

and the corresponding recursive equation for updating the equalizer coefficients once very N iterations is

$$\hat{\mathbf{C}}_{(k+1)N} = \hat{\mathbf{C}}_{kN} - \Delta \bar{\mathbf{G}}_{kN} \tag{11-1-17}$$

In effect, the averaging operation performed in (11-1-16) reduces the noise in the estimate of the gradient vector, as shown by Gardner (1984).

An alternative approach is to filter the noisy gradient vectors by a lowpass filter and use the output of the filter as an estimate of the gradient vector. For example, a simple lowpass filter for the noisy gradients yields as an output

$$\bar{\mathbf{G}}_k = w \bar{\mathbf{G}}_{k-1} + (1-w) \hat{\mathbf{G}}_k, \quad \bar{\mathbf{G}}(0) = \hat{\mathbf{G}}(0) \tag{11-1-18}$$

where the choice of $0 \leq w < 1$ determines the bandwidth of the lowpass filter. When w is close to unity, the filter bandwidth is small and the effective averaging is performed over many gradient vectors. On the other hand, when w is small, the lowpass filter has a large bandwidth and, hence, it provides little averaging of the gradient vectors. With the filtered gradient vectors given by

(11-1-18) in place of $\mathbf{G}_k$, we obtain the filtered gradient LMS algorithm given by

$$\hat{\mathbf{C}}_{k+1} = \hat{\mathbf{C}}_k - \Delta\bar{\mathbf{G}}_k \tag{11-1-19}$$

In the above discussion, it has been assumed that the receiver has knowledge of the transmitted information sequence in forming the error signal between the desired symbol and its estimate. Such knowledge can be made available during a short training period in which a signal with a known information sequence is transmitted to the receiver for initially adjusting the tap weights. The length of this sequence must be at least as long as the length of the equalizer so that the spectrum of the transmitted signal adequately covers the bandwidth of the channel being equalized.

In practice, the training sequence is often selected to be a periodic pseudo-random sequence, such as a maximum length shift-register sequence whose period N is equal to the length of the equalizer ($N = 2K + 1$). In this case, the gradient is usually averaged over the length of the sequence as indicated in (11-1-16) and the equalizer is adjusted once a period according to (11-1-17). A practical scheme for continuous adjustment of the tap weights may be either a decision-directed mode of operation in which decisions on the information symbols are assumed to be correct and used in place of I_k in forming the error signal ε_k, or one in which a known pseudo-random-probe sequence is inserted in the information-bearing signal either additively or by interleaving in time and the tap weights adjusted by comparing the received probe symbols with the known transmitted probe symbols. In the decision-directed mode of operation, the error signal becomes $\tilde{\varepsilon}_k = \tilde{I}_k - \hat{I}_k$, where $\tilde{I}_k$ is the decision of the receiver based on the estimate $\hat{I}_k$. As long as the receiver is operating at low error rates, an occasional error will have a negligible effect on the convergence of the algorithm.

If the channel response changes, this change is reflected in the coefficients $\{f_k\}$ of the equivalent discrete-time channel model. It is also reflected in the error signal ε_k, since it depends on $\{f_k\}$. Hence, the tap weights will be changed according to (11-1-11) to reflect the change in the channel. A similar change in the tap weights occurs if the statistics of the noise or the information sequence change. Thus, the equalizer is adaptive.

11-1-3 Convergence Properties of the LMS Algorithm

The convergence properties of the LMS algorithm given by (11-1-11) are governed by the step-size parameter Δ. We shall now consider the choice of the parameter Δ to ensure convergence of the steepest-descent algorithm in (11-1-7), which employs the exact value of the gradient.

From (11-1-7) and (11-1-8), we have

$$\begin{aligned}\mathbf{C}_{k+1} &= \mathbf{C}_k - \Delta\mathbf{G}_k \\ &= (\mathbf{I} - \Delta\boldsymbol{\Gamma})\mathbf{C}_k + \Delta\boldsymbol{\xi}\end{aligned} \tag{11-1-20}$$