ECE 5520: Digital Communications
Lecture Notes
Fall 2009

Dr. Neal Patwari
University of Utah
Department of Electrical and Computer Engineering

Contents

Lecture 1

Today: (1) Syllabus (2) Intro to Digital Communications

1 Class Organization

Textbook Few textbooks cover solely digital communications (without analog) in an introductory communications course. But graduates today will almost always encounter / be developing solely *digital* communication systems. So half of most textbooks are useless; and that the other half is sparse and needs supplemental material. For example, the past text was the 'standard' text in the area for an undergraduate course, Proakis & Salehi, J.G. Proakis and M. Salehi, *Communication Systems Engineering*, 2nd edition, Prentice Hall, 2001. Students didn't like that I had so many supplemental readings. This year's text covers primarily digital communications, and does it in depth. Finally, I find it to be very well-written. And, there are few options in this area. I will provide additional readings solely to provide another presentation style or fit another learning style. Unless specified, these are optional.

Lecture Notes I type my lecture notes. I have taught ECE 5520 previously and have my lecture notes from past years. I'm constantly updating my notes, even up to the lecture time. These can be available to you at lecture, and/or after lecture online. However, you must accept these conditions:

1. **Taking notes is important**: I find most learning requires some writing on your part, not just watching. Please take your own notes.
2. **My written notes do not and cannot reflect everything said during lecture**: I answer questions and understand your perspective better after hearing your questions, and I try to tailor my approach during the lecture. If I didn't, you could just watch a recording!

2 Introduction

A digital communication system conveys discrete-time, discrete-valued information across a physical channel. Information sources might include audio, video, text, or data. They might be continuous-time (analog) signals (audio, images) and even 1-D or 2-D. Or, they may already be digital (discrete-time, discrete-valued). Our object is to convey the signals or data to another place (or time) with as faithful representation as possible.

In this section we talk about what we'll cover in this class, and more importantly, what we won't cover.

2.1 "Executive Summary"

Here is the one sentence version: We will study how to efficiently encode digital data on a noisy, bandwidth-limited analog medium, so that decoding the data (*i.e.*, reception) at a receiver is simple, efficient, and high-fidelity.

The keys points stuffed into that one sentence are:

1. Digital information on an analog medium: We can send waveforms, *i.e.*, real-valued, continuous-time functions, on the channel (medium). These waveforms are from a discrete set of possible waveforms. What set of waveforms should we use? Why?

2. Decoding the data: When receiving a signal (a function) in noise, none of the original waveforms will match exactly. How do you make a decision about which waveform was sent?

3. What makes a receiver difficult to realize? What choices of waveforms make a receiver simpler to implement? What techniques are used in a receiver to compensate?

4. Efficiency, Bandwidth, and Fidelity: Fidelity is the correctness of the received data (*i.e.*, the opposite of error rate). What is the tradeoff between energy, bandwidth, and fidelity? We all want high fidelity, and low energy consumption and bandwidth usage (the costs of our communication system).

You can look at this like an impedance matching problem from circuits. You want, for power efficiency, to have the source impedance match the destination impedance. In digital comm, this means that we want our waveform choices to match the channel and receiver to maximize the efficiency of the communication system.

2.2 Why not Analog?

The previous text used for this course, by Proakis & Salehi, has an extensive analysis and study of analog communication systems, such as radio and television broadcasting (Chapter 3). In the recent past, this course would study both analog and digital communication systems. Analog systems still exist and will continue to exist; however, development of new systems will almost certainly be of digital communication systems. Why?

- Fidelity
- Energy: transmit power, and device power consumption
- Bandwidth efficiency: due to coding gains
- Moore's Law is decreasing device costs for digital hardware
- Increasing need for digital information
- More powerful information security

2.3 Networking Stack

In this course, we study digital communications from bits to bits. That is, we study how to take ones and zeros from a transmitter, send them through a medium, and then (hopefully) correctly identify the same ones and zeros at the receiver. There's a lot more than this to the digital communication systems which you use on a daily basis (*e.g.*, iPhone, WiFi, Bluetooth, wireless keyboard, wireless car key).

To manage complexity, we (engineers) don't try to build a system to do everything all at once. We typically start with an application, and we build a layered network to handle the application. The 7-layer OSI stack, which you would study in a CS computer networking class, is as follows:

- Application
- Presentation (*)
- Session (*)
- Transport
- Network

- Link Layer
- Physical (PHY) Layer

(Note that there is also a 5-layer model in which * layers are considered as part of the application layer.) ECE 5520 is part of the bottom layer, the physical layer. In fact, the physical layer has much more detail. It is primarily divided into:

- Multiple Access Control (MAC)
- Encoding
- Channel / Medium

We can control the MAC and the encoding chosen for a digital communication.

2.4 Channels and Media

We can chose from a few media, but we largely can't change the properties of the medium (although there are exceptions). Here are some media:

- EM Spectra: (anything above 0 Hz) Radio, Microwave, mm-wave bands, light
- Acoustic: ultrasound
- Transmission lines, waveguides, optical fiber, coaxial cable, wire pairs, ...
- Disk (data storage applications)

2.5 Encoding / Decoding Block Diagram

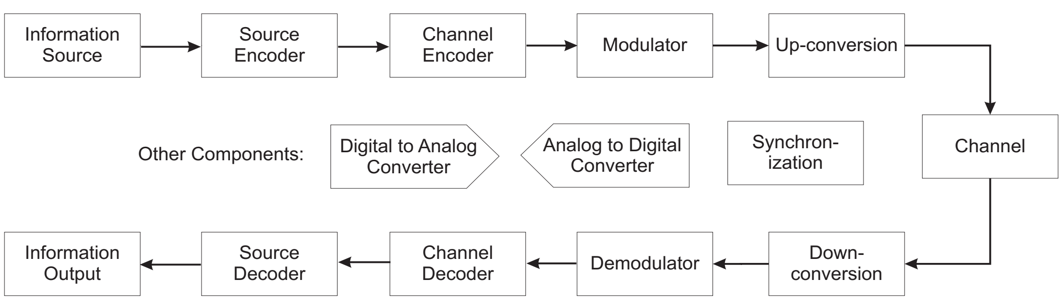


Figure 1: Block diagram of a single-user digital communication system, including (top) transmitter, (middle) channel, and (bottom) receiver.

Notes:

- Information source comes from higher networking layers. It may be continuous or packetized.
- Source Encoding: Finding a compact digital representation for the data source. Includes sampling of continuous-time signals, and quantization of continuous-valued signals. Also includes compression of those sources (lossy, or lossless). What are some compression methods that you're familiar with? We present an introduction to source encoding at the end of this course.

- Channel encoding refers to redundancy added to the signal such that any bit errors can be corrected. A channel decoder, because of the redundancy, can correct some bit errors. We will not study channel encoding, but it is a topic in the (ECE 6520) Coding Theory.
- Modulation refers to the digital-to-analog conversion which produces a continuous-time signal that can be sent on the physical channel. It is analogous to impedance matching - proper matching of a modulation to a channel allows optimal information transfer, like impedance matching ensured optimal power transfer. Modulation and demodulation will be the main focus of this course.
- Channels: See above for examples. Typical models are additive noise, or linear filtering channel.

Why do we do both source encoding (which compresses the signal as much as possible) and also channel encoding (which adds redundancy to the signal)? Because of Shannon's source-channel coding separation theorem. He showed that (given enough time) we can consider them separately without additional loss. And separation, like layering, reduces complexity to the designer.

2.6 Channels

A channel can typically be modeled as a linear filter with the addition of noise. The noise comes from a variety of sources, but predominantly:

1. Thermal background noise: Due to the physics of living above 0 Kelvin. Well modeled as Gaussian, and white; thus it is referred to as additive white Gaussian noise (AWGN).
2. Interference from other transmitted signals. These other transmitters whose signals we cannot completely cancel, we lump into the 'interference' category. These may result in non-Gaussian noise distribution, or non-white noise spectral density.

The linear filtering of the channel result from the physics and EM of the medium. For example, attenuation in telephone wires varies by frequency. Narrowband wireless channels experience fading that varies quickly as a function of frequency. Wideband wireless channels display multipath, due to multiple time-delayed reflections, diffractions, and scattering of the signal off of the objects in the environment. All of these can be modeled as linear filters.

The filter may be constant, or time-invariant, if the medium, the TX and RX do not move or change. However, for mobile radio, the channel may change very quickly over time. Even for stationary TX and RX, in real wireless channels, movement of cars, people, trees, etc. in the environment may change the channel slowly over time.

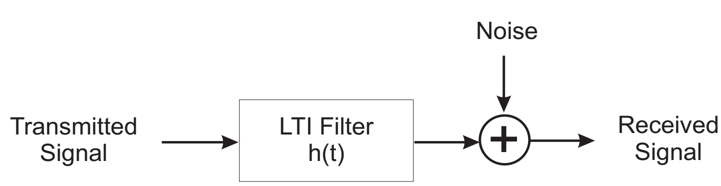


Figure 2: Linear filter and additive noise channel model.

In this course, we will focus primarily on the AWGN channel, but we will mention what variations exist for particular channels, and how they are addressed.

2.7 Topic: Random Processes

Random things in a communication system:

- Noise in the channel
- Signal (bits)
- Channel filtering, attenuation, and fading
- Device frequency, phase, and timing offsets

These random signals often pass through LTI filters, and are sampled. We want to build the best receiver possible despite the impediments. Optimal receiver design is something that we study using probability theory.

We have to tolerate errors. Noise and attenuation of the channel will cause bit errors to be made by the demodulator and even the channel decoder. This may be tolerated, or a higher layer networking protocol (eg., TCP-IP) can determine that an error occurred and then re-request the data.

2.8 Topic: Frequency Domain Representations

To fit as many signals as possible onto a channel, we often split the signals by frequency. The concept of sharing a channel is called *multiple access* (MA). Separating signals by frequency band is called frequency-division multiple access (FDMA). For the wireless channel, this is controlled by the FCC (in the US) and called spectrum allocation. There is a tradeoff between frequency requirements and time requirements, which will be a major part of this course. The Fourier transform of our modulated, transmitted signal is used to show that it meets the spectrum allocation limits of the FCC.

2.9 Topic: Orthogonality and Signal spaces

To show that signals sharing the same channel don't interfere with each other, we need to show that they are *orthogonal*. This means, in short, that a receiver can uniquely separate them. Signals in different frequency bands are orthogonal.

We can also employ multiple orthogonal signals in a single transmitter and receiver, in order to provide multiple independent means (dimensions) on which to modulate information. We will study orthogonal signals, and learn an algorithm to take an arbitrary set of signals and output a set of orthogonal signals with which to represent them. We'll use signal spaces to show graphically the results, as the example in Figure 36.

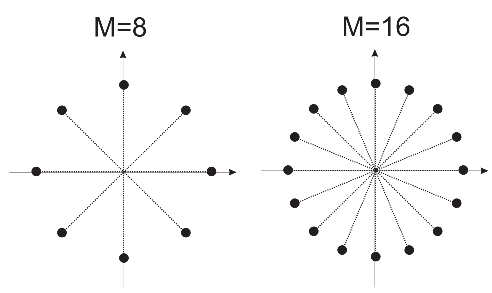


Figure 3: Example signal space diagram for M-ary Phase Shift Keying, for (a) $M = 8$ and (b) $M = 16$. Each point is a vector which can be used to send a 3 or 4 bit sequence.

2.10 Related classes

1. Pre-requisites: (ECE 5510) Random Processes; (ECE 3500) Signals and Systems.
2. Signal Processing: (ECE 5530): Digital Signal Processing
3. Electromagnetics: EM Waves, (ECE 5320-5321) Microwave Engineering, (ECE 5324) Antenna Theory, (ECE 5411) Fiberoptic Systems
4. Breadth: (ECE 5325) Wireless Communications
5. Devices and Circuits: (ECE 3700) Fundamentals of Digital System Design, (ECE 5720) Analog IC Design
6. Networking: (ECE 5780) Embedded System Design, (CS 5480) Computer Networks
7. Advanced Classes: (ECE 6590) Software Radio, (ECE 6520) Information Theory and Coding, (ECE 6540): Estimation Theory

---

Lecture 2

Today: (1) Power, Energy, dB (2) Time-domain concepts (3) Bandwidth, Fourier Transform

Two of the biggest limitations in communications systems are (1) *energy / power*; and (2) *bandwidth*. Today's lecture provides some tools to deal with power and energy, and starts the review of tools to analyze frequency content and bandwidth.

3 Power and Energy

Recall that energy is power times time. **Use the units**: energy is measured in Joules (J); power is measured in Watts (W) which is the same as Joules/second (J/sec). Also, recall that our standard in signals and systems is define our signals, such as $x(t)$, as voltage signals (V). When we want to know the power of a signal we assume it is being dissipated in a 1 Ohm resistor, so $|x(t)|^2$ is the power dissipated at time t (since power is equal to the voltage squared divided by the resistance).

A signal $x(t)$ has energy defined as

$$E = \int_{-\infty}^{\infty} |x(t)|^2 dt$$

For some signals, E will be infinite because the signal is non-zero for an infinite duration of time (it is always on). These signals we call *power signals* and we compute their power as

$$P = \lim_{T\rightarrow\infty} \frac{1}{2T} \int_{-T}^{T} |x(t)|^2 dt$$

The signal with finite energy is called an *energy signal*.

3.1 Discrete-Time Signals

In this book, we refer to discrete samples of the sampled signal x as $x(n)$. You may be more familiar with the $x[n]$ notation. But, Matlab uses parentheses also; so we'll follow the Rice text notation. Essentially, whenever you see a function of n (or k, l, m), it is a discrete-time function; whenever you see a function of t (or perhaps τ) it is a continuous-time function. I'm sorry this is not more obvious in the notation.

For discrete-time signals, energy and power are defined as:

$$E = \sum_{n=-\infty}^{\infty} |x(n)|^2 \quad (1)$$

$$P = \lim_{N\to\infty} \frac{1}{2N+1} \sum_{n=-N}^{N} |x(n)|^2 \quad (2)$$

3.2 Decibel Notation

We often use a decibel (dB) scale for power. If P_{lin} is the power in Watts, then

$$[P]_{\text{dBW}} = 10 \log_{10} P_{lin}$$

Decibels are more general - they can apply to other unitless quantities as well, such as a gain (loss) $L(f)$ through a filter $H(f)$,

$$[L(f)]_{\text{dB}} = 10 \log_{10} |H(f)|^2 \quad (3)$$

Note: Why is the capital B used? Either the lowercase 'b' in the SI system is reserved for bits, so when the 'bel' was first proposed as $\log_{10}(\cdot)$, it was capitalized; or it referred to a name 'Bell' so it was capitalized. In either case, we use the unit decibel $10 \log_{10}(\cdot)$ which is then abbreviated as dB in the SI system.

Note that (3) could also be written as:

$$[L(f)]_{\text{dB}} = 20 \log_{10} |H(f)| \quad (4)$$

Be careful with your use of 10 vs. 20 in the dB formula.

- Only use 20 as the multiplier if you are converting from voltage to power; *i.e.*, taking the $\log_{10}$ of a voltage and expecting the result to be a dB power value.

Our standard is to consider power gains and losses, not voltage gains and losses. So if we say, for example, the channel has a loss of 20 dB, this refers to a loss in power. In particular, the output of the channel has 100 times less power than the input to the channel.

Remember these two dB numbers:

- 3 dB: This means the number is double in linear terms.
- 10 dB: This means the number is ten times in linear terms.

And maybe this one:

- 1 dB: This means the number is a little over 25% more (multiply by 5/4) in linear terms.

With these three numbers, you can quickly convert losses or gains between linear and dB units without a calculator. Just convert any dB number into a sum of multiples of 10, 3, and 1.

Example: Convert dB to linear values:

1. 30 dBW
2. 33 dBm
3. -20 dB
4. 4 dB

Example: Convert linear values to dB:

1. 0.2 W
2. 40 mW

Example: Convert power relationships to dB:
Convert the expression to one which involves only dB terms.

1. $P_{y,lin} = 100 P_{x,lin}$
2. $P_{o,lin} = G_{connector,lin} L_{cable,lin}^{-d}$, where $P_{o,lin}$ is the received power in a fiber-optic link, where d is the cable length (typically in units of km), $G_{connector,lin}$ is the gain in any connectors, and $L_{cable,lin}$ is a loss in a 1 km cable.
3. $P_{r,lin} = P_{t,lin} \frac{G_{t,lin} G_{t,lin} \lambda^2}{(4\pi d)^2}$, where λ is the wavelength (m), d is the path length (m), and $G_{t,lin}$ and $G_{t,lin}$ are the linear gains in the antennas, $P_{t,lin}$ is the transmit power (W) and $P_{r,lin}$ is the received power (W). This is the Friis free space path loss formula.

These last two are what we will need in Section 6.4, when we discuss link budgets. The main idea is that we have a limited amount of power which will be available at the receiver.

4 Time-Domain Concept Review

4.1 Periodicity

Def'n: *Periodic (continuous-time)*
A signal $x(t)$ is periodic if $x(t) = x(t + T_0)$ for some constant $T_0 \neq 0$ for all $t \in \mathbb{R}$. The smallest such constant $T_0 > 0$ is the *period*.

If a signal is not periodic it is *aperiodic*.
Periodic signals have Fourier series representations, as defined in Rice Ch. 2.

Def'n: *Periodic (discrete-time)*
A DT signal $x(n)$ is periodic if $x(n) = x(n + N_0)$ for some integer $N_0 \neq 0$, for all integers n. The smallest positive integer N_0 is the period.

4.2 Impulse Functions

Def'n: *Impulse Function*
The (Dirac) impulse function $\delta(t)$ is the function which makes

$$\int_{-\infty}^{\infty} x(t)\delta(t)dt = x(0) \tag{5}$$

true for any function $x(t)$ which is continuous at $t = 0$.

We are defining a function by its most important property, the 'sifting property'. Is there another definition which is more familiar?

Solution:

$$\delta(t) = \lim_{T \to 0} \begin{cases} 1/T, & -T \le t \le T \\ 0, & o.w. \end{cases}$$

You can visualize $\delta(t)$ here as an infinitely high, infinitesimally wide pulse at the origin, with area one. This is why it 'pulls out' the value of $x(t)$ in the integral in (5).

Other properties of the impulse function:

- Time scaling,
- Symmetry,
- Sifting at arbitrary time t_0,

The continuous-time unit step function is

$$u(t) = \begin{cases} 1, & t \ge 0 \\ 0, & o.w. \end{cases}$$

Example: Sifting Property
What is $\int_{-\infty}^{\infty} \frac{\sin(\pi t)}{\pi t} \delta(1-t)dt$?

The discrete-time impulse function (also called the Kronecker delta or δ_K) is defined as:

$$\delta(n) = \begin{cases} 1, & n = 0 \\ 0, & o.w. \end{cases}$$

(There is no need to get complicated with the math; this is well defined.) Also,

$$u(n) = \begin{cases} 1, & n \ge 0 \\ 0, & o.w. \end{cases}$$

5 Bandwidth

Bandwidth is another critical resource for a digital communications system; we have various definitions to quantify it. In short, it isn't easy to describe a signal in the frequency domain with a single number. And, in the end, a system will be designed to meet a spectral mask required by the FCC or system standard.

	Periodicity	
Time	*Periodic*	*Aperiodic*
Continuous-Time	Fourier Series $x(t) \leftrightarrow a_k$	Laplace Transform $x(t) \leftrightarrow X(s)$ Fourier Transform $x(t) \leftrightarrow X(j\omega)$ $X(j\omega) = \int_{t=-\infty}^{\infty} x(t)e^{-j\omega t}dt$ $x(t) = \frac{1}{2\pi}\int_{\omega=-\infty}^{\infty} X(j\omega)e^{j\omega t}d\omega$
Discrete-Time	Discrete Fourier Transform (DFT) $x(n) \leftrightarrow X[k]$ $X[k] = \sum_{n=0}^{N-1} x(n)e^{-j\frac{2\pi}{N}kn}$ $x(n) = \frac{1}{N}\sum_{k=0}^{N-1} X[k]e^{j\frac{2\pi}{N}nk}$	z-Transform $x(n) \leftrightarrow X(z)$ Discrete Time Fourier Transform (DTFT) $x(n) \leftrightarrow X(e^{j\Omega})$ $X(e^{j\Omega}) = \sum_{n=-\infty}^{\infty} x(n)e^{-j\Omega n}$ $x(n) = \frac{1}{2\pi}\int_{n=-\pi}^{\pi} X(e^{j\Omega})d\Omega$

Table 1: Frequency Transforms

Intuitively, bandwidth is the maximum extent of our signal's frequency domain characterization, call it $X(f)$. A baseband signal absolute bandwidth is often defined as the W such that $X(f) = 0$ for all f except for the range $-W \leq f \leq W$. Other definitions for bandwidth are

- 3-dB bandwidth: B_{3dB} is the value of f such that $|X(f)|^2 = |X(0)|^2/2$.
- 90% bandwidth: $B_{90\%}$ is the value which captures 90% of the energy in the signal:

$$\int_{-B_{90\%}}^{B_{90\%}} |X(f)|^2 df = 0.90 \int_{|=-\infty}^{\infty} Xd|(f)|^2 df$$

As a motivating example, I mention the square-root raised cosine (SRRC) pulse, which has the following desirable Fourier transform:

$$H_{RRC}(f) = \begin{cases} \sqrt{T_s}, & 0 \leq |f| \leq \frac{1-\alpha}{2T_s} \\ \sqrt{\frac{T_s}{2}\left\{1 + \cos\left[\frac{\pi T_s}{\alpha}\left(|f| - \frac{1-\alpha}{2T_s}\right)\right]\right\}}, & \frac{1-\alpha}{2T_s} \leq |f| \leq \frac{1+\alpha}{2T_s} \\ 0, & o.w. \end{cases} \quad (6)$$

where α is a parameter called the "rolloff factor". We can actually analyze this using the properties of the Fourier transform and many of the standard transforms you'll find in a Fourier transform table.

The SRRC and other pulse shapes are discussed in Appendix A, and we will go into more detail later on. The purpose so far is to motivate practicing up on frequency transforms.

5.1 Continuous-time Frequency Transforms

Notes about continuous-time frequency transforms:

1. You are probably most familiar with the Laplace Transform. To convert it to the Fourier tranform, we replace s with $j\omega$, where ω is the radial frequency, with units radians per second (rad/s).

2. You may prefer the radial frequency representation, but also feel free to use the rotational frequency f (which has units of cycles per sec, or Hz. Frequency in Hz is more standard for communications; you should use it for intuition. In this case, just substitute $\omega = 2\pi f$. You could write $X(j2\pi f)$ as the notation for this, but typically you'll see it abbreviated as $X(f)$. Note that the definition of the Fourier tranform in the f domain loses the $\frac{1}{2\pi}$ in the inverse Fourier transform definition.s

$$\begin{aligned} X(j2\pi f) &= \int_{t=-\infty}^{\infty} x(t) e^{-j2\pi f t} dt \\ x(t) &= \int_{f=-\infty}^{\infty} X(j2\pi f) e^{j2\pi f t} df \end{aligned} \tag{7}$$

3. The Fourier series is limited to purely periodic signals. Both Laplace and Fourier transforms are *not* limited to periodic signals.

4. Note that $e^{j\alpha} = \cos(\alpha) + j\sin(\alpha)$.

See Table 2.4.4 in the Rice book.

Example: Square Wave
Given a rectangular pulse $x(t) = \text{rect}(t/T_s)$,

$$x(t) = \begin{cases} 1, & -T_s/2 < t \le T_s/2 \\ 0, & o.w. \end{cases}$$

What is the Fourier transform $X(f)$? Calculate both from the definition and from a table.

Solution: Method 1: From the definition:

$$\begin{aligned} X(j\omega) &= \int_{t=-T_s/2}^{T_s/2} e^{-j\omega t} dt \\ &= \frac{1}{-j\omega} e^{-j\omega t} \Big|_{t=-T_s/2}^{T_s/2} \\ &= \frac{1}{-j\omega} \left(e^{-j\omega T_s/2} - e^{j\omega T_s/2} \right) \\ &= 2\frac{\sin(\omega T_s/2)}{\omega} = T_s \frac{\sin(\omega T_s/2)}{\omega T_s/2} \end{aligned}$$

This uses the fact that $\frac{1}{-2j}\left(e^{-j\alpha} - e^{j\alpha}\right) = \sin(\alpha)$. While it is sometimes convenient to replace $\sin(\pi x)/(\pi x)$ with $\text{sinc} x$, it is confusing because $\text{sinc}(x)$ is sometimes defined as $\sin(\pi x)/(\pi x)$ and sometimes defined as $(\sin x)/x$. No standard definition for 'sinc' exists! Rather than make a mistake because of this, the Rice book always writes out the expression fully. I will try to follow suit.
Method 2: From the tables and properties:

$$x(t) = g(2t) \text{ where } g(t) = \begin{cases} 1, & |t| < T_s \\ 0, & o.w. \end{cases} \tag{8}$$

From the table, $G(j\omega) = 2T_s \frac{\sin(\omega T_s)}{\omega T_s}$. From the properties, $X(j\omega) = \frac{1}{2} G\left(j\frac{\omega}{2}\right)$. So

$$X(j\omega) = T_s \frac{\sin(\omega T_s/2)}{\omega T_s/2}$$

See Figure 4(a).

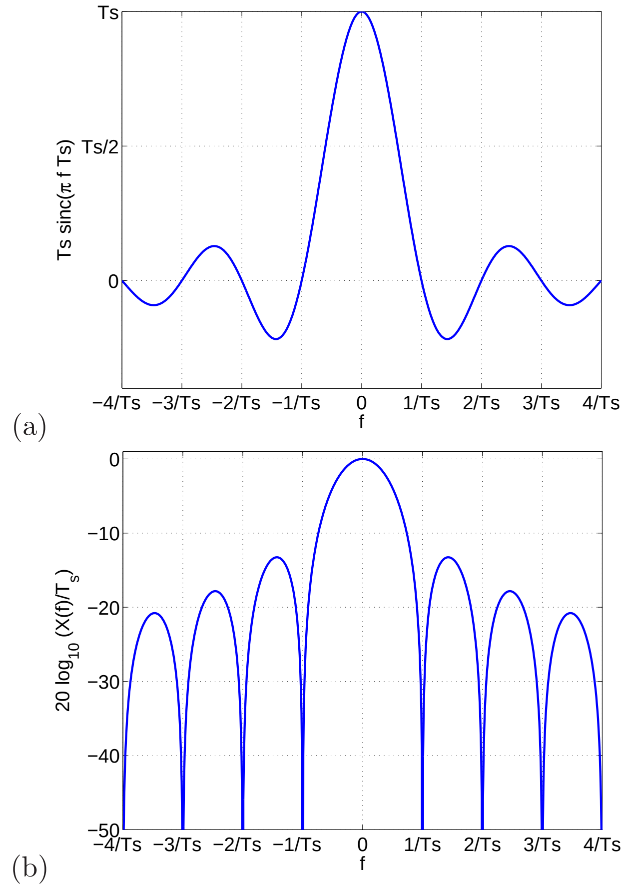


Figure 4: (a) Fourier transform $X(j2\pi f)$ of rect pulse with period T_s, and (b) Power vs. frequency $20\log_{10}(X(j2\pi f)/T_s)$.

Question: What if $Y(j\omega)$ was a rect function? What would the inverse Fourier transform $y(t)$ be?

5.1.1 Fourier Transform Properties

See Table 2.4.3 in the Rice book.

Assume that $\mathfrak{F}\{x(t)\} = X(j\omega)$. Important properties of the Fourier transform:

1. Duality property:

$$
\begin{aligned}
x(j\omega) &= \mathfrak{F}\{X(-t)\} \\
x(-j\omega) &= \mathfrak{F}\{X(t)\}
\end{aligned}
$$

   (Confusing. It says is that you can go backwards on the Fourier transform, just remember to flip the result around the origin.)

2. Time shift property:

$$\mathfrak{F}\{x(t-t_0)\} = e^{-j\omega t_0}X(j\omega)$$

3. Scaling property: for any real $a \neq 0$,

$$\mathfrak{F}\left\{x(at)\right\} = \frac{1}{|a|} X\left(j\frac{\omega}{a}\right)$$

4. Convolution property: if, additionally $y(t)$ has Fourier transform $X(j\omega)$,

$$\mathfrak{F}\left\{x(t) \star y(t)\right\} = X(j\omega) \cdot Y(j\omega)$$

5. Modulation property:

$$\mathfrak{F}\left\{x(t)cos(\omega_0 t)\right\} = \frac{1}{2}X(\omega - \omega_0) + \frac{1}{2}X(\omega + \omega_0)$$

6. Parceval's theorem: The energy calculated in the frequency domain is equal to the energy calculated in the time domain.

$$\int_{t=-\infty}^{\infty} |x(t)|^2 dt = \int_{f=-\infty}^{\infty} |X(f)|^2 df = \frac{1}{2\pi} \int_{\omega=-\infty}^{\infty} |X(j\omega)|^2 d\omega$$

   So do whichever one is easiest! Or, check your answer by doing both.

5.2 Linear Time Invariant (LTI) Filters

If a (deterministic) signal $x(t)$ is input to a LTI filter with impulse response $h(t)$, the output signal is

$$y(t) = h(t) \star x(t)$$

Using the above convolution property,

$$Y(j\omega) = X(j\omega) \cdot H(j\omega)$$

5.3 Examples

Example: Applying FT Properties
If $w(t)$ has the Fourier transform

$$W(j\omega) = \frac{j\omega}{1 + j\omega}$$

find $X(j\omega)$ for the following waveforms:

1. $x(t) = w(2t + 2)$
2. $x(t) = e^{-jt}w(t - 1)$
3. $x(t) = 2\frac{\partial w(t)}{\partial t}$
4. $x(t) = w(1 - t)$

Solution: To be worked out in class.

Lecture 3

Today: (1) Bandpass Signals (2) Sampling

Solution: From the Examples at the end of Lecture 2:

1. Let $x(t) = z(2t)$ where $z(t) = w(t+2)$. Then
$$Z(j\omega) = e^{j2\omega}\frac{j\omega}{1+j\omega}, \text{ and } X(j\omega) = \frac{1}{2}Z(j\omega/2).$$
So $X(j\omega) = \frac{1}{2}e^{j\omega}\frac{j\omega/2}{1+j\omega/2}$. Alternatively, let $x(t) = r(t+1)$ where $r(t) = w(2t)$. Then
$$R(j\omega) = \frac{1}{2}W(j\omega/2), \text{ and } X(j\omega) = e^{j\omega}R(j\omega).$$
Again, $X(j\omega) = \frac{1}{2}e^{j\omega}\frac{j\omega/2}{1+j\omega/2}$.

2. Let $x(t) = e^{-jt}z(t)$ where $z(t) = w(t-1)$. Then $Z(\jmath\omega) = e^{-j\omega}W(j\omega)$, and $X(j\omega) = Z(j(\omega-1))$. So $X(j\omega) = e^{-j(\omega-1)}W(j(\omega-1)) = e^{-j(\omega+1)}\frac{j(\omega+1)}{1+j(\omega+1)}$.

3. $X(j\omega) = 2j\omega\frac{j\omega}{1+j\omega} = -\frac{2\omega^2}{1+j\omega}$.

4. Let $x(t) = y(-t)$, where $y(t) = w(1+t)$. Then
$$X(j\omega) = Y(-j\omega), \text{ where } Y(j\omega) = e^{j\omega}W(j\omega) = e^{j\omega}\frac{j\omega}{1+j\omega}$$
So $X(j\omega) = e^{-j\omega}\frac{-j\omega}{1-j\omega}$.

6 Bandpass Signals

<u>Def'n:</u> *Bandpass Signal*
A bandpass signal $x(t)$ has Fourier transform $X(j\omega) = 0$ for all ω such that $|\omega \pm \omega_c| > W/2$, where $W/2 < \omega_c$. Equivalently, a bandpass signal $x(t)$ has Fourier transform $X(f) = 0$ for all f such that $|f \pm f_c| > B/2$, where $B/2 < f_c$.

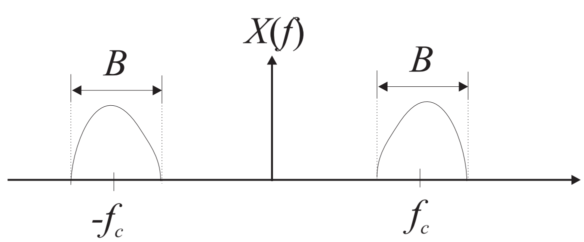


Figure 5: Bandpass signal with center frequency f_c.

Realistically, $X(j\omega)$ will not be exactly zero outside of the bandwidth, there will be some 'sidelobes' out of the main band.

6.1 Upconversion

We can take a baseband signal $x_C(t)$ and 'upconvert' it to be a bandpass filter by multiplying with a cosine at the desired center frequency, as shown in Fig. 6.

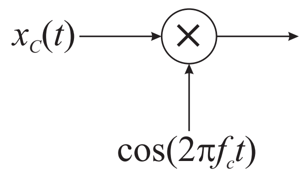


Figure 6: Modulator block diagram.

Example: Modulated Square Wave
We dealt last time with the square wave $x(t) = \text{rect}(t/T_s)$. This time, we will look at:

$$x(t) = \text{rect}(t/T_s)\cos(\omega_c t)$$

where ω_c is the center frequency in radians/sec ($f_c = \omega_c/(2\pi)$ is the center frequency in Hz). Note that I use "rect" as follows:

$$\text{rect}(t) = \begin{cases} 1, & -1/2 < t \leq 1/2 \\ 0, & o.w. \end{cases}$$

What is $X(j\omega)$?

Solution:

$$\begin{aligned} X(j\omega) &= \mathfrak{F}\{\text{rect}(t/T_s)\} \star \mathfrak{F}\{\cos(\omega_c t)\} \\ X(j\omega) &= \frac{1}{2\pi} T_s \frac{\sin(\omega T_s/2)}{\omega T_s/2} \star \pi\left[\delta(\omega-\omega_c) + \delta(\omega+\omega_c)\right] \\ X(j\omega) &= \frac{T_s}{2}\frac{\sin[(\omega-\omega_c)T_s/2]}{(\omega-\omega_c)T_s/2} + \frac{T_s}{2}\frac{\sin[(\omega+\omega_c)T_s/2]}{(\omega+\omega_c)T_s/2} \end{aligned}$$

The plot of $X(j2\pi f)$ is shown in Fig. 7.

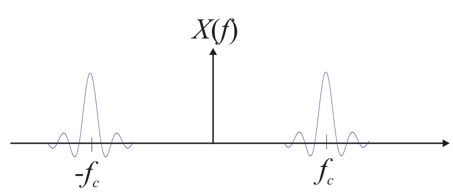


Figure 7: Plot of $X(j2\pi f)$ for modulated square wave example.

6.2 Downconversion of Bandpass Signals

How will we get back the desired baseband signal $x_C(t)$ at the receiver?

1. Multiply (again) by the carrier $2\cos(\omega_c t)$.
2. Input to a low-pass filter (LPF) with cutoff frequency ω_c.

This is shown in Fig. 8.

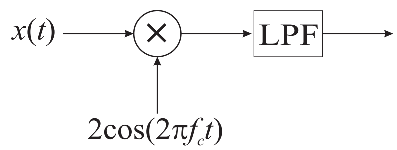


Figure 8: Demodulator block diagram.

Assume the general modulated signal was,

$$\begin{aligned} x(t) &= x_C(t)\cos(\omega_c t) \\ X(j\omega) &= \frac{1}{2}X_C(j(\omega-\omega_c)) + \frac{1}{2}X_C(j(\omega+\omega_c)) \end{aligned}$$

What happens after the multiplication with the carrier in the receiver?

Solution:

$$\begin{aligned} X_1(j\omega) &= \frac{1}{2\pi}X(j\omega) \star 2\pi\left[\delta(\omega-\omega_c)+\delta(\omega+\omega_c)\right] \\ X_1(j\omega) &= \frac{1}{2}X_C(j(\omega-2\omega_c)) + \frac{1}{2}X_C(j\omega) + \\ &\quad \frac{1}{2}X_C(j\omega) + \frac{1}{2}X_C(j(\omega+2\omega_c)) \end{aligned}$$

There are components at $2\omega_c$ and $-2\omega_c$ which will be canceled by the LPF. (Does the LPF need to be ideal?)

$$X_2(j\omega) = X_C(j\omega)$$

So $x_2(t) = x_C(t)$.

7 Sampling

A common statement of the Nyquist sampling theorem is that a signal can be sampled at twice its bandwidth. But the theorem really has to do with signal reconstruction from a sampled signal.

Theorem: (Nyquist Sampling Theorem.) Let $x_c(t)$ be a baseband, continuous signal with bandwidth B (in Hz), *i.e.*, $X_c(j\omega) = 0$ for all $|\omega| \geq 2\pi B$. Let $x_c(t)$ be sampled at multiples of T, where $\frac{1}{T} \geq 2B$ to yield the sequence $\{x_c(nT)\}_{n=-\infty}^{\infty}$. Then

$$x_c(t) = 2BT\sum_{n=-\infty}^{\infty} x_c(nT)\frac{\sin(2\pi B(t-nT))}{2\pi B(t-nT)}. \tag{9}$$

Proof: Not covered.

Notes:

- This is an interpolation procedure.
- Given $\{x_n\}_{n=-\infty}^{\infty}$, how would you find the maximum of $x(t)$?
- This is only precise when $X(j\omega) = 0$ for all $|\omega| \geq 2\pi B$.

7.1 Aliasing Due To Sampling

Essentially, sampling is the multiplication of a impulse train (at period T with the desired signal $x(t)$:

$$\begin{aligned} x_{sa}(t) &= x(t) \sum_{n=-\infty}^{\infty} \delta(t - nT) \\ x_{sa}(t) &= \sum_{n=-\infty}^{\infty} x(nT)\delta(t - nT) \end{aligned}$$

What is the Fourier transform of $x_{sa}(t)$?

Solution: In the frequency domain, this is a convolution:

$$\begin{aligned} X_{sa}(j\omega) &= X(j\omega) \star \frac{2\pi}{T} \sum_{n=-\infty}^{\infty} \delta\left(\omega - \frac{2\pi n}{T}\right) \\ &= \frac{2\pi}{T} \sum_{n=-\infty}^{\infty} X\left(j\left(\omega - \frac{2\pi n}{T}\right)\right) \quad \text{for all } \omega \\ &= \frac{1}{T} X(j\omega) \quad \text{for } |\omega| < 2\pi B \end{aligned} \tag{10}$$

This is shown graphically in the Rice book in Figure 2.12.

The Fourier transform of the sampled signal is many copies of $X(j\omega)$ strung at integer multiples of $2\pi/T$, as shown in Fig. 9.

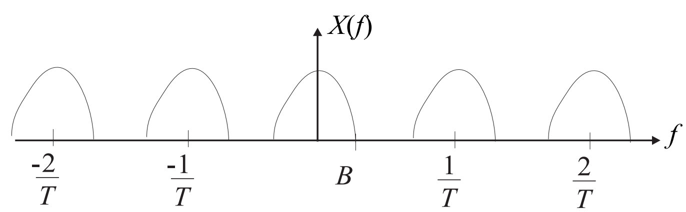


Figure 9: The effect of sampling on the frequency spectrum in terms of frequency f in Hz.

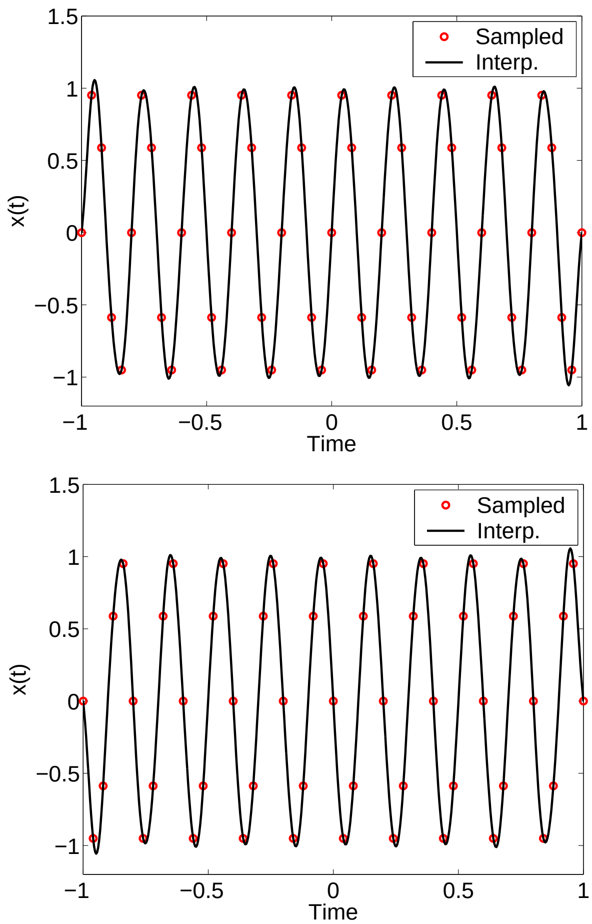
Example: Sinusoid sampled above and below Nyquist rate
Consider two sinusoidal signals sampled at $1/T = 25$ Hz:

$$\begin{aligned} x_1(nT) &= \sin(2\pi 5nT) \\ x_2(nT) &= \sin(2\pi 20nT) \end{aligned}$$

What are the two frequencies of the sinusoids, and what is the Nyquist rate? Which of them does the Nyquist theorem apply to? Draw the spectrums of the continuous signals $x_1(t)$ and $x_2(t)$, and indicate what the spectrum is of the sampled signals.

Figure 10 shows what happens when the Nyquist theorem is applied to the each signal (whether or not it is valid). What observations would you make about Figure 10(b), compared to Figure 10(a)?

Example: Square vs. round pulse shape
Consider the square pulse considered before, $x_1(t) = \text{rect}(t/T_s)$. Also consider a parabola pulse (this doesn't



(a)

(b)

Figure 10: Sampled (a) $x_1(nT)$ and (b) $x_2(nT)$ are interpolated (—) using the Nyquist interpolation formula.

really exist in the wild – I'm making it up for an example.)

$$x_2(t) = \begin{cases} 1 - \left(\frac{2t}{T_s}\right)^2, & -\frac{1}{2T_s} \leq t \leq \frac{1}{2T_s} \\ 0, & o.w. \end{cases}$$

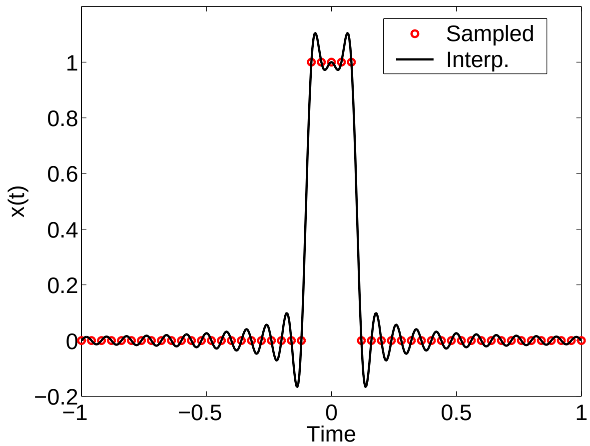
What happens to $x_1(t)$ and $x_2(t)$ when they are sampled at rate T?

In the Matlab code `EgNyquistInterpolation.m` we set $T_s = 1/5$ and $T = 1/25$. Then, the sampled pulses are interpolated using (9). Even though we've sampled at a pretty high rate, the reconstructed signals will not be perfect representations of the original, in particular for $x_1(t)$. See Figure 11.

7.2 Connection to DTFT

Recall (10). We calculated the Fourier transform of the product by doing the convolution in the frequency domain. Instead, we can calculate it directly from the formula.

(a)



(b)

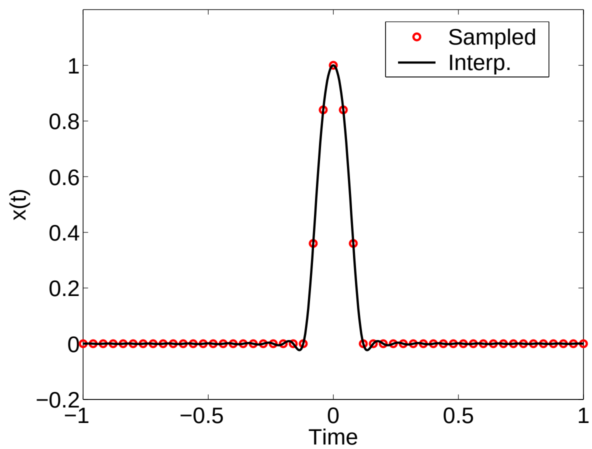


Figure 11: Sampled (a) $x_1(nT)$ and (b) $x_2(nT)$ are interpolated (—) using the Nyquist interpolation formula.

Solution:

$$\begin{aligned} X_{sa}(j\omega) &= \int_{t=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} x(nT)\delta(t-nT)e^{-j\omega t} \\ &= \sum_{n=-\infty}^{\infty} x(nT) \int_{t=-\infty}^{\infty} \delta(t-nT)e^{-j\omega t} \\ &= \sum_{n=-\infty}^{\infty} x(nT)e^{-j\omega nT} \end{aligned} \tag{11}$$

The discrete-time Fourier transform of $x_{sa}(t)$, I denote DTFT $\{x_{sa}(t)\}$, and it is:

$$\text{DTFT}\{x_{sa}(t)\} = X_d(e^{j\Omega}) = \sum_{n=-\infty}^{\infty} x(nT)e^{-j\Omega n}$$

Essentially, the difference between the DTFT and the Fourier transform *of the sampled signal* is the relationship

$$\Omega = \omega T = 2\pi f T$$

But this defines the relationship only between the Fourier transform of the sampled signal. How can we relate this to the FT of the continuous-time signal? A: Using (10). We have that $X_d(e^{j\omega}) = X_{sa}\left(\frac{\Omega}{2\pi T}\right)$. Then plugging into (10),

Solution:

$$\begin{aligned} X_d(e^{j\Omega}) &= X_{sa}\left(j\frac{\Omega}{T}\right) \\ &= \frac{2\pi}{T}\sum_{n=-\infty}^{\infty} X\left(j\left(\frac{\Omega}{T} - \frac{2\pi n}{T}\right)\right) \\ &= \frac{2\pi}{T}\sum_{n=-\infty}^{\infty} X\left(j\left(\frac{\Omega - 2\pi n}{T}\right)\right) \end{aligned} \tag{12}$$

If $x(t)$ is sufficiently bandlimited, $X_d(e^{j\Omega}) \propto X(j\Omega/T)$ in the interval $-\pi < \Omega < \pi$. This relationship holds for the Fourier transform of the original, *continuous-time signal* between $-\pi < \Omega < \pi$ **if and only if** the original signal $x(t)$ is sufficiently **bandlimited** so that the Nyquist sampling theorem applies.

Notes:

- Most things in the DTFT table in Table 2.8 (page 68) are analogous to continuous functions which are not bandlimited. So - don't expect the last form to apply to any continuous function.
- The DTFT is periodic in Ω with period 2π.
- Don't be confused: The DTFT is a continuous function of Ω.

Lecture 4

Today: (1) Example: Bandpass Sampling (2) Orthogonality / Signal Spaces I

7.3 Bandpass sampling

Bandpass sampling is the use of sampling to perform down-conversion via frequency aliasing. We assume that the signal is truly a bandpass signal (has no DC component). Then, we sample with rate $1/T$ and we rely on aliasing to produce an image the spectrum at a relatively low frequency (less than one-half the sampling rate).

Example: Bandpass Sampling
This is Example 2.6.2 (pp. 60-65) in the Rice book. In short, we have a bandpass signal in the frequency band, 450 Hz to 460 Hz. The center frequency is 455 Hz. We want to sample the signal. We have two choices:

1. Sample at more than twice the maximum frequency.
2. Sample at more than twice the bandwidth.

The first gives a sampling frequency of more than 920 Hz. The latter gives a sampling frequency of more than 20 Hz. There is clearly a benefit to the second part. However, traditionally, we would need to *downconvert the signal to baseband* in order to sample it at the lower rate. (See Lecture 3 notes).

What is the sampled spectrum, and what is the bandwidth of the sampled signal in radians/sample if the sampling rate is:

1. 1000 samples/sec?
2. 140 samples/sec?

Solution:

1. **Sample at 1000 samples/sec**: $\Omega_c = 2\pi \times \frac{455}{500} = \frac{455}{500}\pi$. The signal is very sparse - it occupies only $2\pi \frac{10 \text{ Hz}}{1000 \text{ samples/sec}} = \pi/50$ radians/sample.

2. **Sample at 140 samples/sec**: Copies of the entire frequency spectrum will appear in the frequency domain at multiples of 140 Hz or $2\pi 140$ radians/sec. The discrete-time frequency is modulo 2π, so the center frequency of the sampled signal is:

$$\Omega_c = 2\pi \times \frac{455}{140} \mod 2\pi = \frac{13\pi}{2} \mod 2\pi = \frac{\pi}{2}$$

   The value $\pi/2$ radians/sample is equivalent to 35 cycles/second. The bandwidth is $2\pi\frac{10}{140} = \pi/7$ radians/sample of sampled spectrum.

8 Orthogonality

From the first lecture, we talked about how digital communications is largely about using (and choosing some combination of) a discrete set of waveforms to convey information on the (continuous-time, real-valued) channel. At the receiver, the idea is to determine which waveforms were sent and in what quantities.

This choice of waveforms is partially determined by bandwidth, which we have covered. It is also determined by orthogonality. Loosely, orthogonality of waveforms means that some combination of the waveforms can be uniquely separated at the receiver. This is *the critical concept* needed to understand digital receivers. To build a better framework for it, we'll start with something you're probably more familiar with: vector multiplication.

8.1 Inner Product of Vectors

We often use an idea called *inner product*, which you're familiar with from vectors (as the dot product). If we have two vectors $\mathbf{x}$ and $\mathbf{y}$,

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \quad \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

Then we can take their inner product as:

$$\mathbf{x}^T\mathbf{y} = \sum_i x_i y_i$$

Note the ^{T} transpose 'in-between' an inner product. The inner product is also denoted $\langle \mathbf{x}, \mathbf{y} \rangle$. Finally, the *norm* of a vector is the square root of the inner product of a vector with itself,

$$\|\mathbf{x}\| = \sqrt{\langle \mathbf{x}, \mathbf{x} \rangle}$$

A norm is always indicated with the $\|\cdot\|$ symbols outside of the vector.

Example: Example Inner Products
Let $\mathbf{x} = [1,1,1]^T$, $\mathbf{y} = [0,1,1]^T$, $\mathbf{z} = [1,0,0]^T$. What are:

- $\langle \mathbf{x}, \mathbf{y} \rangle$?
- $\langle \mathbf{z}, \mathbf{y} \rangle$?
- $\|\mathbf{x}\|$?
- $\|\mathbf{z}\|$?

Recall the inner product between two vectors can also be written as

$$\mathbf{x}^T\mathbf{y} = \|\mathbf{x}\|\|\mathbf{y}\|\cos\theta$$

where θ is the angle between vectors $\mathbf{x}$ and $\mathbf{y}$. In words, the inner product is a measure of 'same-direction-ness': when positive, the two vectors are pointing in a similar direction, when negative, they are pointing in somewhat opposite directions; when zero, they're at cross-directions.

8.2 Inner Product of Functions

The inner product is not limited to vectors. It also is applied to the 'space' of *square-integrable real functions*, *i.e.*, those functions $x(t)$ for which

$$\int_{-\infty}^{\infty} x^2(t)dt < \infty.$$

The 'space' of *square-integrable complex functions*, *i.e.*, those functions $x(t)$ for which

$$\int_{-\infty}^{\infty} |x(t)|^2 dt < \infty$$

also have an inner product. These norms are:

$$\begin{aligned} \langle x(t), y(t) \rangle &= \int_{-\infty}^{\infty} x(t)y(t)dt \quad \text{Real functions} \\ \langle x(t), y(t) \rangle &= \int_{-\infty}^{\infty} x(t)y^*(t)dt \quad \text{Complex functions} \end{aligned}$$

As a result,

$$\begin{aligned} \|x(t)\| &= \sqrt{\int_{-\infty}^{\infty} x^2(t)dt} \quad \text{Real functions} \\ \|x(t)\| &= \sqrt{\int_{-\infty}^{\infty} |x(t)|^2 dt} \quad \text{Complex functions} \end{aligned}$$

Example: Sine and Cosine
Let

$$\begin{aligned} x(t) &= \begin{cases} \cos(2\pi t), & 0 < t \leq 1 \\ 0, & o.w. \end{cases} \\ y(t) &= \begin{cases} \sin(2\pi t), & 0 < t \leq 1 \\ 0, & o.w. \end{cases} \end{aligned}$$

What are $\|x(t)\|$ and $\|y(t)\|$? What is $\langle x(t), y(t)\rangle$?

Solution: Using $\cos^2 x = \frac{1}{2}(1+\cos 2x)$,

$$\|x(t)\| = \sqrt{\int_0^1 \cos^2(2\pi t)dt} = \sqrt{\frac{1}{2}\left(1+\frac{1}{2\pi}\sin(2\pi t)\right) dt\Big|_0^1} = \sqrt{1/2}$$

The solution for $\|y(t)\|$ also turns out to be $\sqrt{1/2}$. Using $\sin 2x = 2\cos x \sin x$, the inner product is,

$$\begin{aligned}
\langle x(t), y(t)\rangle &= \int_0^1 \cos(2\pi t)\sin(2\pi t)dt \\
&= \int_0^1 \frac{1}{2}\sin(4\pi t)dt \\
&= \frac{-1}{8\pi}\cos(4\pi t)\Big|_0^1 = \frac{-1}{8\pi}(1-1) = 0
\end{aligned}$$

8.3 Definitions

<u>Def'n:</u> *Orthogonal*
Two signals $x(t)$ and $y(t)$ are orthogonal if

$$\langle x(t), y(t)\rangle = 0$$

<u>Def'n:</u> *Orthonormal*
Two signals $x(t)$ and $y(t)$ are orthonormal if they are orthogonal and they both have norm 1, *i.e.*,

$$\|x(t)\| = 1 \quad \|y(t)\| = 1$$

(a) Are $x(t)$ and $y(t)$ from the sine/cosine example above orthogonal? (b) Are they orthonormal?

Solution: (a) Yes. (b) No. They could be orthonormal if they'd been scaled by $\sqrt{2}$.

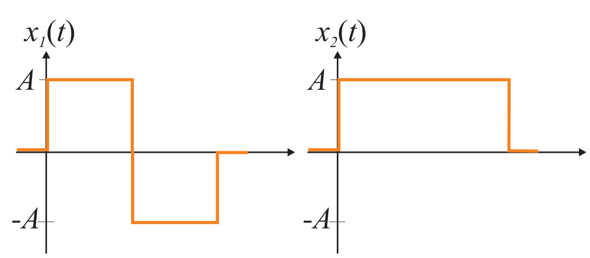


Figure 12: Two Walsh-Hadamard functions.

Example: Walsh-Hadamard 2 Functions
Let

$$x_1(t) = \begin{cases} A, & 0 < t \le 0.5 \\ -A, & 0.5 < t \le 1 \\ 0, & o.w. \end{cases}$$

$$x_2(t) = \begin{cases} A, & 0 < t \le 1 \\ 0, & o.w. \end{cases}$$

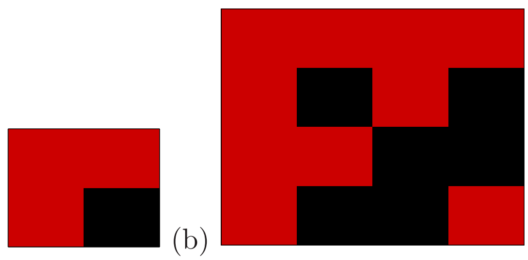
These are shown in Fig. 12. (a) Are $x_1(t)$ and $x_2(t)$ orthogonal? (b) Are they orthonormal?

Solution: (a) Yes. (b) Only if $A = 1$.

8.4 Orthogonal Sets

Def'n: *Orthogonal Set*
M signals $x_1(t), \ldots, x_M(t)$ are mutually orthogonal, or form an orthogonal set, if $\langle x_i(t), x_j(t) \rangle = 0$ for all $i \neq j$.

Def'n: *Orthonormal Set*
M mutually orthogonal signals $x_1(t), \ldots, x_M(t)$ are form an orthonormal set, if $\|x_i\| = 1$ for all i.



(a) (b)

Figure 13: Walsh-Hadamard 2-length and 4-length functions in image form. Each function is a row of the image, where crimson red is 1 and black is -1.

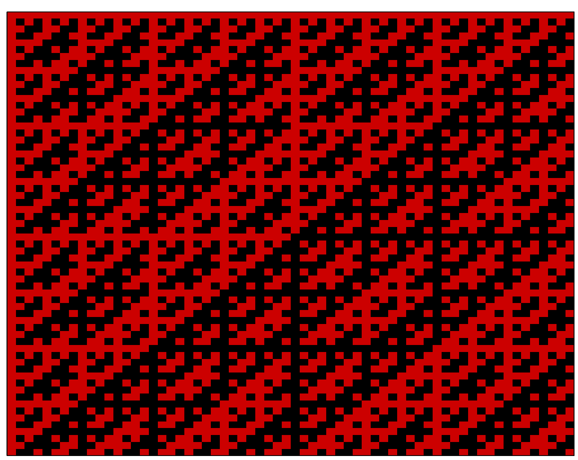


Figure 14: Walsh-Hadamard 64-length functions in image form.

Do the 64 WH functions form an orthonormal set? Why or why not?

Solution: Yes. Every pair i, j with $i \neq j$ agrees half of the time, and disagrees half of the time. So the function is half-time 1 and half-time -1 so they cancel and have inner product of zero. The norm of the function in row i is 1 since the amplitude is always $+1$ or -1.

Example: CDMA and Walsh-Hadamard
This example is just for your information (not for any test). The set of 64-length Walsh-Hadamard sequences

is used in CDMA (IS-95) in two ways:

1. Reverse Link: *Modulation*: For each group of 6 bits to be send, the modulator chooses one of the 64 possible 64-length functions.
2. Forward Link: *Channelization* or *Multiple Access*: The base station assigns each user a single WH function (channel). It uses 64 functions (channels). The data on each channel is multiplied by a Walsh function so that it is orthogonal to the data received by all other users.

Lecture 5

Today: (1) Orthogonal Signal Representations

9 Orthonormal Signal Representations

Last time we defined the inner product for vectors and for functions. We talked about orthogonality and orthonormality of functions and sets of functions. Now, we consider how to take a set of arbitrary signals and represent them as vectors in an *orthonormal basis*.

What are some common orthonormal signal representations?

- Nyquist sampling: sinc functions centered at sampling times.
- Fourier series: complex sinusoids at different frequencies
- Sine and cosine at the same frequency
- Wavelets

And, we will come up with others. Each one has a limitation – only a certain set of functions can be exactly represented in a particular signal representation. Essentially, we must limit the set of possible functions to a set. That is, some subset of all possible functions.

We refer to the set of arbitrary signals as:

$$\mathcal{S} = \{s_0(t), s_1(t), \ldots, s_{M-1}(t)\}$$

For example, a transmitter may be allowed to send any one of these M signals in order to convey information to a receiver.

We're going to introduce another set called an orthonormal basis:

$$\mathcal{B} = \{\phi_0(t), \phi_1(t), \ldots, \phi_{N-1}(t)\}$$

Each function in $\mathcal{B}$ is called a basis function. You can think of $\mathcal{B}$ as an unambiguous and useful language to represent the signals from our set $\mathcal{S}$. Or, in analogy to color printers, a color printer can produce any of millions of possible colors (signal set), only using black, red, blue, and green (basis set) (or black, cyan, magenta, yellow).

9.1 Orthonormal Bases

Def'n: *Span*
The span of the set $\mathcal{B}$ is the set of all functions which are linear combinations of the functions in $\mathcal{B}$. The span is referred to as Span$\{\mathcal{B}\}$ and is

$$\text{Span}\{\mathcal{B}\} = \left\{ \sum_{k=0}^{N-1} a_k \phi_k(t) \right\}_{a_0,\ldots,a_{N-1}\in\mathbb{R}}$$

Def'n: *Orthogonal Basis*
For any arbitrary set of signals $\mathcal{S} = \{s_i(t)\}_{i=0}^{M-1}$, the *orthogonal basis* is an orthogonal set of the smallest size N, $\mathcal{B} = \{\phi_i(t)\}_{i=0}^{N-1}$, for which $s_i(t) \in \text{Span}\{\mathcal{B}\}$ for every $i = 0, \ldots, M-1$. The value of N is called the dimension of the signal set.

Def'n: *Orthonormal Basis*
The *orthonormal basis* is an orthogonal basis in which each basis function has norm 1.

Notes:

- The orthonormal basis is not unique. If you can find one basis, you can use, for example, $\{-\phi_k(t)\}_{k=0}^{N-1}$ as another orthonormal basis.
- All orthogonal bases will have size N: no smaller basis can include all signals $s_i(t)$ in its span, and a larger set would not be a basis by definition.

The signal set might be a set of signals we can use to transmit particular bits (or sets of bits). (As the Walsh-Hadamard functions are used in IS-95 reverse link). An orthonormal basis for an arbitrary signal set tells us:

- how to build the receiver,
- how to represent the signals in 'signal-space', and
- how to quickly analyze the error rate of the scheme.

9.2 Synthesis

Consider one of the signals in our signal set, $s_i(t)$. Given that it is in the span of our basis $\mathbf{B}$, it can be represented as a linear combination of the basis functions,

$$s_i(t) = \sum_{k=0}^{N-1} a_{i,k} \phi_k(t)$$

The particular constants $a_{i,k}$ are calculated using the inner product:

$$a_{i,k} = \langle s_i(t), \phi_k(t) \rangle$$

The *projection* of one signal i onto basis function k is defined as $a_{i,k}\phi_k(t) = \langle s_i(t), \phi_k(t)\rangle \phi_k(t)$. Then the signal is equal to the sum of its projections onto the basis functions.

Why is this?

Solution: Since $s_i(t) \in \text{Span}\{\mathcal{B}\}$, we know there are some constants $\{a_{i,k}\}_k^{N-1}$ such that

$$s_i(t) = \sum_{k=0}^{N-1} a_{i,k}\phi_k(t)$$

Taking the inner product of both sides with $\phi_j(t)$,

$$\begin{aligned} \langle s_i(t), \phi_j(t)\rangle &= \left\langle \sum_{k=0}^{N-1} a_{i,k}\phi_k(t), \phi_j(t) \right\rangle \\ \langle s_i(t), \phi_j(t)\rangle &= \sum_{k=0}^{N-1} a_{i,k}\langle \phi_k(t), \phi_j(t)\rangle \\ \langle s_i(t), \phi_j(t)\rangle &= a_{i,j} \end{aligned}$$

So, now we can now represent a signal by a vector,

$$\mathbf{s}_i = [a_{i,0}, a_{i,1}, \ldots, a_{i,N-1}]^T$$

This and the basis functions **completely represent each signal**. Plotting $\{\mathbf{s}_i\}_i$ in an N dimensional grid is termed a *constellation diagram*. Generally, this space that we're plotting in is called *signal space*.

We can also synthesize any of our M signals in the signal set by adding the proper linear combination of the N bases. By choosing one of the M signals, we convey information, specifically, $log_2 M$ bits of information. (Generally, we choose M to be a power of 2 for simplicity).

See Figure 5.5 in Rice (page 254), which shows a block diagram of how a transmitter would synthesize one of the M signals to send, based on an input bitstream.

Example: Position-shifted pulses
Plot the signal space diagram for the signals,

$$\begin{aligned} s_0(t) &= u(t) - u(t-1) \\ s_1(t) &= u(t-1) - u(t-2) \\ s_2(t) &= u(t) - u(t-2) \end{aligned}$$

given the orthonormal basis,

$$\begin{aligned} \phi_0(t) &= u(t) - u(t-1) \\ \phi_1(t) &= u(t-1) - u(t-2) \end{aligned}$$

What are the signal space vectors, $\mathbf{s}_0$, $\mathbf{s}_1$, and $\mathbf{s}_2$?

Solution: They are $\mathbf{s}_0 = [1,0]^T$, $\mathbf{s}_1 = [0,1]^T$, and $\mathbf{s}_2 = [1,1]^T$. They are plotted in the signal space diagram in Figure 15.

Energy:

- Energy can be calculated in signal space as

$$\text{Energy}\{s_i(t)\} = \int_{-\infty} \infty s_i^2(t)dt = \sum_{k=0}^{N-1} a_{i,k}^2$$

  Proof?

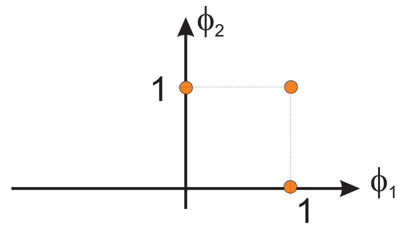


Figure 15: Signal space diagram for position-shifted signals example.

- We will find out later that it is the distances between the points in signal space which determine the bit error rate performance of the receiver.

$$d_{i,j} = \sqrt{\sum_{k=0}^{N-1} (a_{i,k} - a_{j,k})^2}$$

  for i, j in $\{0, \ldots, M-1\}$.

- Although different ON bases can be used, the energy and distance between points will not change.

Example: Amplitude-shifted signals
Now consider

$$\begin{aligned} s_0(t) &= 1.5[u(t) - u(t-1)] \\ s_1(t) &= 0.5[u(t) - u(t-1)] \\ s_2(t) &= -0.5[u(t) - u(t-1)] \\ s_3(t) &= -1.5[u(t) - u(t-1)] \end{aligned}$$

and the orthonormal basis,

$$\phi_1(t) = u(t) - u(t-1)$$

What are the signal space vectors for the signals $\{s_i(t)\}$? What are their energies?

Solution: $\mathbf{s}_0 = [1.5]$, $\mathbf{s}_1 = [0.5]$, $\mathbf{s}_2 = [-0.5]$, $\mathbf{s}_3 = [-1.5]$. See Figure 16.

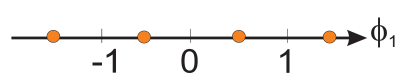


Figure 16: Signal space diagram for amplitude-shifted signals example.

Energies are just the squared magnitude of the vector: 2.25, 0.25, 0.25, and 2.25, respectively.

9.3 Analysis

At a receiver, our job will be to analyze the received signal (a function) and to decide which of the M possible signals was sent. This is the task of *analysis*. It turns out that an orthonormal bases makes our analysis very straightforward and elegant.

We won't receive exactly what we sent - there will be additional functions added to the signal function we sent.

- Thermal noise
- Interference from other users
- Self-interference

We won't get into these problems today. But, we might say that if we send signal m, *i.e.*, $s_m(t)$ from our signal set, then we would receive

$$r(t) = s_m(t) + w(t)$$

where the $w(t)$ is the sum of all of the additive signals that we did not intend to receive. But $w(t)$ might not (probably not) be in the span of our basis $\mathcal{B}$, so $r(t)$ would not be in Span$\{\mathcal{B}\}$ either. What is the best approximation to $r(t)$ in the signal space? Specifically, what is $\hat{r}(t) \in \text{Span}\{\mathcal{B}\}$ such that the energy of the difference between $\hat{r}(t)$ and $r(t)$ is minimized, *i.e.*,

$$\underset{\hat{r}(t)\in\text{Span}\{\mathcal{B}\}}{\text{argmin}} \int_{-\infty}^{\infty} |\hat{r}(t) - r(t)|^2 dt \tag{13}$$

Solution: Since $\hat{r}(t) \in \text{Span}\{\mathcal{B}\}$, it can be represented as a vector in signal space,

$$\mathbf{x} = [x_0, x_1, \ldots, x_{N-1}]^T.$$

and the synthesis equation is

$$\hat{r}(t) = \sum_{k=0}^{N-1} x_k \phi_k(t)$$

If you plug in the above expression for $\hat{r}(t)$ into (13), and then find the minimum with respect to each x_k, you'd see that the minimum error is at

$$x_k = \int_{-\infty}^{\infty} r(t)\phi_k(t) dt$$

that is, $x_k = \langle r(t), \phi_k(t) \rangle$, for $k = 0, \ldots, N$.

Example: Analysis using a Walsh-Hadamard 2 Basis
See Figure 17. Let $s_0(t) = \phi_0(t)$ and $s_1(t) = \phi_1(t)$. What is $\hat{r}(t)$?

Solution:

$$\hat{r}(t) = \begin{cases} 1, & 0 \le t < 1 \\ -1/2, & 1 \le t < 2 \\ 0, & o.w. \end{cases}$$

Lecture 6

Today: (1) Multivariate Distributions

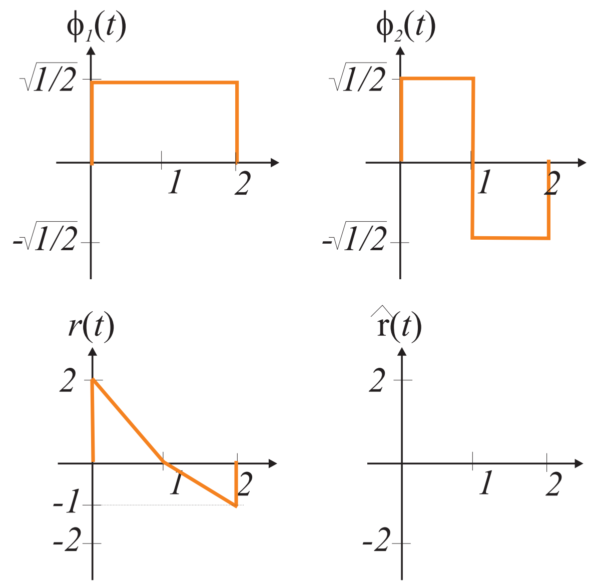


Figure 17: Signal and basis functions for Analysis example.

10 Multi-Variate Distributions

For two random variables X_1 and X_2,

- Joint CDF: $F_{X_1,X_2}(x_1,x_2) = P[\{X_1 \leq x_1\} \cap \{X_2 \leq x_2\}]$ It is the probability that both events happen simultaneously.
- Joint pmf: $P_{X_1,X_2}(x_1,x_2) = P[\{X_1 = x_1\} \cap \{X_2 = x_2\}]$ It is the probability that both events happen simultaneously.
- Joint pdf: $f_{X_1,X_2}(x_1,x_2) = \frac{\partial^2}{\partial x_1 \partial x_2} F_{X_1,X_2}(x_1,x_2)$

The pdf and pmf integrate / sum to one, and are non-negative. The CDF is non-negative and non-decreasing, with $\lim_{x_i \to .-\infty} F_{X_1,X_2}(x_1,x_2) = 0$ and $\lim_{x_1,x_2 \to .+\infty} F_{X_1,X_2}(x_1,x_2) = 1$.

Note: Two errors in Rice book dealing with the definition of the CDF. Eqn 4.9 should be:

$$F_X(x) = P[X \leq x] = \int_{-\infty}^{x} f_X(t)dt$$

and Eqn 4.15 should be:

$$F_X(x) = P[X \leq x]$$

To find the probability of an event, you integrate. For example, for event $B \in S$,

- Discrete case: $P[B] = \sum \sum_{(X_1,X_2) \in B} P_{X_1,X_2}(x_1,x_2)$
- Continuous Case: $P[B] = \int \int_{(x_1,x_2) \in B} f_{X_1,X_2}(x_1,x_2) dx_1 dx_2$

The marginal distributions are:

- Marginal pmf: $P_{X_2}(x_2) = \sum_{x_1 \in S_{X_1}} P_{X_1,X_2}(x_1, x_2)$
- Marginal pdf: $f_{X_2}(x_2) = \int_{x_1 \in S_{X_1}} f_{X_1,X_2}(x_1, x_2) dx_1$

Two random variables X_1 and X_2 are independent iff for all x_1 and x_2,

- $P_{X_1,X_2}(x_1, x_2) = P_{X_1}(x_1) P_{X_2}(x_2)$
- $f_{X_1,X_2}(x_1, x_2) = f_{X_2}(x_2) f_{X_1}(x_1)$

10.1 Random Vectors

Def'n: *Random Vector*
A random vector (R.V.) is a list of multiple random variables $X_1, X_2, \ldots, X_n$,

$$\mathbf{X} = [X_1, X_2, \ldots, X_n]^T$$

Here are the Models of R.V.s:

1. The CDF of R.V. $\mathbf{X}$ is $F_{\mathbf{X}}(\mathbf{x}) = F_{X_1,\ldots,X_n}(x_1, \ldots, x_n) = P[X_1 \le x_1, \ldots, X_n \le x_n]$.
2. The pmf of a discrete R.V. $\mathbf{X}$ is $P_{\mathbf{X}}(\mathbf{x}) = P_{X_1,\ldots,X_n}(x_1, \ldots, x_n) = P[X_1 = x_1, \ldots, X_n = x_n]$.
3. The pdf of a continuous R.V. $\mathbf{X}$ is $f_{\mathbf{X}}(\mathbf{x}) = f_{X_1,\ldots,X_n}(x_1, \ldots, x_n) = \frac{\partial^n}{\partial x_1 \cdots \partial x_n} F_{\mathbf{X}}(\mathbf{x})$.

10.2 Conditional Distributions

Given event $B \in S$ which has $P[B] > 0$, the joint probability conditioned on event B is

- Discrete case:
$$P_{X_1,X_2|B}(x_1, x_2) = \begin{cases} \frac{P_{X_1,X_2}(x_1,x_2)}{P[B]}, & (X_1, X_2) \in B \\ 0, & o.w. \end{cases}$$
- Continuous Case:
$$f_{X_1,X_2|B}(x_1, x_2) = \begin{cases} \frac{f_{X_1,X_2}(x_1,x_2)}{P[B]}, & (X_1, X_2) \in B \\ 0, & o.w. \end{cases}$$

Given r.v.s X_1 and X_2,

- Discrete case. The conditional pmf of X_1 given $X_2 = x_2$, where $P_{X_2}(x_2) > 0$, is
$$P_{X_1|X_2}(x_1|x_2) = P_{X_1,X_2}(x_1, x_2)/P_{X_2}(x_2)$$
- Continuous Case: The conditional pdf of X_1 given $X_2 = x_2$, where $f_{X_2}(x_2) > 0$, is
$$f_{X_1|X_2}(x_1|x_2) = f_{X_1,X_2}(x_1, x_2)/f_{X_2}(x_2)$$

Def'n: *Bayes' Law*
Bayes' Law is a reformulation of he definition of the marginal pdf. It is written either as:

$$f_{X_1,X_2}(x_1,x_2) = f_{X_2|X_1}(x_2|x_1)f_{X_1}(x_1)$$

or

$$f_{X_1|X_2}(x_1|x_2) = \frac{f_{X_2|X_1}(x_2|x_1)f_{X_1}(x_1)}{f_{X_2}(x_2)}$$

10.3 Simulation of Digital Communication Systems

A simulation of a digital communication system is often used to estimate a bit error rate. Each bit can either be demodulated without error, or with error. Thus the simulation of one bit is a Bernoulli trial. This trial E_i is in error ($E_1 = 1$) with probability p_e (the true bit error rate) and correct ($E_i = 0$) with probability $1 - p_e$. What type of random variable is E_i?

Simulations run many bits, say N bits through a model of the communication system, and count the number of bits that are in error. Let $S = \sum_{i=1}^{N} E_i$, and assume that $\{E_i\}$ are independent and identically distributed (i.i.d.).

1. What type of random variable is S?
2. What is the pmf of S?
3. What is the mean and variance of S?

Solution: E_i is called a Bernoulli r.v. and S is called a Binomial r.v., with pmf

$$P_S(s) = \binom{N}{s} p_e^s (1-p_e)^{N-s}$$

The mean of S is the the same as the mean of the sum of $\{E_i\}$,

$$\begin{aligned} E_S[S] &= E_{\{E_i\}}\left[\sum_{i=1}^{N} E_i\right] = \sum_{i=1}^{N} E_{E_i}[E_i] \\ &= \sum_{i=1}^{N} [(1-p)\cdot 0 + p \cdot 1] = Np \end{aligned}$$

We can find the variance of S the same way:

$$\begin{aligned} \text{Var}_S[S] &= \text{Var}_{\{E_i\}}\left[\sum_{i=1}^{N} E_i\right] = \sum_{i=1}^{N} \text{Var}_{E_i}[E_i] \\ &= \sum_{i=1}^{N} \left[(1-p)\cdot(0-p)^2 + p\cdot(1-p)^2\right] \\ &= \sum_{i=1}^{N} \left[(1-p)p^2 + (1-p)(p-p^2)\right] \\ &= Np(1-p) \end{aligned}$$

We may also be interested knowing how many bits to run in order to get an estimate of the bit error rate. For example, if we run a simulation and get zero bit errors, we won't have a very good idea of the bit error rate. Let T_1 be the time (number of bits) up to and including the first error.

1. What type of random variable is T_1?
2. What is the pmf of T_1?
3. What is the mean of T_1?

Solution: T_1 is a Geometric r.v. with pmf

$$P_{T_1}(t) = (1 - p_e)^{t-1} p_e$$

The mean of T_1 is

$$E[T_1] = \frac{1}{p_e}$$

Note the variance of T_1 is $\text{Var}[T_1] = (1 - p_e)/p_e^2$, so the standard deviation for very low p_e is almost the same as the expected value.

So, even if we run an experiment until the first bit error, our estimate of p_e will have relatively high variance.

10.4 Mixed Discrete and Continuous Joint Variables

This was not covered in ECE 5510, although it was in the Yates & Goodman textbook. We'll often have X_1 discrete and X_2 continuous.

Example: Digital communication system in noise
Let X_1 is the transmitted signal voltage, and X_2 is the received signal voltage, which is modeled as

$$X_2 = aX_1 + N$$

where N is a continuous-valued random variable representing the additive noise of the channel, and a is a constant which represents the attenuation of the channel. In a digital system, X_1 may take a discrete set of values, *e.g.*, $\{1.5, 0.5, -0.5, -1.5\}$. But the noise N may be continuous, *e.g.*, Gaussian with mean μ and variance σ^2. As long as $\sigma^2 > 0$, then X_2 will be continuous-valued.

Work-around: We will sometimes use a pdf for a discrete random variable. For instance, if

$$P_{X_1}(x_1) = \begin{cases} 0.6, & x_1 = 0 \\ 0.4, & x_1 = 1 \\ 0, & o.w. \end{cases}$$

then we would write a pdf with the probabilities as amplitudes of a dirac delta function centered at the value that has that probability,

$$f_{X_1}(x_1) = 0.6\delta(x_1) + 0.4\delta(x_1 - 1)$$

This pdf has integral 1, and non-negative values for all x_1. Any probability integral will return the proper value. For example, the probability that $-0.5 < x_1 < 0.5$ would be an integral that would return 0.6.

Example: Joint distribution of X_1, X_2
Consider the channel model $X_2 = X_1 + N$, and

$$f_N(n) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-n^2/(2\sigma^2)}$$

where σ^2 is some known variance, and the r.v. X_1 is independent of N with

$$P_{X_1}(x_1) = \begin{cases} 0.5, & x_1 = 0 \\ 0.5, & x_1 = 1 \\ 0, & o.w. \end{cases}.$$

1. What is the pdf of X_2?
2. What is the joint p.d.f. of (X_1, X_2) ?

Solution:

1. When two independent r.v.s are added, the pdf of the sum is the *convolution* of the pdfs of the inputs. Writing
$$f_{X_1}(x_1) = 0.5\delta(x_1) + 0.5\delta(x_1 - 1)$$
we convolve this with $f_N(n)$ above, to get
$$\begin{aligned} f_{X_2}(x_2) &= 0.5 f_N(x_2) + 0.5 f_N(x_2 - 1) \\ f_{X_2}(x_2) &= \frac{1}{2\sqrt{2\pi\sigma^2}} \left[e^{-x_2^2/(2\sigma^2)} + e^{-(x_2-1)^2/(2\sigma^2)} \right] \end{aligned}$$

2. What is the joint p.d.f. of (X_1, X_2) ? Since X_1 and X_2 are NOT independent, we cannot simply multiply the marginal pdfs together. It is necessary to use Bayes' Law.
$$f_{X_1,X_2}(x_1, x_2) = f_{X_2|X_1}(x_2|x_1) f_{X_1}(x_1)$$
Given a value of X_1 (either 0 or 1) we can write down the pdf of X_2. So break this into two cases:
$$\begin{aligned} f_{X_1,X_2}(x_1, x_2) &= \begin{cases} f_{X_2|X_1}(x_2|0) f_{X_1}(0), & x_1 = 0 \\ f_{X_2|X_1}(x_2|1) f_{X_1}(1), & x_1 = 1 \\ 0, & o.w. \end{cases} \\ f_{X_1,X_2}(x_1, x_2) &= \begin{cases} 0.5 f_N(x_2), & x_1 = 0 \\ 0.5 f_N(x_2 - 1), & x_1 = 1 \\ 0, & o.w. \end{cases} \\ f_{X_1,X_2}(x_1, x_2) &= 0.5 f_N(x_2)\delta(x_1) + \\ & \quad 0.5 f_N(x_2 - 1)\delta(x_1 - 1) \end{aligned}$$
These last two lines are completely equivalent. Use whichever seems more convenient for you. See Figure 18.

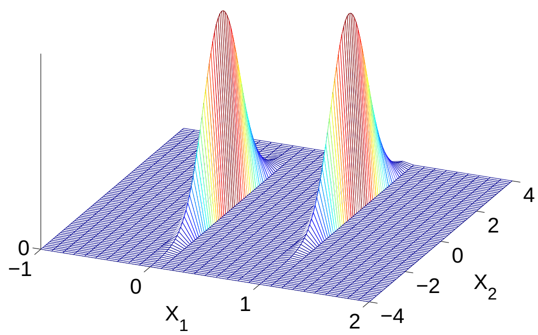


Figure 18: Joint pdf of X_1 and X_2, the input and output (respectively) of the example additive noise communication system, when $\sigma = 1$.

10.5 Expectation

Def'n: *Expected Value (Joint)*
The expected value of a function $g(X_1, X_2)$ of random variables X_1 and X_2 is given by,

1. Discrete: $E[g(X_1, X_2)] = \sum_{X_1 \in S_{X_1}} \sum_{X_2 \in S_{X_2}} g(X_1, X_2) P_{X_1,X_2}(x_1, x_2)$
2. Continuous: $E[g(X_1, X_2)] = \int_{X_1 \in S_{X_1}} \int_{X_2 \in S_{X_2}} g(X_1, X_2) f_{X_1,X_2}(x_1, x_2)$

Typical functions $g(X_1, X_2)$ are:

- Mean of X_1 or X_2: $g(X_1, X_2) = X_1$ or $g(X_1, X_2) = X_2$ will result in the means μ_{X_1} and μ_{X_2}.
- Variance (or 2nd central moment) of X_1 or X_2: $g(X_1, X_2) = (X_1 - \mu_{X_1})^2$ or $g(X_1, X_2) = (X_2 - \mu_{X_2})^2$. Often denoted $\sigma^2_{X_1}$ and $\sigma^2_{X_2}$.
- Covariance of X_1 and X_2: $g(X_1, X_2) = (X_1 - \mu_{X_1})(X_2 - \mu_{X_2})$.
- Expected value of the product of X_1 and X_2, also called the 'correlation' of X_1 and X_2: $g(X_1, X_2) = X_1 X_2$.

10.6 Gaussian Random Variables

For a single Gaussian r.v. X with mean μ_X and variance σ^2_X, we have the pdf,

$$f_X(x) = \frac{1}{\sqrt{2\pi\sigma^2_X}} e^{-\frac{(x-\mu_X)^2}{2\sigma^2}}$$

Consider Y to be Gaussian with mean 0 and variance 1. Then, the CDF of Y is denoted as $F_Y(y) = P[Y \le y] = \Phi(y)$. So, for X, which has non-zero mean and non-unit-variance, we can write its CDF as

$$F_X(x) = P[X \le x] = \Phi\left(\frac{x - \mu_X}{\sigma_X}\right)$$

You can prove this by showing that the event $X \leq x$ is the same as the event

$$\frac{X - \mu_X}{\sigma_X} \leq \frac{x - \mu_X}{\sigma_X}$$

Since the left-hand side is a unit-variance, zero mean Gaussian random variable, we can write the probability of this event using the unit-variance, zero mean Gaussian CDF.

10.6.1 Complementary CDF

The probability that a unit-variance, zero mean Gaussian r.v. X exceeds some value x is one minus the CDF, that is, $1 - \Phi(x)$. This is so common in digital communications, it is given its own name, $Q(x)$,

$$Q(x) = P[X > x] = 1 - \Phi(x)$$

What is $Q(x)$ in integral form?

$$Q(x) = \int_x^\infty \frac{1}{\sqrt{2\pi}} e^{-w^2/2} dw$$

For an Gaussian r.v. X with variance σ_X^2,

$$P[X > x] = Q\left(\frac{x - \mu_X}{\sigma_X}\right) = 1 - \Phi\left(\frac{x - \mu_X}{\sigma_X}\right)$$

10.6.2 Error Function

In math, in some texts, and in Matlab, the $Q(x)$ function is not used. Instead, there is a function called $\text{erf}(x)$

$$\text{erf}(x) \triangleq \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$$

Example: Relationship between $Q(\cdot)$ and erf$(\cdot)$
What is the functional relationship between $Q(\cdot)$ and $\text{erf}(\cdot)$?

Solution: Substituting $t = u/\sqrt{2}$ (and thus $dt = du/\sqrt{2}$),

$$\begin{aligned} \text{erf}(x) &\triangleq \frac{2}{\sqrt{2\pi}} \int_0^{\sqrt{2}x} e^{-u^2/2} du \\ &= 2 \int_0^{\sqrt{2}x} \frac{1}{\sqrt{2\pi}} e^{-u^2/2} du \\ &= 2\left(\Phi(\sqrt{2}x) - \frac{1}{2}\right) \end{aligned}$$

Equivalently, we can write $\Phi(\cdot)$ in terms of the $\text{erf}(\cdot)$ function,

$$\Phi(\sqrt{2}x) = \frac{1}{2}\text{erf}(x) + \frac{1}{2}$$

Finally let $y = \sqrt{2}x$, so that

$$\Phi(y) = \frac{1}{2}\text{erf}\left(\frac{y}{\sqrt{2}}\right) + \frac{1}{2}$$

Or in terms of $Q(\cdot)$,

$$Q(y) = 1 - \Phi(y) = \frac{1}{2} - \frac{1}{2}\text{erf}\left(\frac{y}{\sqrt{2}}\right) \tag{14}$$

You should go to Matlab and create a function Q(y) which implements:

```
function rval = Q(y)
rval = 1/2 - 1/2 .* erf(y./sqrt(2));
```

10.7 Examples

Example: Probability of Error in Binary Example
As in the previous example, we have a model system in which the receiver sees $X_2 = X_1 + N$. Here, $X_1 \in \{0, 1\}$ with equal probabilities and N is independent of X_1 and zero-mean Gaussian with variance 1/4. The receiver decides as follows:

- If $X_2 \leq 1/3$, then decide that the transmitter sent a '0'.
- If $X_2 > 1/3$, then decide that the transmitter sent a '1'.

1. Given that $X_1 = 1$, what is the probability that the receiver decides that a '0' was sent?
2. Given that $X_1 = 0$, what is the probability that the receiver decides that a '1' was sent?

Solution:

1. Given that $X_1 = 1$, since $X_2 = X_1 + N$, it is clear that X_2 is also a Gaussian r.v. with mean 1 and variance 1/4. Then the probability that the receiver decides '0' is the probability that $X_2 \leq 1/3$,

$$\begin{aligned} P[error|X_1 = 1] &= P[X_2 \leq 1/3] \\ &= P\left[\frac{X_2 - 1}{\sqrt{1/4}} \leq \frac{1/3 - 1}{\sqrt{1/4}}\right] \\ &= 1 - Q((-2/3)/(1/2)) \\ &= 1 - Q(-4/3) \end{aligned}$$

2. Given that $X_1 = 0$, the probability that the receiver decides '1' is the probability that $X_2 > 1/3$,

$$\begin{aligned} P[error|X_1 = 0] &= P[X_2 > 1/3] \\ &= P\left[\frac{X_2}{\sqrt{1/4}} > \frac{1/3}{\sqrt{1/4}}\right] \\ &= Q((1/3)/(1/2)) \\ &= Q(2/3) \end{aligned}$$

10.8 Gaussian Random Vectors

Def'n: *Multivariate Gaussian R.V.*
An n-length R.V. $\mathbf{X}$ is multivariate Gaussian with mean $\mu_{\mathbf{X}}$, and covariance matrix $C_{\mathbf{X}}$ if it has the pdf,

$$f_{\mathbf{X}}(\mathbf{x}) = \frac{1}{\sqrt{(2\pi)^n det(C_{\mathbf{X}})}} \exp\left[-\frac{1}{2}(\mathbf{x}-\mu_{\mathbf{X}})^T C_{\mathbf{X}}^{-1}(\mathbf{x}-\mu_{\mathbf{X}})\right]$$

where $det()$ is the determinant of the covariance matrix, and $C_{\mathbf{X}}^{-1}$ is the inverse of the covariance matrix.

Any linear combination of jointly Gaussian random variables is another jointly Gaussian random variable. For example, if we have a matrix A and we let a new random vector $\mathbf{Y} = A\mathbf{X}$, then $\mathbf{Y}$ is also a Gaussian random vector with mean $A\mu_{\mathbf{X}}$ and covariance matrix $AC_{\mathbf{X}}A^T$.

If the elements of $\mathbf{X}$ were independent random variables, the pdf would be the product of the individual pdfs (as with any random vector) and in this case the pdf would be:

$$f_{\mathbf{X}}(\mathbf{x}) = \frac{1}{\sqrt{(2\pi)^n \prod_i \sigma_i^2}} \exp\left[-\sum_{i=1}^{n} \frac{(x_i-\mu_{X_i})^2}{2\sigma_{X_i}^2}\right]$$

Section 4.3 spends some time with 2-D Gaussian random vectors, which is the dimension with which we spend most of our time in this class.

10.8.1 Envelope

The envelope of a 2-D random vector is its distance from the origin $(0,0)$. Specifically, if the vector is (X_1, X_2), the envelope R is

$$R = \sqrt{X_1^2 + X_2^2} \tag{15}$$

If both X_1 and X_2 are i.i.d. Gaussian with mean 0 and variance σ^2, the pdf of R is

$$f_R(r) = \begin{cases} \frac{r}{\sigma^2}\exp\left(-\frac{r^2}{2\sigma^2}\right), & r \ge 0 \\ 0, & o.w. \end{cases}$$

This is the *Rayleigh* distribution. If instead X_1 and X_2 are independent Gaussian with means μ_1 and μ_2, respectively, and identical variances σ^2, the envelope R would now have a Rice (a.k.a. Rician) distribution,

$$f_R(r) = \begin{cases} \frac{r}{\sigma^2}\exp\left(-\frac{r^2+s^2}{2\sigma^2}\right) I_0\left(\frac{rs}{\sigma_2}\right), & r \ge 0 \\ 0, & o.w. \end{cases}$$

where $s^2 = \mu_1^2 + \mu_2^2$, and $I_0(\cdot)$ is the zeroth order modified Bessel function of the first kind.

Note that both the Rayleigh and Rician pdfs can be derived from the Gaussian distribution and (15) using the transformation of random variables methods studied in ECE 5510. Both pdfs are sometimes needed to derive probability of error formulas.

Lecture 7

Today: (1) Matlab Simulation (2) Random Processes (3) Correlation and Matched Filter Receivers

11 Random Processes

A *random process* $X(t,s)$ is a function of time t and outcome (realization) s. Outcome s lies in sample space S. Outcome or realization is included because there could be ways to record X even at the same time. For example, multiple receivers would record different noisy signals even of the same transmission.

A *random sequence* $X(n,s)$ is a sequence of random variables indexed by time index n and realization $s \in S$.

Typically, we omit the "s" when writing the name of the random process or random sequence, and abbreviate it to $X(t)$ or $X(n)$, respectively.

11.1 Autocorrelation and Power

<u>Def'n:</u> *Mean Function*
The mean function of the random process $X(t,s)$ is

$$\mu_X(t) = E\left[X(t,s)\right]$$

Note the mean is taken over all possible realizations s. If you record one signal over all time t, you don't have anything to average to get the mean function $\mu_X(t)$.

<u>Def'n:</u> *Autocorrelation Function*
The autocorrelation function of a random process $X(t)$ is

$$R_X(t,\tau) = E\left[X(t)X(t-\tau)\right]$$

The autocorrelation of a random sequence $X(n)$ is

$$R_X(n,k) = E\left[X(n)X(n-k)\right]$$

<u>Def'n:</u> *Wide-sense stationary (WSS)*
A random process is wide-sense stationary (WSS) if

1. $\mu_X = \mu_X(t) = E\left[X(t)\right]$ is independent of t.
2. $R_X(t,\tau)$ depends only on the time difference τ and not on t. We then denote the autocorrelation function as $R_X(\tau)$.

A random process is wide-sense stationary (WSS) if

1. $\mu_X = \mu_X(n) = E\left[X(n)\right]$ is independent of n.
2. $R_X(n,k)$ depends only on k and not on n. We then denote the autocorrelation function as $R_X(k)$.

The power of a signal is given by $R_X(0)$.

We can estimate the autocorrelation function of an ergodic, wide-sense stationary random process from one realization of a power-type signal $x(t)$,

$$R_X(\tau) = \lim_{T\to\infty} \frac{1}{T} \int_{t=-T/2}^{T/2} x(t)x(t-\tau)dt$$

This is not a critical topic for this class, but we now must define "ergodic". For an *ergodic* random process, its time averages are equivalent to its ensemble averages. An example that helps demonstrate the difference between time averages and ensemble averages is the non-ergodic process of opinion polling. Consider what happens when a pollster takes either a time-average or a ensemble average:

- Time Average: The pollster asks the same person, every day for N days, whether or not she will vote for person X. The time average is taken by dividing the total number of "Yes"s by N.
- Ensemble Average: The pollster asks N different people, on the same day, whether or not they will vote for person X. The ensemble average is taken by dividing the total number of "Yes"s by N.

Will they come up with the same answer? Perhaps; but probably not. This makes the process non-ergodic.

Power Spectral Density We have seen that for a WSS random process $X(t)$ (and for a random sequence $X(n)$ we compute the power spectral density as,

$$\begin{aligned} S_X(f) &= \mathfrak{F}\{R_X(\tau)\} \\ S_X(e^{j\Omega}) &= \text{DTFT}\{R_X(k)\} \end{aligned}$$

11.1.1 White Noise

Let $X(n)$ be a WSS random sequence with autocorrelation function

$$R_X(k) = E[X(n)X(n-k)] = \sigma^2\delta(k)$$

This says that each element of the sequence $X(n)$ has zero covariance with every other sample of the sequence, *i.e.*, it is uncorrelated with $X(m), m \neq n$.

- Does this mean it is independent of every other sample?
- What is the PSD of the sequence?

This sequence is commonly called 'white noise' because it has equal parts of every frequency (analogy to light).

For continuous time signals, white noise is a commonly used approximation for thermal noise. Let $X(t)$ be a WSS random process with autocorrelation function

$$R_X(\tau) = E[X(t)X(t-\tau)] = \sigma^2\delta(\tau)$$

- What is the PSD of the sequence?
- What is the power of the signal?

Realistically, thermal noise is not constant in frequency, because as frequency goes very high (*e.g.*, 10^{15} Hz), the power spectral density goes to zero. The Rice book (4.5.2) has a good analysis of the physics of thermal noise.

12 Correlation and Matched-Filter Receivers

We've been talking about how our digital transmitter can transmit at each time one of a set of signals $\{s_i(t)\}_{i=1}^M$. This transmission conveys to us which of M messages we want to send, that is, $\log_2 M$ bits of information.

At the receiver, we receive signal $s_i(t)$ scaled and corrupted by noise:

$$r(t) = bs_i(t) + n(t)$$

How do we decide which signal i was transmitted?

12.1 Correlation Receiver

What we've been talking about it the inner product. In other terms, the inner product is a correlation:

$$\langle r(t), s_i(t) \rangle = \int_{-\infty}^{\infty} r(t) s_i(t) dt$$

But we want to build a receiver that does the minimum amount of calculation. If $s_i(t)$ are non-orthogonal, then we can reduce the amount of correlation (and thus multiplication and addition) done in the receiver by instead, correlating with the basis functions, $\{\phi_k(t)\}_{k=1}^N$. Correlation with the basis functions gives

$$x_k = \langle r(t), \phi_k(t) \rangle = \int_{-\infty}^{\infty} r(t) \phi_k(t) dt$$

for $k = 1 \ldots N$. As notation,

$$\mathbf{x} = [x_1, x_2, \ldots, x_N]^T$$

Now that we've done these N correlations (inner products), we can compute the estimate of the received signal as

$$\hat{r}(t) = \sum_{k=0}^{K-1} x_k \phi_k(t)$$

This is the 'correlation receiver', shown in Figure 5.1.6 in Rice (page 226).

Noise-free channel Since $r(t) = b s_i(t) + n(t)$, if $n(t) = 0$ and $b = 1$ then

$$\mathbf{x} = \mathbf{a}_i$$

where $\mathbf{a}_i$ is the signal space vector for $s_i(t)$.

Noisy Channel Now, let $b = 1$ but consider $n(t)$ to be a white Gaussian random process with zero mean and PSD $S_N(f) = N_0/2$. Define

$$n_k = \langle n(t), \phi_k \rangle = \int_{-\infty}^{\infty} n(t) \phi_k(t) dt.$$

What is $\mathbf{x}$ in terms of $\mathbf{a}_i$ and the noise $\{n_k\}$? What are the mean and covariance of $\{n_k\}$?

Solution:

$$\begin{aligned} x_k &= \int_{-\infty}^{\infty} r(t) \phi_k(t) dt \\ &= \int_{-\infty}^{\infty} [s_i(t) + n(t)] \phi_k(t) dt \\ &= a_{i,k} + \int_{-\infty}^{\infty} n(t) \phi_k(t) dt \\ &= a_{i,k} + n_k \end{aligned}$$

First, n_k is zero mean:

$$E[n_k] = \int_{-\infty}^{\infty} E[n(t)] \phi_k(t) dt = 0$$

Next we can show that $n_1, \ldots, n_N$ are i.i.d. by calculating the autocorrelation $R_n(m,k)$.

$$
\begin{aligned}
R_n(m,k) &= E\left[n_k n_m\right] \\
&= \int_{t=-\infty}^{\infty} \int_{\tau=-\infty}^{\infty} E\left[n(t)n(\tau)\right] \phi_k(t)\phi_m(\tau) d\tau dt \\
&= \int_{t=-\infty}^{\infty} \int_{\tau=-\infty}^{\infty} \frac{N_0}{2}\delta(t-\tau)\phi_k(t)\phi_m(\tau) d\tau dt \\
&= \frac{N_0}{2} \int_{t=-\infty}^{\infty} \phi_k(t)\phi_m(t) d\tau dt \\
&= \frac{N_0}{2}\delta_{k,m} = \begin{cases} \frac{N_0}{2}, & m = k \\ 0, & o.w. \end{cases}
\end{aligned}
$$

Is $\{n_k\}$ WSS? Is it a Gaussian random sequence?

Since the noise components are independent, then x_k are also independent. (Why?) What is the pdf of x_k? What is the pdf of $\mathbf{x}$?

Solution: x_k are independent because $x_k = a_{i,k} + n_k$ and $a_{i,k}$ is a deterministic constant. Then since n_k is Gaussian,

$$
f_{X_k}(x_k) = \frac{1}{\sqrt{2\pi(N_0/2)}} e^{-\frac{(x_k - a_{i,k})^2}{2(N_0/2)}}
$$

And, since $\{X_k\}$ are independent,

$$
\begin{aligned}
f_{\mathbf{x}}(\mathbf{x}) &= \prod_{k=1}^{N} f_{X_k}(x_k) \\
&= \prod_{k=1}^{N} \frac{1}{\sqrt{2\pi(N_0/2)}} e^{-\frac{(x_k - a_{i,k})^2}{2(N_0/2)}} \\
&= \frac{1}{\left[2\pi(N_0/2)\right]^{N/2}} e^{-\frac{\sum_{k=1}^{N}(x_k - a_{i,k})^2}{2(N_0/2)}}
\end{aligned}
$$

An example is in `ece5510_lec07.m`.

12.2 Matched Filter Receiver

Above we said that

$$
x_k = \int_{t=-\infty}^{\infty} r(t)\phi_k(t) dt
$$

But there really are finite limits – let's say that the signal has a duration T, and then rewrite the integral as

$$
x_k = \int_{t=0}^{T} r(t)\phi_k(t) dt
$$

This can be written as

$$
x_k = \int_{t=0}^{T} r(t)h_k(T-t) dt
$$

where $h_k(t) = \phi_k(T-t)$. (Plug in $(T-t)$ in this formula and the Ts cancel out and only positive t is left.) This is the output of a convolution, taken at time T,

$$x_k = r(t) \star h_k(t)|_{t=T}$$

Or equivalently

$$x_k = r(t) \star \phi_k(T-t)|_{t=T}$$

This is Rice Section 5.1.4.
Notes:

- The x_k can be seen as the output of a 'matched' filter at time T.
- This works at time T. The output for other times will be different in the correlation and matched filter.
- These are just two different physical implementations. We might, for example, have a physical filter with the impulse response $\phi_k(T-t)$ and thus it is easy to do a matched filter implementation.
- It may be easier to 'see' why the correlation receiver works.

Try out the Matlab code, `correlation_and_matched_filter_rx.m`, which is posted on WebCT.

12.3 Amplitude

Before we said $b = 1$. A real channel attenuates! The job of our receiver will be to amplify the signal by approximately $1/b$. This effectively multiplies the noise. In general, textbooks will assume that the automatic gain control (AGC) works properly, and then will assume the noise signal is multiplied accordingly.

Lecture 8

Today: (1) Optimal Binary Detection

12.4 Review

At a digital transmitter, we can transmit at each time one of a set of signals $\{s_i(t)\}_{i=0}^{M-1}$. This transmission conveys to us which of M messages we want to send, that is, $\log_2 M$ bits of information.

At the receiver, we assume we receive signal $s_i(t)$ corrupted by noise:

$$r(t) = s_i(t) + n(t)$$

How do we decide which signal $i \in \{0, \ldots, M-1\}$ was transmitted? We split the task into down-conversion, gain control, correlation or matched filter reception, and detection, as shown in Figure 19.

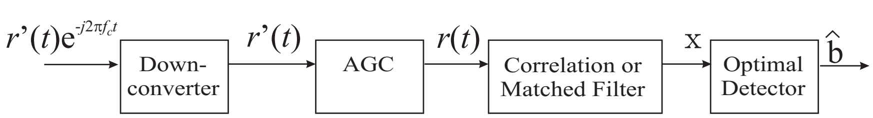


Figure 19: A block diagram of the receiver blocks which discussed in lecture 7 & 8.

12.5 Correlation Receiver

Our signal set can be represented by the orthonormal basis functions, $\{\phi_k(t)\}_{k=0}^{K-1}$. Correlation with the basis functions gives

$$x_k = \langle r(t), \phi_k(t) \rangle = a_{i,k} + n_k$$

for $k = 0 \ldots K-1$. We denote vectors:

$$\begin{aligned} \mathbf{x} &= [x_0, x_1, \ldots, x_{K-1}]^T \\ \mathbf{a}_i &= [a_{i,0}, a_{i,1}, \ldots, a_{i,K-1}]^T \\ \mathbf{n} &= [n_0, n_1, \ldots, n_{K-1}]^T \end{aligned}$$

Hopefully, $\mathbf{x}$ and $\mathbf{a}_i$ should be close since i was actually sent. In an example Matlab simulation, `ece5520_lec07.m`, we simulated sending $\mathbf{a}_i = [1,1]^T$ and receiving $r(t)$ (and thus $\mathbf{x}$) in noise.

In general, the pdf of $\mathbf{x}$ is multivariate Gaussian with each component x_k independent, because:

- $x_k = a_{i,k} + n_k$
- $a_{i,k}$ is a deterministic constant
- The $\{n_k\}$ are i.i.d. Gaussian.

The joint pdf of $\mathbf{x}$ is

$$f_{\mathbf{X}}(\mathbf{x}) = \prod_{k=0}^{K-1} f_{X_k}(x_k) = \frac{1}{[2\pi(N_0/2)]^{N/2}} e^{-\frac{\sum_{k=1}^{N}(x_k - a_{i,k})^2}{2(N_0/2)}} \quad (16)$$

We showed that there are two ways to implement this: the correlation receiver, and the matched filter receiver. Both result in the same output $\mathbf{x}$. We simulated this in the Matlab code `correlation_and_matched_filter_rx.m`.

13 Optimal Detection

We receive $r(t)$ and use the matched filter or correlation receiver to calculate $\mathbf{x}$. Now, how exactly do we decide which $s_i(t)$ was sent?

Consider this in signal space, as shown in Figure 20.

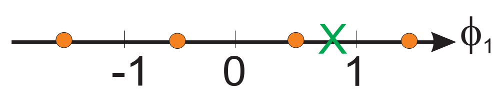


Figure 20: Signal space diagram of a PAM system with an X to mark the receiver output $\mathbf{x}$.

Given the signal space diagram of the transmitted signals and the received signal space vector $\mathbf{x}$, what rules should we have to decide on i? This is the topic of detection. Optimal detection uses rules designed to minimize the probability that a symbol error will occur.

13.1 Overview

We'll show two things:

- If each symbol i is equally probable, then the decision will be, pick the i with $\mathbf{a}_i$ closest to $\mathbf{x}$.
- If symbols aren't equally probable, then we'll need to shift the decision boundaries.

Detection theory is a major learning objective of this course. So even though it is somewhat theoretical, it is very applicable in digital receivers. Further, detection is applicable in a wide variety of problems, for example,

- Medical applications: Does an image show a tumor? Does a blood sample indicate a disease? Does a breathing sound indicate an obstructed airway?
- Communications applications: Receiver design, signal detection.
- Radar applications: Obstruction detection, motion detection.

13.2 Bayesian Detection

When we say 'optimal detection' in the Bayesian detection framework, we mean that we want the smallest probability of error. The probability of error is denoted

$$P[\text{error}]$$

By error, we mean that a different signal was detected than the one that was sent. At the start of every detection problem, we list the events that could have occurred, *i.e.*, the symbols that could have been sent. We follow all detection and statistics textbooks and label these classes H_i, and write:

$$\begin{aligned} H_0 : &\quad r(t) = s_0(t) + n(t) \\ H_1 : &\quad r(t) = s_1(t) + n(t) \\ \cdots &\quad \cdots \\ H_{M-1} : &\quad r(t) = s_{M-1}(t) + n(t) \end{aligned}$$

This must be a complete listing of events. That is, the events $H_0 \cup H_1 \cup \cdots \cup H_{M-1} = S$, where the $\cup$ means union, and S is the complete event space.

14 Binary Detection

Let's just say for now that there are only two signals $s_0(t)$ and $s_1(t)$, and only one basis function $\phi_0(t)$. Then, instead of vectors $\mathbf{x}$, $\mathbf{a}_0$ and $\mathbf{a}_1$, and $\mathbf{n}$, we'll have only scalars: x, a_0 and a_1, and n. When we talk about them as random variables, we'll substitute N for n and X for x. We need to decide from X whether $s_0(t)$ or $s_1(t)$ was sent.

The event listing is,

$$\begin{aligned} H_0 : &\quad r(t) = s_0(t) + n(t) \\ H_1 : &\quad r(t) = s_1(t) + n(t) \end{aligned}$$

Equivalently,

$$\begin{aligned} H_0 : &\quad X = a_0 + N \\ H_1 : &\quad X = a_1 + N \end{aligned}$$

We use the law of total probability to say that

$$P[\text{error}] = P[\text{error} \cap H_0] + P[\text{error} \cap H_1] \tag{17}$$

Where the cap means 'and'. Then using Bayes' Law,

$$P[\text{error}] = P[\text{error}|H_0]\, P[H_0] + P[\text{error}|H_1]\, P[H_1]$$

14.1 Decision Region

We're making a decision based only on X. Over some set R_0 of values of X, we'll decide that H_0 happened ($s_0(t)$ was sent). Over a different set R_1 of values, we'll decide H_1 occurred (that $s_1(t)$ was sent). We can't be indecisive, so

- There is no overlap: $R_0 \cap R_1 = \emptyset$.
- There are no values of x disregarded: $R_0 \cup R_1 = S$.

14.2 Formula for Probability of Error

So the probability of error is

$$P[\text{error}] = P[X \in R_1 | H_0] P[H_0] + P[X \in R_0 | H_1] P[H_1] \tag{18}$$

The probability that X is in R_1 is one minus the probability that it is in R_0, since the two are complementary sets.

$$\begin{aligned} P[\text{error}] &= (1 - P[X \in R_0|H_0])P[H_0] + P[X \in R_0|H_1] P[H_1] \\ P[\text{error}] &= P[H_0] - P[X \in R_0|H_0] P[H_0] + P[X \in R_0|H_1] P[H_1] \end{aligned}$$

Now note that probabilities that $X \in R_0$ are integrals over the event (region) R_0.

$$\begin{aligned} P[\text{error}] &= P[H_0] - \int_{x \in R_0} f_{X|H_0}(x|H_0) P[H_0]\, dx \\ &\quad + \int_{x \in R_0} f_{X|H_1}(x|H_1) P[H_1]\, dx \\ &= P[H_0] \\ &\quad + \int_{x \in R_0} \left(f_{X|H_1}(x|H_1) P[H_1] - f_{X|H_0}(x|H_0) P[H_0] \right) dx \end{aligned} \tag{19}$$

We've got a lot of things in the expression in (19), but the only thing we can change is the region R_0. Everything else is determined by the time we get to this point. So the question is, how do you pick R_0 to minimize (19)?

14.3 Selecting R_0 to Minimize Probability of Error

We can see what the integrand looks like. Figure 21 shows the conditional probability density functions. Figure **??** shows the joint densities (the conditional pdfs multiplied by the bit probabilities $P[H_0]$ and $P[H_1]$. Finally, Figure **??** shows the full integrand of (19), the difference between the joint densities.

We can pick R_0 however we want - we just say what region of x, and the integral in (19) will integrate over it. The objective is to minimize the probability of error. Which x's should we include in the region? Should we include x which has a positive value of the integrand? Or should we include the parts of x which have a negative value of the integrand?

Solution: Select R_0 to be all x such that the integrand is negative.

Then R_0 is the area in which

$$f_{X|H_0}(x|H_0) P[H_0] > f_{X|H_1}(x|H_1) P[H_1]$$

If $P[H_0] = P[H_1]$, then this is the region in which X is more probable given H_0 than given H_1.

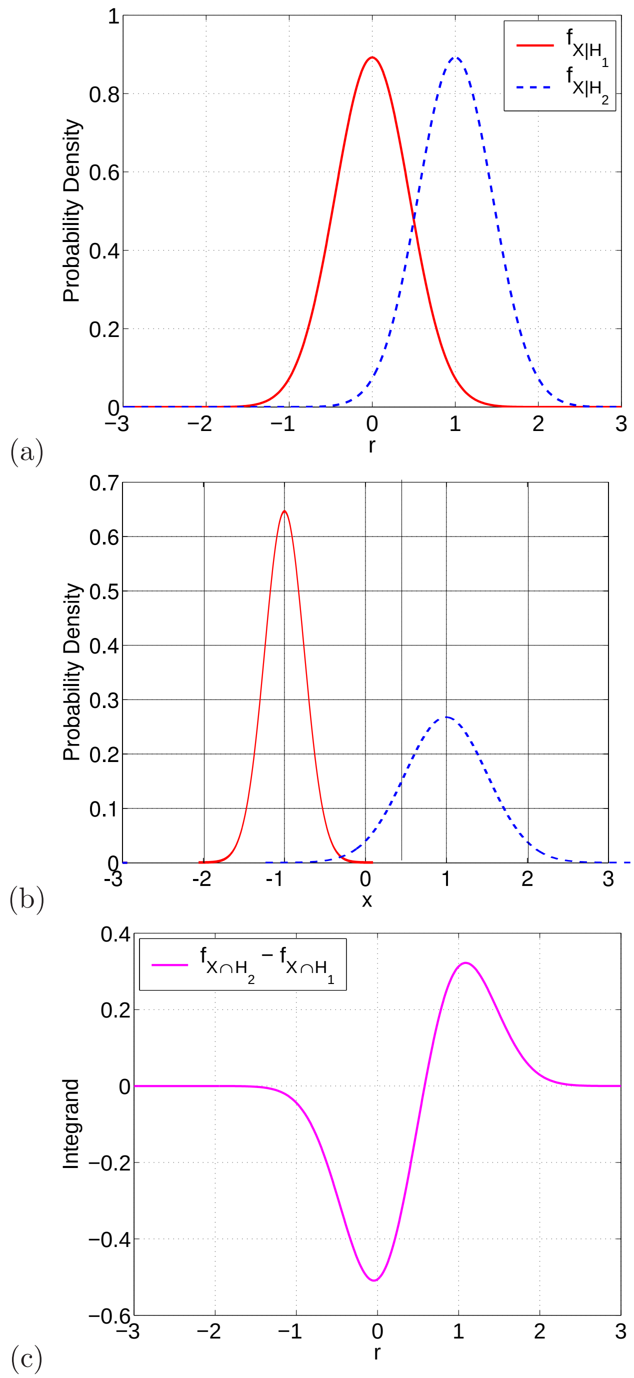


Figure 21: The (a) conditional p.d.f.s (likelihood functions) $f_{X|H_0}(x|H_0)$ and $f_{X|H_1}(x|H_1)$, (b) joint p.d.f.s $f_{X|H_0}(x|H_0)P[H_0]$ and $f_{X|H_1}(x|H_1)P[H_1]$, and (c) difference between the joint p.d.f.s, $f_{X|H_1}(x|H_1)P[H_1] - f_{X|H_0}(x|H_0)P[H_0]$, which is the integrand in (19).

Rearranging the terms,

$$\frac{f_{X|H_1}(x|H_1)}{f_{X|H_0}(x|H_0)} < \frac{P[H_0]}{P[H_1]} \tag{20}$$

The left hand side is called the likelihood ratio. The right hand side is a threshold. Whenever x indicates that the likelihood ratio is less than the threshold, then we'll decide H_0, *i.e.*, that $s_0(t)$ was sent. Otherwise, we'll decide H_1, *i.e.*, that $s_1(t)$ was sent.

Equation (20) is a very general result, applicable no matter what conditional distributions x has.

14.4 Log-Likelihood Ratio

For the Gaussian distribution, the math gets much easier if we take the log of both sides. Why can we do this?

Solution: 1. Both sides are positive, 2. The log() function is strictly increasing.

Now, the *log-likelihood ratio* is

$$\log \frac{f_{X|H_1}(x|H_1)}{f_{X|H_0}(x|H_0)} < \log \frac{P[H_0]}{P[H_1]}$$

14.5 Case of $a_0 = 0$, $a_1 = 1$ in Gaussian noise

In this example, $n \sim \mathcal{N}(0, \sigma_N^2)$. In addition, assume for a minute that $a_0 = 0$ and $a_1 = 1$. What is:

1. The log of a Gaussian pdf?
2. The log-likelihood ratio?
3. The decision regions for x?

Solution: What is the log of a Gaussian pdf?

$$\begin{aligned} \log f_{X|H_0}(x|H_0) &= \log \left[\frac{1}{\sqrt{2\pi\sigma_N^2}} e^{-\frac{x^2}{2\sigma_N^2}} \right] \\ &= -\frac{1}{2}\log(2\pi\sigma_N^2) - \frac{x^2}{2\sigma_N^2} \end{aligned} \tag{21}$$

The $\log f_{X|H_1}(x|H_1)$ term will be the same but with $(x-1)^2$instead of x^2. Continuing with the log-likelihood ratio,

$$\begin{aligned} \log f_{X|H_1}(x|H_1) - \log f_{X|H_0}(x|H_0) &< \log \frac{P[H_0]}{P[H_1]} \\ \frac{x^2}{2\sigma_N^2} - \frac{(x-1)^2}{2\sigma_N^2} &< \log \frac{P[H_0]}{P[H_1]} \\ x^2 - (x-1)^2 &< 2\sigma_N^2 \log \frac{P[H_0]}{P[H_1]} \\ 2x - 1 &< 2\sigma_N^2 \log \frac{P[H_0]}{P[H_1]} \\ x &< \frac{1}{2} + \sigma_N^2 \log \frac{P[H_0]}{P[H_1]} \end{aligned}$$

In the end result, there is a simple test for x - if it is below the *decision threshold*, decide H_0. If it is above the decision threshold,

$$x > \frac{1}{2} + \sigma_N^2 \log \frac{P[H_0]}{P[H_1]}$$

decide H_1. Rather than writing both inequalities each time, we use the following notation:

$$x \underset{H_0}{\overset{H_1}{\gtrless}} \frac{1}{2} + \sigma_N^2 \log \frac{P[H_0]}{P[H_1]}$$

This completely describes the detector receiver.

For simplicity, we also write $x \underset{H_0}{\overset{H_1}{\gtrless}} \gamma$ where

$$\gamma = \frac{1}{2} + \sigma_N^2 \log \frac{P[H_0]}{P[H_1]} \tag{22}$$

14.6 General Case for Arbitrary Signals

If, instead of $a_0 = 0$ and $a_1 = 1$, we had arbitrary values for them (the signal space representations of $s_0(t)$ and $s_1(t)$), we could have derived the result in the last section the same way. As long as $a_0 < a_1$, we'd still have $r \underset{H_0}{\overset{H_1}{\gtrless}} \gamma$, but now,

$$\gamma = \frac{a_0 + a_1}{2} + \frac{\sigma_N^2}{a_1 - a_0} \log \frac{P[H_0]}{P[H_1]} \tag{23}$$

14.7 Equi-probable Special Case

If symbols are equally likely, $P[H_1] = P[H_0]$, then $\frac{P[H_1]}{P[H_0]} = 1$ and the logarithm of the fraction is zero. So then

$$x \underset{H_0}{\overset{H_1}{\gtrless}} \frac{a_0 + a_1}{2}$$

The decision above says that if x is closer to a_0, decide that $s_0(t)$ was sent. And if x is closer to a_1, decide that $s_1(t)$ was sent. The boundary is exactly half-way in between the two signal space vectors.

This receiver is also called a maximum likelihood detector, because we only decide which likelihood function is higher (neither is scaled by the prior probabilities $P[H_0]$ or $P[H_1]$.

14.8 Examples

Example: When H_1 becomes less likely, which direction will the optimal threshold move, towards a_0 or towards a_1?

Solution: Towards a_1.

Example: Let $a_0 = -1$, $a_1 = 1$, $\sigma_N^2 = 0.1$, $P[H_1] = 0.4$, and $P[H_0] = 0.6$. What is the decision threshold for x?

Solution: From (23),

$$\gamma = 0 + \frac{0.1}{2}\log\frac{0.6}{0.4} = 0.05\log 1.5 \approx 0.0203$$

Example: Can the decision threshold be higher than both a_0 and a_1 in this binary, one-dimensional signalling, receiver?

Given a_0, a_1, σ_N^2, $P[H_1]$, and $P[H_0]$, you should be able to calculate the optimal decision threshold γ.

Example: In this example, given all of the above constants and the optimal threshold γ, calculate the probability of error from (18).
Starting from

$$P[\text{error}] = P[x \in R_1|H_0]\,P[H_0] + P[x \in R_0|H_1]\,P[H_1]$$

we can use the decision regions in (22) to write

$$P[\text{error}] = P[x > \gamma|H_0]\,P[H_0] + P[x < \gamma|H_1]\,P[H_1]$$

What is the first probability, given that $r|H_0$ is Gaussian with mean a_0 and variance σ_N^2? What is the second probability, given that $x|H_1$ is Gaussian with mean a_1 and variance σ_N^2? What is then the overall probability of error?

Solution:

$$\begin{aligned}
P[x > \gamma|H_0] &= \mathrm{Q}\left(\frac{\gamma - a_0}{\sigma_N}\right) \\
P[x < \gamma|H_1] &= 1 - \mathrm{Q}\left(\frac{\gamma - a_1}{\sigma_N}\right) = \mathrm{Q}\left(\frac{a_1 - \gamma}{\sigma_N}\right) \\
P[\text{error}] &= P[H_0]\,\mathrm{Q}\left(\frac{\gamma - a_0}{\sigma_N}\right) + P[H_1]\,\mathrm{Q}\left(\frac{a_1 - \gamma}{\sigma_N}\right)
\end{aligned}$$

14.9 Review of Binary Detection

We did three things to prove some things about the optimal detector:

- We wrote the formula for the probability of error.
- We found the decision regions which minimized the probability of error.
- We used the log operator to show that for the Gaussian error case the decision regions are separated by a single threshold.
- We showed the formula for that threshold, both in the equi-probable symbol case, and in the general case.

Lecture 9

Today: (1) Finish Lecture 8 (2) Intro to M-ary PAM

Sample exams (2007-8) and solutions posted on WebCT. Of course, by putting these sample exams up, you know the actual exam won't contain the same exact problems. There is no guarantee this exam will be the same level of difficulty of either past exam.
Notes on 2008 sample exam:

- Problem 1(b) was not material covered this semester.
- Problem 3 is not on the topic of detection theory, it is on the probability topic. There is a typo: $f_N(n)$ should be $f_N(t)$.
- Problem 6 typo: $x_1(t) = \begin{cases} t, & -1 \le t \le 1 \\ 0, & o.w. \end{cases}$ and $x_1(t) = \begin{cases} \frac{1}{2}(3t^2-1), & -1 \le t \le 1 \\ 0, & o.w. \end{cases}$. That is, the "x"s on the RHS of the given expressions should be "t"s.

Notes on 2007 sample exam:

- Problem 2 should have said "with period T_{sa}" rather than "at rate T_{sa}".
- The book that year used $\varphi_i(t)$ instead of $\phi_i(t)$ as the notation for a basis function, and α_i instead of $\mathbf{s}_i$ as the signal space vector for signal $s_i(t)$.
- Problem 6 is on the topic of detection theory.

15 Pulse Amplitude Modulation (PAM)

Def'n: *Pulse Amplitude Modulation (PAM)*
M-ary Pulse Amplitude Modulation is the use of M scaled versions of a single basis function $p(t) = \phi_0(t)$ as a signal set,

$$s_i(t) = a_i p(t), \quad \text{for } i = 0, \ldots, M$$

where a_i is the amplitude of waveform i.

Each sent waveform (a.k.a. "symbol") conveys $k = \log_2 M$ bits of information. Note we're calling it $p(t)$ instead of $\phi_0(t)$, and a_i instead of $a_{i,0}$, both for simplicity, since there's only one basis function.

15.1 Baseband Signal Examples

When you compose a signal of a number of symbols, you'll just add each time-delayed signal to the transmitted waveform. The resulting signal we'll call $s(t)$ (without any subscript) and is

$$s(t) = \sum_n a(n) p(t - nT_s)$$

where $a(n) \in \{a_0, \ldots, a_{M-1}\}$ is the nth symbol transmitted, and T_s is the symbol period (not the sample period!). Instead of just sending one symbol, we are sending one symbol every T_s units of time. That makes our *symbol rate* as $1/T_s$ symbols per second.

Note: Keep in mind that when $s(t)$ is to be transmitted on a bandpass channel, it is modulated with a carrier $\cos(2\pi f_c t)$,

$$x(t) = \Re\left\{ s(t) e^{j2\pi f_c t} \right\}$$

Rice Figures 5.2.3 and 5.2.6 show continuous-time and discrete-time realizations, respectively, of PAM transmitters and receivers.

Example: Binary Bipolar PAM
Figure 22 shows a 4-ary PAM signal set using amplitudes $a_0 = -A$, $a_1 = A$. It shows a signal set using square pulses,

$$\phi_0(t) = p(t) = \begin{cases} 1/\sqrt{T_s}, & 0 \leq t < T_s \\ 0, & o.w. \end{cases}$$

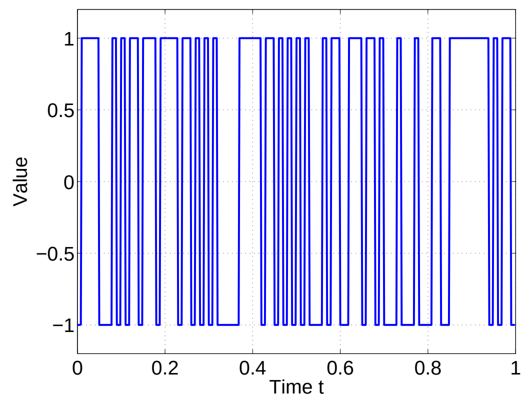


Figure 22: Example signal for binary bipolar PAM example for $A = \sqrt{T_s}$.

Example: Binary Unipolar PAM
Unipolar binary PAM uses the amplitudes $a_0 = 0$, $a_1 = A$. The average energy per symbol has decreased. This is also called *On-Off Keying* (OOK). If the y-axis in Figure 22 was scaled such that the minimum was 0 instead of -1, it would represent unipolar PAM.

Example: 4-ary PAM
A 4-ary PAM signal set using amplitudes $a_0 = -3A$, $a_1 = -A$, $a_2 = A$, $a_3 = 3A$ is shown in Figure 23. It shows a signal set using square pulses,

$$p(t) = \begin{cases} 1/\sqrt{T_s}, & 0 \leq t < T_s \\ 0, & o.w. \end{cases}$$

Typical M-ary PAM (for all even M) uses the following amplitudes:

$$-(M-1)A, -(M-3)A, \ldots, -A, A, \ldots, +(M-3)A, +(M-1)A$$

Rice Figure 5.2.1 shows the signal space diagram of M-ary PAM for different values of M.

15.2 Average Bit Energy in M-ary PAM

Assuming that each symbol $s_i(t)$ is equally likely to be sent, we want to calculate the average bit energy $\mathcal{E}_b$, which is $1/\log_2 M$ times the symbol energy $\mathcal{E}_s$,

$$\begin{aligned} \mathcal{E}_s &= E\left[\int_{-\infty}^{\infty} a_i^2 p^2(t) dt\right] = \int_{-\infty}^{\infty} E\left[a_i^2\right] \phi_0^2(t) dt \\ &= E\left[a_i^2\right] \end{aligned} \quad (24)$$

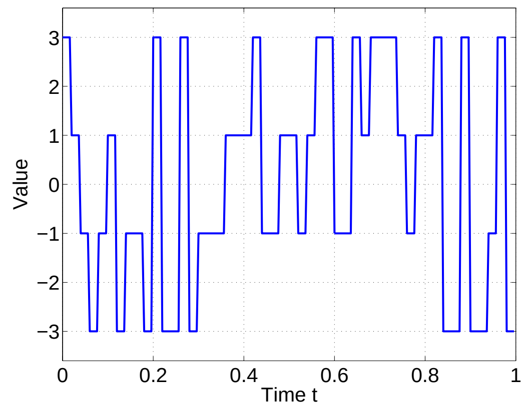


Figure 23: Example signal for 4-ary PAM example.

Let $\{a_0, \ldots, a_{M-1}\} = -(M-1)A, -(M-3)A, \ldots, (M-1)A$, as described above to be the typical M-ary PAM case. Then

$$\begin{aligned} E\left[a_i^2\right] &= \frac{A^2}{M}\left[(1-M)^2 + (3-M)^2 + \cdots + (M-1)^2\right] \\ &= \frac{A^2}{M}\sum_{m=1}^{M}(2m-1-M)^2 = \frac{A^2}{M}\frac{M(M^2-1)}{3} \\ &= A^2\frac{(M^2-1)}{3} \end{aligned}$$

So $\mathcal{E}_s = \frac{(M^2-1)}{3}A^2$, and

$$\mathcal{E}_b = \frac{1}{\log_2 M}\frac{(M^2-1)}{3}A^2$$

Lecture 10

Today: Review for Exam 1

16 Topics for Exam 1

1. Fourier transform of signals, properties (12)
2. Bandpass signals
3. Sampling and aliasing
4. Orthogonality / Orthonormality
5. Correlation and Covariance
6. Signal space representation
7. Correlation / matched filter receivers

Make sure you know how to do each HW problem in HWs 1-3. It will be good to know the 'tricks' of how each problem is done, since they will reappear.

Read over the lecture notes 1-7. If anything is confusing, ask me.

You may have one side of an 8.5×11 sheet of paper for notes. You will be provided with Table 2.4.4 and Table 2.4.8.

17 Additional Problems

17.1 Spectrum of Communication Signals

1. Bateman Problem 1.9: An otherwise ideal mixer (multiplier) generates an internal DC offset which sums with the baseband signal prior to multiplication with the $\cos(2\pi f_c t)$ carrier. How does this affect the spectrum of the output signal?

2. Haykin & Moyer Problem 2.25. A signal $x(t)$ of finite energy is applied to a square law device whose output $y(t)$ is defined by $y(t) = x^2(t)$. The spectrum of $x(t)$ is bandlimited to $-W \leq \omega \leq W$. Prove that the spectrum of $y(t)$ is bandlimited to $-2W \leq \omega \leq 2W$.

3. Consider the pulse shape
$$x(t) = \begin{cases} \cos\left(\frac{\pi t}{T_s}\right), & -\frac{T_s}{2} < t < \frac{T_s}{2} \\ 0, & o.w. \end{cases}$$
   (a) Draw a plot of the pulse shape $x(t)$.
   (b) Find the Fourier transform of $x(t)$.

4. Rice 2.39, 2.52

17.2 Sampling and Aliasing

1. Assume that $x(t)$ is a bandlimited signal with $X(f) = 0$ for $|f| > W$. When $x(t)$ is sampled at a rate $T = \frac{1}{2W}$, its samples X_n are,
$$X_n = \begin{cases} 1, & n = \pm 1 \\ 2, & n = 0 \\ 0, & o.w. \end{cases}$$
   What was the original continuous-time signal $x(t)$? Or, if $x(t)$ cannot be determined, why not?

2. Rice 2.99, 2.100

17.3 Orthogonality and Signal Space

1. (from L.W. Couch 2007, Problem 2-49). Three functions are shown in Figure P2-49.
   (a) Show that these functions are orthogonal over the interval $(-4, 4)$.
   (b) Find the scale factors needed to scale the functions to make them into an orthonormal set over the interval $(-4, 4)$.
   (c) Express the waveform
$$w(t) = \begin{cases} 1, & 0 \leq t \leq 4 \\ 0, & o.w. \end{cases}$$
   in signal space using the orthonormal set found in part (b).

2. Rice Example 5.1.1 on Page 219.

3. Rice 5.2, 5.5, 5.13, 5.14, 5.16, 5.23, 5.26

17.4 Random Processes, PSD

1. (From Couch 2007 Problem 2-80) An RC low-pass filter is the circuit shown in Figure 24. Its transfer function is

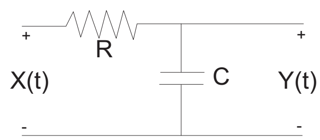


Figure 24: An RC low-pass filter.

$$H(f) = \frac{Y(f)}{X(f)} = \frac{1}{1 + j2\pi \mathrm{RC} f}$$

Given that the PSD of the input signal is flat and constant at 1, *i.e.*, $S_X(f) = 1$, design an RC low-pass filter that will attenuate this signal by 20 dB at 15 kHz. That is, find the value of RC to satisfy the design specifications.

2. Let a binary 1-D communication system be described by two possible signals, and noise with different variance depending on which signal is sent, *i.e.*,

$$\begin{aligned} H_0: &\quad r = a_0 + n_0 \\ H_1: &\quad r = a_1 + n_1 \end{aligned}$$

where $\alpha_0 = -1$ and $\alpha_1 = 1$, and n_0 is zero-mean Gaussian with variance 0.1, and n_1 is zero-mean Gaussian with variance 0.3.

(a) What is the conditional pdf of r given H_0?

(b) What is the conditional pdf of r given H_1?

17.5 Correlation / Matched Filter Receivers

1. A binary communication system uses the following equally-likely signals,

$$\begin{aligned} s_1(t) &= \begin{cases} \cos\left(\frac{\pi t}{T}\right), & -\frac{T}{2} \leq t \leq \frac{T}{2} \\ 0, & o.w. \end{cases} \\ s_2(t) &= \begin{cases} -\cos\left(\frac{\pi t}{T}\right), & -\frac{T}{2} \leq t \leq \frac{T}{2} \\ 0, & o.w. \end{cases} \end{aligned}$$

At the receiver, a signal $x(t) = s_i(t) + n(t)$ is received.

(a) Is $s_1(t)$ an energy-type or power-type signal?

(b) What is the energy or power (depending on answer to (a)) in $s_1(t)$?

(c) Describe in words and a block diagram the operation of an correlation receiver.

2. Proakis & Salehi 7.8. Suppose that two signal waveforms $s_1(t)$ and $s_2(t)$ are orthogonal over the interval $(0, T)$. A sample function $n(t)$ of a zero-mean, white noise process is correlated with $s_1(t)$ and $_2(t)$ to yield,

$$n_1 = \int_0^T s_1(t)n(t)dt$$
$$n_2 = \int_0^T s_2(t)n(t)dt$$

   Prove that $E[n_1 n_2] = 0$.

3. Proakis & Salehi 7.16. Prove that when a sinc pulse $g_T(t) = \sin(\pi t/T)/(\pi t/T)$ is passed through its matched filter, the output is the same sinc pulse.

Lecture 11

Today: (1) M-ary PAM Intro from Lecture 9 notes (2) Probability of Error in PAM

18 Probability of Error in Binary PAM

This is in Section 6.1.2 of Rice.

How are $N_0/2$ and σ_N related? Recall from lecture 7 that the variance of n_k came out to be $N_0/2$. We had also called the variance of noise component n_k as σ_N^2. So:

$$\sigma_N^2 = \frac{N_0}{2}. \tag{25}$$

18.1 Signal Distance

We mentioned, when talking about signal space diagrams, a distance between vectors,

$$d_{i,j} = \|\mathbf{a}_i - \mathbf{a}_j\| = \left[\sum_{k=1}^{M}(a_{i,k} - a_{j,k})^2\right]^{1/2}$$

For the 1-D, two signal case, there is only $d_{1,0}$,

$$d_{1,0} = |a_1 - a_0| \tag{26}$$

18.2 BER Function of Distance, Noise PSD

We have consistently used $a_1 > a_0$. In the Gaussian noise case (with equal variance of noise in H_0 and H_1), and with $P[H_0] = P[H_1]$, we came up with threshold $\gamma = (a_0 + a_1)/2$, and

$$P[\text{error}] = P[x > \gamma | H_0] P[H_0] + P[x < \gamma | H_1] P[H_1]$$

Which simplified using the Q function,

$$\begin{aligned} P[x > \gamma | H_0] &= \text{Q}\left(\frac{\gamma - a_0}{\sigma_N}\right) \\ P[x < \gamma | H_1] &= 1 - \text{Q}\left(\frac{\gamma - a_1}{\sigma_N}\right) = \text{Q}\left(\frac{a_1 - \gamma}{\sigma_N}\right) \end{aligned} \tag{27}$$

Thus

$$\begin{aligned} P[x > \gamma | H_0] &= \text{Q}\left(\frac{(a_1 - a_0)/2}{\sigma_N}\right) \\ P[x < \gamma | H_1] &= \text{Q}\left(\frac{(a_1 - a_0)/2}{\sigma_N}\right) \end{aligned} \tag{28}$$

So both are equal, and again with $P[H_0] = P[H_1] = 1/2$,

$$P[\text{error}] = \text{Q}\left(\frac{(a_1 - a_0)/2}{\sigma_N}\right) \tag{29}$$

We'd assumed $a_1 > a_0$, so if we had also calculated for the case $a_1 < a_0$, we'd have seen that in general, the following formula works:

$$P[\text{error}] = \text{Q}\left(\frac{|a_1 - a_0|}{2\sigma_N}\right)$$

Then, using (25) and (26), we have

$$P[\text{error}] = \text{Q}\left(\frac{d_{0,1}}{2\sqrt{N_0/2}}\right) = \text{Q}\left(\sqrt{\frac{d_{0,1}^2}{2N_0}}\right) \tag{30}$$

This will be important as we talk about binary PAM. This expression,

$$\text{Q}\left(\sqrt{\frac{d_{0,1}^2}{2N_0}}\right)$$

Is one that we will see over and over again in this class.

18.3 Binary PAM Error Probabilities

For binary PAM ($M = 2$) it takes one symbol to encode one bit. Thus 'bit' and 'symbol' are interchangeable.

In signal space, our signals $s_0(t) = a_0 p(t)$ and $s_1(t) = a_1 p(t)$ are just $s_0 = a_0$ and $s_1 = a_1$. Now we can use (30) to compute the probability of error.

For bipolar signaling, when $A_0 = -A$ and $A_1 = A$,

$$P[\text{error}] = \text{Q}\left(\sqrt{\frac{4A^2}{2N_0}}\right) = \text{Q}\left(\sqrt{\frac{2A^2}{N_0}}\right)$$

For unipolar signaling, when $A_0 = 0$ and $A_1 = A$,

$$P[\text{error}] = \text{Q}\left(\sqrt{\frac{A^2}{2N_0}}\right)$$

However, this doesn't take into consideration that the unipolar signaling method uses only half of the energy to transmit. Since half of the bits are exactly zero, they take no signal energy. Using (24), what is the average energy of bipolar and unipolar signaling?

Solution: For the bipolar signalling, $A_m^2 = A^2$ always, so $\mathcal{E}_b = \mathcal{E}_s = A^2$. For the unipolar signalling, A_m^2 equals A^2 with probability 1/2 and zero with probability 1/2. Thus $\mathcal{E}_b = \mathcal{E}_s = \frac{1}{2}A^2$.

Now, we re-write the probability of error expressions in terms of $\mathcal{E}_b$. For bipolar signaling,

$$P\,[\text{error}] = \mathrm{Q}\left(\sqrt{\frac{4A^2}{2N_0}}\right) = \mathrm{Q}\left(\sqrt{\frac{2\mathcal{E}_b}{N_0}}\right)$$

For unipolar signaling,

$$P\,[\text{error}] = \mathrm{Q}\left(\sqrt{\frac{2\mathcal{E}_b}{2N_0}}\right) = \mathrm{Q}\left(\sqrt{\frac{\mathcal{E}_b}{N_0}}\right)$$

Discussion These probabilities of error are shown in Figure 25.

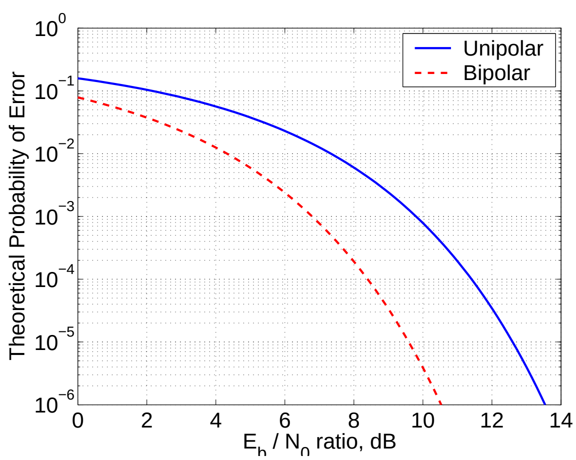


Figure 25: Probability of Error in binary PAM signalling.

- Even for the same bit energy $\mathcal{E}_b$, the bit error rate of bipolar PAM beats unipolar PAM.
- What is the difference in $\mathcal{E}_b/N_0$ for a constant probability of error?
- What is the difference in terms of probability of error for a given $\mathcal{E}_b/N_0$?

Lecture 12

Today: (0) Exam 1 Return, (1) Multi-Hypothesis Detection Theory, (2) Probability of Error in M-ary PAM

19 Detection with Multiple Symbols

When we only had $s_0(t)$ and $s_1(t)$, we saw that our decision was based on which of the following was higher:

$$f_{X|H_0}(x|H_0)P[H_0] \quad \text{and} \quad f_{X|H_1}(x|H_1)P[H_1]$$

For M-ary signals, we'll see that we need to consider the highest of all M possible events H_m,

$$\begin{array}{rl} H_0: & r(t) = s_0(t) + n(t) \\ H_1: & r(t) = s_1(t) + n(t) \\ \cdots & \cdots \\ H_{M-1}: & r(t) = s_M(t) + n(t) \end{array}$$

which have joint probabilities,

$$\begin{array}{rl} H_0: & f_{X|H_0}(x|H_0)P[H_0] \\ H_1: & f_{X|H_1}(x|H_1)P[H_1] \\ \cdots & \cdots \\ H_{M-1}: & f_{X|H_{M-1}}(x|H_{M-1})P[H_{M-1}] \end{array}$$

We will find which of these joint probabilities is highest. For this class, we'll only consider the case of equally probable signals. (While equi-probable signals is sometimes not the case for $M = 2$ binary detection, it is very rare in higher M communication systems.) If $P[H_0] = \cdots = P[H_{M-1}]$ then we only need to find the i that makes the likelihood $f_{X|H_i}(x|H_i)$ maximum, that is, maximum likelihood detection.

$$\text{Symbol Decision} = \arg\max_i f_{X|H_i}(x|H_i)$$

For Gaussian (conditional) r.v.s with equal variances σ_N^2, it is better to maximize the log of the likelihood rather than the likelihood directly, so

$$\log f_{X|H_i}(x|H_i) = -\frac{1}{2}\log(2\pi\sigma_N^2) - \frac{(x-a_i)^2}{2\sigma_N^2}$$

This is maximized when $(x - a_i)^2$ is minimized. Essentially, this is a (squared) distance between x and a_i. So, the decision is, find the a_i which is closest to x.

The decision between two neighboring signal vectors a_i and a_{i+1} will be

$$r \underset{H_i}{\overset{H_{i+1}}{\gtrless}} \gamma_{i,i+1} = \frac{a_i + a_{i+1}}{2}$$

As an example: 4-ary PAM. See Figure 26.

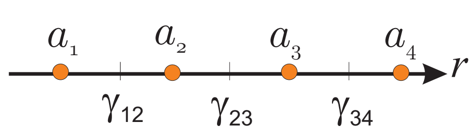


Figure 26: Signal space representation and decision thresholds for 4-ary PAM.

20 M-ary PAM Probability of Error

20.1 Symbol Error

The probability that we don't get the symbol correct is the probability that x does not fall within the range between the thresholds in which it belongs. Here, each $a_i = A_i$. Also the noise $\sigma_N^2 = N_0/2$, as discussed above.

Assuming neighboring symbols a_i are spaced by $2A$, the decision threshold is always A away from the symbol values. For the symbols i in the 'middle',

$$P(\text{symbol error}|H_i) = 2\text{Q}\left(\frac{A}{\sqrt{N_0/2}}\right)$$

For the symbols i on the 'sides',

$$P(\text{symbol error}|H_i) = \text{Q}\left(\frac{A}{\sqrt{N_0/2}}\right)$$

So overall,

$$P(\text{symbol error}) = \frac{2(M-1)}{M}\text{Q}\left(\frac{A}{\sqrt{N_0/2}}\right)$$

20.1.1 Symbol Error Rate and Average Bit Energy

How does this relate to the average bit energy $\mathcal{E}_b$? From last lecture, $\mathcal{E}_b = \frac{1}{\log_2 M}\frac{(M^2-1)}{3}A^2$, which means that

$$A = \sqrt{\frac{3\log_2 M}{M^2-1}\mathcal{E}_b}$$

So

$$P(\text{symbol error}) = \frac{2(M-1)}{M}\text{Q}\left(\sqrt{\frac{6\log_2 M}{M^2-1}\frac{\mathcal{E}_b}{N_0}}\right) \tag{31}$$

Equation (31) is plotted in Figure 27.

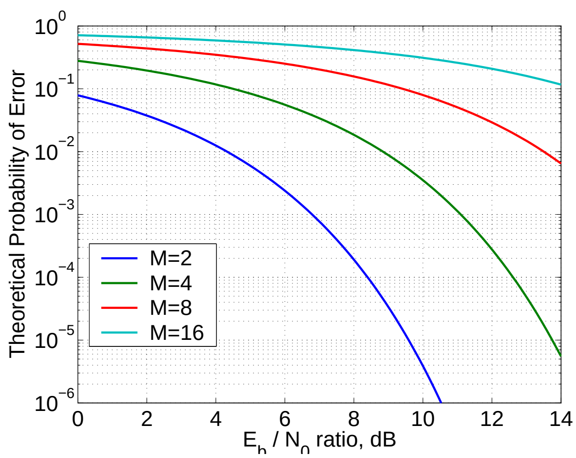


Figure 27: Probability of Symbol Error in M-ary PAM.

20.2 Bit Errors and Gray Encoding

For binary PAM, there are only two symbols, one will be assigned binary 0 and the other binary 1. When you make one symbol error (decide H_0 or H_1 in error) then it will cause one bit error.

For M-ary PAM, bits and symbols are not synonymous. Instead, we have to carefully assign bit codes to symbols $1 \ldots M$.

Example: Bit coding of $M = 4$ symbols
While the two options shown in Figure 28 both assign 2-bits to each symbol in unique ways, one will lead to a higher bit error rate than the other. Is one is better or worse?

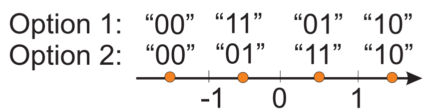


Figure 28: Two options for assigning bits to symbols.

The key is to recall the model for noise. It will not shift a signal uniformly across symbols. It will tend to leave the received signal x close to the original signal a_i. The neighbors of a_i will be more likely than distant signal space vectors.

Thus Gray encoding will only change one bit across boundaries, as in Option 2 in Figure 28.

Example: Bit coding of $M = 8$ symbols
Assign three bits to each symbol such that any two nearest neighbors are different in only one bit (Gray encoding).

Solution: Here is one solution.

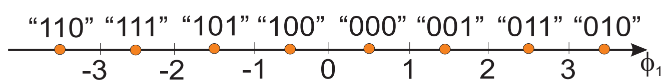


Figure 29: Gray encoding for 8-ary PAM.

20.2.1 Bit Error Probabilities

How many bit errors are caused by a symbol error in M-ary PAM?

- One. If Gray encoding is used, the errors will tend to be just one bit flipped, more than multiple bits flipped. At least at high E_b/N_0,

$$P(\text{error}) \approx \frac{1}{\log_2 M} P(\text{symbol error}) \tag{32}$$

- Maybe more, up to $\log_2 M$ in the worst case. Then, we need to study further the probability that x will jump more than one decision region.

As of now - if you have to estimate a bit error rate for M-ary PAM - use (32). We will show that this approximation is very good, in almost all cases. We will also show examples in multi-dimensional signalling (*e.g.*, QAM) when this is not a good approximation.

Lecture 13

Today: (1) Pulse Shaping / ISI, (2) N-Dim Detection Theory

21 Inter-symbol Interference

1. Consider the spectrum of the ideal 1-D PAM system with a square pulse.
2. Consider the time-domain of the signal which is close to a rect function in the frequency domain.

We don't want either: (1) occupies too much spectrum, and (2) occupies too much time.

1. If we try (1) above and use FDMA (frequency division multiple access) then the interference is *out-of-band interference*.
2. If we try (2) above and put symbols right next to each other in time, our own symbols experience interference called *inter-symbol interference*.

In reality, we want to compromise between (1) and (2) and experience only a small amount of both.

Now, consider the effect of filtering in the path between transmitter and receiver. Filters will be seen in

1. Transmitter: Baseband filters come from the limited bandwidth, speed of digital circuitry. RF filters are used to limit the spectral output of the signal (to meet out-of-band interference requirements).
2. Channel: The wideband radio channel is a sum of time-delayed impulses. Wired channels have (non-ideal) bandwidth – the attenuation of a cable or twisted pair is a function of frequency. Waveguides have bandwidth limits (and modes).
3. Receiver: Bandpass filters are used to reduce interference, noise power entering receiver. Note that the 'matched filter' receiver is also a filter!

21.1 Multipath Radio Channel

(Background knowledge). The measured radio channel is a filter, call its impulse response $h(\tau, t)$. If $s(t)$ is transmitted, then

$$r(t) = s(t) \star h(\tau, t)$$

will be received. Typically, the radio channel is modeled as

$$h(\tau, t) = \sum_{l=1}^{L} \alpha_l e^{j\phi_l} \delta(\tau - \tau_l), \tag{33}$$

where α_l and ϕ_l are the amplitude and phase of the lth multipath component, and τ_l is its time delay. Essentially, each reflection, diffraction, or scattering adds an additional impulse to (33) with a particular time delay (depending on the extra path length compared to the line-of-sight path) and complex amplitude (depending on the losses and phase shifts experienced along the path). The amplitudes of the impulses tend to decay over time, but multipath with delays of hundreds of nanoseconds often exist in indoor links and delays of several microseconds often exist in outdoor links.

The channel in (33) has infinite bandwidth (why?) so isn't really what we'd see if we looked just at a narrow band of the channel.

Example: 2400-2480 MHz Measured Radio Channel
The 'Delay Profile' is shown in Figure 30 for an example radio channel. Actually what we observe

$$\begin{aligned}
\text{PDP}(t) &= r(t) \star x(t) = (x(t) \star h(t)) \star x(t) \\
\text{PDP}(t) &= r(t) \star x(t) = R_X(t) \star h(t) \\
\text{PDP}(f) &= H(f) S_X(f)
\end{aligned} \quad (34)$$

Note that 200 ns is 0.2μs is 1/ 5 MHz. At this symbol rate, each symbol would fully bleed into the next symbol.

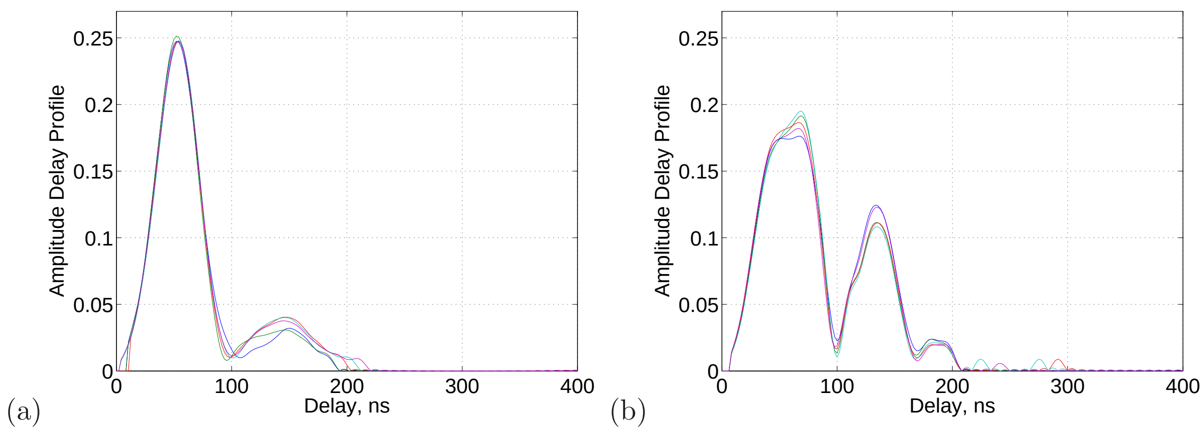


Figure 30: Multiple delay profiles measured on two example links: (a) 13 to 43, and (b) 14 to 43.

How about for a 5 kHz symbol rate? Why or why not?

Other solutions:

- OFDM (Orthogonal Frequency Division Multiplexing)
- Direct Sequence Spread Spectrum / CDMA
- Adaptive Equalization

21.2 Transmitter and Receiver Filters

Example: Bipolar PAM Signal Input to a Filter

How does this effect the correlation receiver and detector? Symbols sent over time are no longer orthogonal – one symbol 'leaks' into the next (or beyond).

22 Nyquist Filtering

Key insight:

- We don't need the leakage to be zero at all time.

- The value out of the matched filter (correlator) is taken only at times nT_s, for integer n.

Nyquist filtering has the objective of, at the matched filter output, making the sum of 'leakage' or ISI from one symbol be exactly 0 at **all** other sampling times nT_s.

Where have you seen this condition before? This condition, that there is an inner product of zero, is the condition for orthogonality of two waveforms. What we need are pulse shapes (waveforms), that when shifted in time by T_s, are orthogonal to each other. In more mathematical language, we need $p(t)$ such that

$$\{\phi_n(t) = p(t - nT_s)\}_{n=\ldots,-1,0,1,\ldots}$$

forms an orthonormal basis. The square root raised cosine, shown in Figure 32 accomplishes this. But lots of other functions also accomplish this. The Nyquist filtering theorem provides an easy means to come up with others.

Theorem: Nyquist Filtering
Proof: A necessary and sufficient condition for $x(t)$ to satisfy

$$x(nT_s) = \begin{cases} 1, & n = 0 \\ 0, & o.w. \end{cases}$$

is that its Fourier transform $X(f)$ satisfy

$$\sum_{m=-\infty}^{\infty} X\left(f + \frac{m}{T_s}\right) = T_s$$

Proof: on page 833, Appendix A, of Rice book.

Basically, $X(f)$ could be:

- $X(f) = \text{rect}(fT_s)$, *i.e.*, exactly constant within $-\frac{1}{2T_s} < f < \frac{1}{2T_s}$ band and zero outside.
- It may bleed into the next 'channel' but the sum of

$$\cdots + X\left(f - \frac{1}{T_s}\right) + X(f) + X\left(f + \frac{1}{T_s}\right) + \cdots$$

  must be constant across all f.

Thus $X(f)$ is allowed to bleed into other frequency bands – but the neighboring frequency-shifted copy of $X(f)$ must be lower s.t. the sum is constant. See Figure 31.

If $X(f)$ only bleeds into one neighboring channel (that is, $X(f) = 0$ for all $|f| > \frac{1}{T_s}$), we denote the difference between the ideal rect function and our $X(f)$ as $\Delta(f)$,

$$\Delta(f) = |X(f) - \text{rect}(fT_s)|$$

then we can rewrite the Nyquist Filtering condition as,

$$\Delta\left(\frac{1}{2T_s} - f\right) = \Delta\left(\frac{1}{2T_s} + f\right)$$

for $-1/T_s \le f < 1/T_s$. Essentially, symmetric about $f = 1/(2T_s)$. See Figure 31

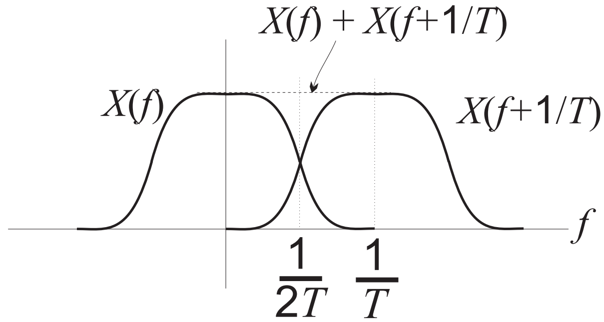


Figure 31: The Nyquist filtering criterion – $1/T_s$-frequency-shifted copies of $X(f)$ must add up to a constant.

Bateman presents this whole condition as "A Nyquist channel response is characterized by the transfer function having a transition band that is symmetrical about a frequency equal to $0.5 \times 1/T_s$."

Activity: Do-it-yourself Nyquist filter. Take a sheet of paper and fold it in half, and then in half the other direction. Cut a function in the thickest side (the edge that you just folded). Leave a tiny bit so that it is not completely disconnected into two halves. Unfold. Drawing a horizontal line for the frequency f axis, the middle is $0.5/T_s$, and the vertical axis it $X(f)$.

22.1 Raised Cosine Filtering

$$H_{RC}(f) = \begin{cases} T_s, & 0 \le |f| \le \frac{1-\alpha}{2T_s} \\ \frac{T_s}{2}\left\{1 + \cos\left[\frac{\pi T_s}{\alpha}\left(|f| - \frac{1-\alpha}{2T_s}\right)\right]\right\}, & \frac{1-\alpha}{2T_s} \le |f| \le \frac{1+\alpha}{2T_s} \\ 0, & o.w. \end{cases} \quad (35)$$

22.2 Square-Root Raised Cosine Filtering

But we usually need do design a system with two identical filters, one at the transmitter and one at the receiver, which in series, produce a zero-ISI filter. In other words, we need $|H(f)|^2$ to meet the Nyquist filtering condition.

$$H_{RRC}(f) = \begin{cases} \sqrt{T_s}, & 0 \le |f| \le \frac{1-\alpha}{2T_s} \\ \sqrt{\frac{T_s}{2}\left\{1 + \cos\left[\frac{\pi T_s}{\alpha}\left(|f| - \frac{1-\alpha}{2T_s}\right)\right]\right\}}, & \frac{1-\alpha}{2T_s} \le |f| \le \frac{1+\alpha}{2T_s} \\ 0, & o.w. \end{cases} \quad (36)$$

23 M-ary Detection Theory in N-dimensional signal space

We are going to start to talk about QAM, a modulation with two dimensional signal vectors, and later even higher dimensional signal vectors. We have developed M-ary detection theory for 1-D signal vectors, and now we will extend that to N-dimensional signal vectors.

Setup:

- Transmit: one of M possible symbols, $\mathbf{a}_0, \ldots, \mathbf{a}_{M-1}$.

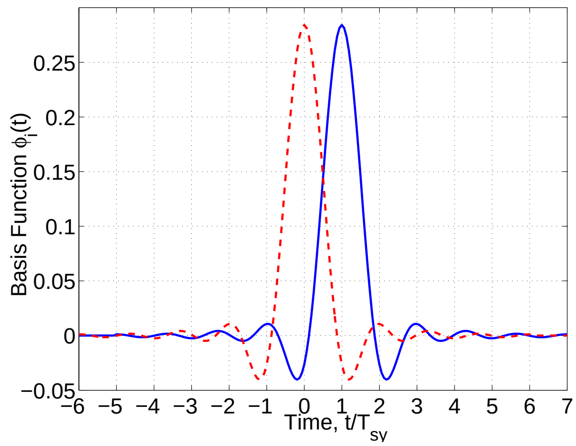


Figure 32: Two SRRC Pulses, delayed in time by nT_s for any integer n, are orthogonal to each other.

- Receive: the symbol plus noise:

$$\begin{array}{rl} H_0 : & \mathbf{X} = \mathbf{a}_0 + \mathbf{n} \\ \cdots & \cdots \\ H_{M-1} : & \mathbf{X} = \mathbf{a}_{M-1} + \mathbf{n} \end{array}$$

- Assume: $\mathbf{n}$ is multivariate Gaussian, each component n_i is independent with zero mean and variance $\sigma_N = N_0/2$.
- Assume: Symbols are equally likely.
- Question: What are the optimal decision regions?

When symbols are equally likely, the optimal decision turns out to be given by the maximum likelihood receiver,

$$\hat{i} = \operatorname*{argmax}_i \log f_{\mathbf{X}|H_i}(\mathbf{x}|H_i) \tag{37}$$

Here,

$$f_{\mathbf{X}|H_i}(\mathbf{x}|H_i) = \frac{1}{(2\pi\sigma_N^2)^{N/2}} \exp\left[-\frac{\sum_j (x_j - a_{i,j})^2}{2\sigma_N^2}\right]$$

which can also be written as

$$f_{\mathbf{X}|H_i}(\mathbf{x}|H_i) = \frac{1}{(2\pi\sigma_N^2)^{N/2}} \exp\left[-\frac{\|\mathbf{x} - \mathbf{a}_i\|^2}{2\sigma_N^2}\right]$$

So when we want to solve (37) we can simplify quickly to:

$$\begin{aligned} \hat{i} &= \operatorname*{argmax}_i \left\{ \log \frac{1}{(2\pi\sigma_N^2)^{N/2}} - \frac{\|\mathbf{x} - \mathbf{a}_i\|^2}{2\sigma_N^2} \right\} \\ \hat{i} &= \operatorname*{argmax}_i - \frac{\|\mathbf{x} - \mathbf{a}_i\|^2}{2\sigma_N^2} \\ \hat{i} &= \operatorname*{argmin}_i \|\mathbf{x} - \mathbf{a}_i\|^2 \\ \hat{i} &= \operatorname*{argmin}_i \|\mathbf{x} - \mathbf{a}_i\| \end{aligned} \tag{38}$$

Again: Just find the $\mathbf{a}_i$ in the signal space diagram which is closest to $\mathbf{x}$.

Pairwise Comparisons When is $\mathbf{x}$ closer to $\mathbf{a}_i$ than to $\mathbf{a}_j$ for some other signal space point j? Solution: **The two decision regions are separated by a straight line** (Note: replace "line" with plane in 3-D, or subspace in N-D). To find this line:

1. Draw a line segment connecting $\mathbf{a}_i$ and $\mathbf{a}_j$.
2. Draw a point in the middle of that line segment.
3. Draw the perpendicular bisector of the line segment through that point.

Why?

Solution: Try to find the locus of point $\mathbf{x}$ which satisfy

$$\|\mathbf{x} - \mathbf{a}_i\|^2 = \|\mathbf{x} - \mathbf{a}_j\|^2$$

You can do this by using the inner product to represent the magnitude squared operator:

$$(\mathbf{x} - \mathbf{a}_i)^T(\mathbf{x} - \mathbf{a}_i) = (\mathbf{x} - \mathbf{a}_j)^T(\mathbf{x} - \mathbf{a}_j)$$

Then use FOIL (multiply out), cancel, and reorganize to find a linear equation in terms of $\mathbf{x}$. This is left as an exercise.

Decision Regions Each pairwise comparison results in a linear division of space. The combined decision region of R_i is the space which is the intersection of all pair-wise decision regions. (All conditions must be satisfied.)

Example: Optimal Decision Regions
See Figure 33.

Lecture 14

Today: (1) QAM & PSK, (2) QAM Probability of Error

24 Quadrature Amplitude Modulation (QAM)

Quadrature Amplitude Modulation (QAM) is a two-dimensional signalling method which uses the in-phase and quadrature (cosine and sine waves, respectively) as the two dimensions. Thus QAM uses two basis functions. These are:

$$\begin{aligned}\phi_0(t) &= \sqrt{2}p(t)\cos(\omega_0 t)\\ \phi_1(t) &= \sqrt{2}p(t)\sin(\omega_0 t)\end{aligned}$$

where $p(t)$ is a pulse shape (like the ones we've looked at previously) with support on $T_1 \leq t \leq T_2$. That is, $p(t)$ is only non-zero within that window.

Previously, we've separated frequency up-conversion and down-conversion from pulse shaping. This definition of the orthonormal basis specifically considers a pulse shape at a frequency ω_0. We include it here because

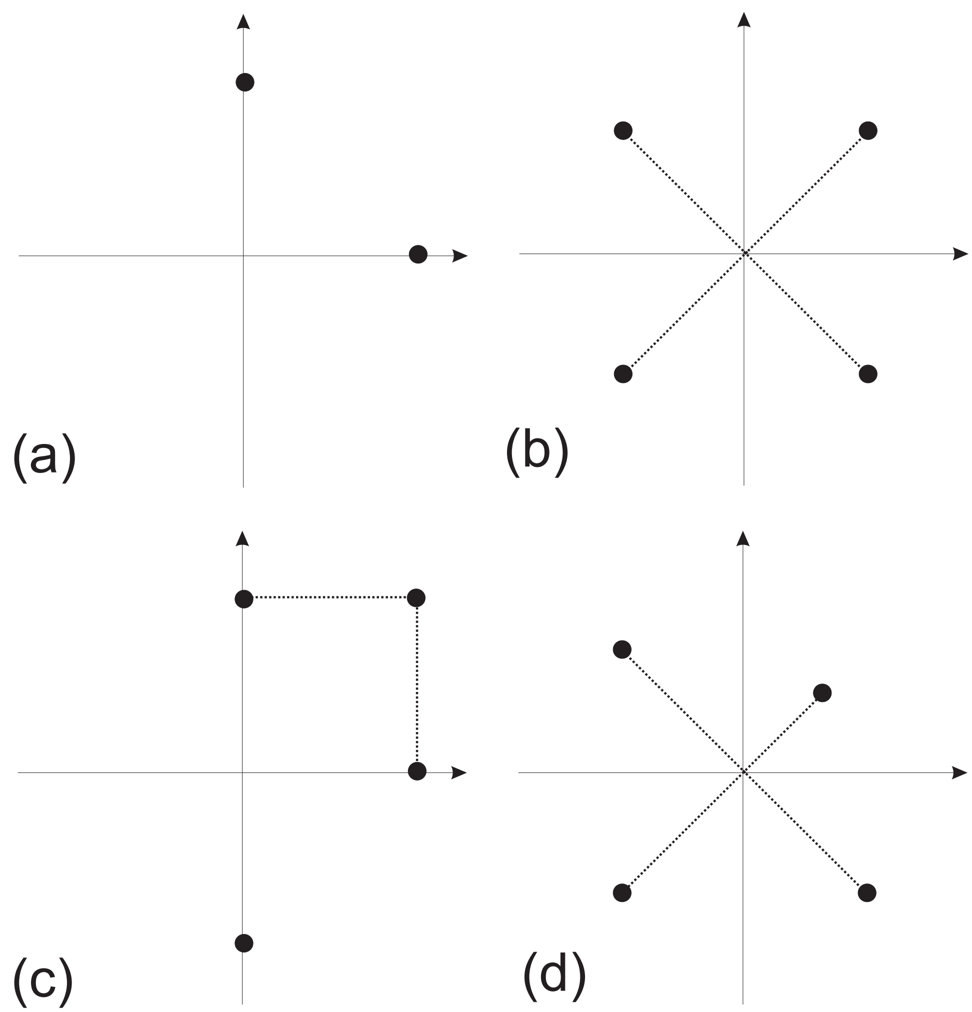


Figure 33: Example signal space diagrams. Draw the optimal decision regions.

it is *critical* to see how, with the same pulse shape $p(t)$, we can have two orthogonal basis functions. (This is not intuitive!)

We have two restrictions on $p(t)$ that makes these two basis functions orthonormal:

- $p(t)$ is unit-energy.
- $p(t)$ is 'low pass'; that is, it has low frequency content compared to $\omega_0 t$.

24.1 Showing Orthogonality

Let's show that the basis functions are orthogonal. We'll need these cosine / sine function relationships:

$$\begin{aligned} \sin(2A) &= 2\cos A \sin A \\ \cos A - \cos B &= -2\sin\left(\frac{A+B}{2}\right)\sin\left(\frac{A-B}{2}\right) \end{aligned}$$

As a first example, let $p(t) = c$ for a constant c , for $T_1 \leq t \leq T_2$. Are the two bases orthogonal? First, show that

$$\langle \phi_0(t), \phi_1(t) \rangle = c^2 \frac{\sin(\omega_0(T_2+T_1))\sin(\omega_0(T_2-T_1))}{\omega_0}$$

Solution: First, see how far we can get before plugging in for $p(t)$:

$$\begin{aligned} \langle \phi_0(t), \phi_1(t) \rangle &= \int_{T_1}^{T_2} \sqrt{2}p(t)\cos(\omega_0 t)\sqrt{2}p(t)\sin(\omega_0 t)dt \\ &= 2\int_{T_1}^{T_2} p^2(t)\cos(\omega_0 t)\sin(\omega_0 t)dt \\ &= \int_{T_1}^{T_2} p^2(t)\sin(2\omega_0 t)dt \end{aligned}$$

Next, consider the constant $p(t) = c$ for a constant c, for $T_1 \leq t \leq T_2$.

$$\begin{aligned} \langle \phi_0(t), \phi_1(t) \rangle &= c^2 \int_{T_1}^{T_2} \sin(2\omega_0 t)dt \\ &= c^2 \left[-\frac{\cos(2\omega_0 t)}{2\omega_0} \right|_{T_1}^{T_2} \\ &= -c^2 \frac{\cos(2\omega_0 T_2) - \cos(2\omega_0 T_1)}{2\omega_0} \\ &= -c^2 \frac{\cos(2\omega_0 T_2) - \cos(2\omega_0 T_1)}{2\omega_0} \end{aligned}$$

I want the answer in terms of $T_2 - T_1$ (for reasons explained below), so

$$\langle \phi_0(t), \phi_1(t) \rangle = c^2 \frac{\sin(\omega_0(T_2+T_1))\sin(\omega_0(T_2-T_1))}{\omega_0}$$

We can see that there are two cases:

1. The pulse duration $T_2 - T_1$ is an integer number of periods. That is, $\omega_0(T_2 - T_1) = \pi k$ for some integer k. In this case, the right sin is zero, and so the correlation is exactly zero.
2. Otherwise, the numerator bounded above and below by +1 and -1, because it is a sinusoid. That is,

$$-\frac{c^2}{\omega_0} \leq \langle \phi_0(t), \phi_1(t) \rangle \leq \frac{c^2}{\omega_0}$$

   Typically, ω_0 is a large number. For example, frequencies $2\pi\omega_0$ could be in the MHz or GHz ranges. Certainly, when we divide by numbers on the order of 10^6 or 10^9, we're going to get a very small inner product. For engineering purposes, $\phi_0(t)$ and $\phi_1(t)$ are orthogonal.

Finally, we can attempt the proof for the case of arbitrary pulse shape $p(t)$. In this case, we use the 'low-pass' assumption that the maximum frequency content of $p(t)$ is much lower than $2\pi\omega_0$. This assumption allows us to assume that $p(t)$ is nearly constant over the course of one cycle of the carrier sinusoid.

This is well-illustrated in Figure 5.15 in the Rice book (page 298). In this figure, we see a pulse modulated by a sine wave at frequency $2\pi\omega_0$. Zooming in on any few cycles, you can see that the pulse $p^2(t)$ is largely constant across each cycle. Thus, when we integrate $p^2(t)\sin(2\omega_0 t)$ across one cycle, we're going to end up with approximately zero. Proof?

Solution: How many cycles are there? Consider the period $[T_1, T_2]$. How many times can it be divided by π/ω_0? Let the integer number be

$$L = \left\lfloor \frac{(T_2 - T_1)\omega_0}{\pi} \right\rfloor.$$

Then each cycle is in the period, $[T_1 + (i-1)\pi/\omega_0, T_1 + i\pi/\omega_0]$, for $i = 1, \ldots, L$. Within each of these cycles, assume $p^2(t)$ is nearly constant. Then, in the same way that the earlier integral was zero, this part of the integral is zero here. The only remainder is the remainder (partial cycle) period, $[T_1 + L\pi/\omega_0, T_2]$. Thus

$$\begin{aligned}
\langle \phi_0(t), \phi_1(t) \rangle &= p^2(T_1 + L\pi/\omega_0) \int_{T_1 + L\pi/\omega_0}^{T_2} \sin(2\omega_0 t) dt \\
&= p^2(T_1 + L\pi/\omega_0) \frac{\cos(2\omega_0 T_2) - \cos(2\omega_0 T_1 + 2\pi L)}{2\omega_0} \\
&= p^2(T_1 + L\pi/\omega_0) \frac{\sin(\omega_0(T_2 + T_1)) \sin(\omega_0(T_2 - T_1))}{\omega_0}
\end{aligned}$$

Again, the inner product is bounded on either side:

$$-\frac{p^2(T_1 + L\pi/\omega_0)}{\omega_0} \leq \langle \phi_0(t), \phi_1(t) \rangle \frac{p^2(T_1 + L\pi/\omega_0)}{\omega_0}$$

which for very large ω_0, is nearly zero.

24.2 Constellation

With these two basis functions, M-ary QAM is defined as an arbitrary signal set $\mathbf{a}_0, \ldots, \mathbf{a}_{M-1}$, where each signal space vector $\mathbf{a}_k$ is two-dimensional:

$$\mathbf{a}_k = [a_{k,0}, a_{k,1}]^T$$

The signal corresponding to symbol k in (M-ary) QAM is thus

$$\begin{aligned} s(t) &= a_{k,0}\phi_0(t) + a_{k,1}\phi_1(t) \\ &= a_{k,0}\sqrt{2}p(t)\cos(\omega_0 t) + a_{k,1}\sqrt{2}p(t)\sin(\omega_0 t) \\ &= \sqrt{2}p(t)\left[a_{k,0}\cos(\omega_0 t) + a_{k,1}\sin(\omega_0 t)\right] \end{aligned}$$

Note that we could also write the signal $s(t)$ as

$$\begin{aligned} s(t) &= \sqrt{2}p(t)\mathbb{R}\left\{a_{k,0}e^{-j\omega_0 t} + j a_{k,1}e^{-j\omega_0 t}\right\} \\ &= \sqrt{2}p(t)\mathbb{R}\left\{e^{-j\omega_0 t}\left(a_{k,0} + j a_{k,1}\right)\right\} \end{aligned} \quad (39)$$

In many textbooks, you will see them write a QAM signal in shorthand as

$$s_{CB}(t) = p(t)(a_{k,0} + j a_{k,1})$$

This is called 'Complex Baseband'. If you do the following operation you can recover the real signal $s(t)$ as

$$s(t) = \sqrt{2}\mathbb{R}\left\{e^{-j\omega_0 t} s_{SB}(t)\right\}$$

You will not be responsible for Complex Baseband notation, but you should be able to read other books and know what they're talking about.

Then, we can see that the signal space representation $\mathbf{a}_k$ is given by

$$\mathbf{a}_k = [a_{k,0}, a_{k,1}]^T$$

for $k = 0, \ldots, M-1$

24.3 Signal Constellations

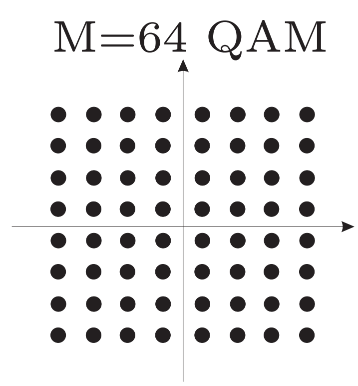


Figure 34: Square signal constellation for 64-QAM.

- See Figure 5.17 for examples of square QAM. These constellations use $M = 2^a$ for some **even** integer a, and arrange the points in a grid. One such diagram for $M = 64$ square QAM is also given here in Figure 34.
- Figure 5.18 shows examples of constellations which use $M = 2^a$ for some **odd** integer a, and arrange the points in a grid. These are either rectangular grids, or use squares with the corners cut out.

24.4 Angle and Magnitude Representation

You can plot $\mathbf{a}_k$ in signal space and see that it has a magnitude (distance from the origin) of $|\mathbf{a}_k| = \sqrt{a_{k,0}^2 + a_{k,1}^2}$ and angle of $\angle \mathbf{a}_k = \tan^{-1} \frac{a_{k,1}}{a_{k,0}}$. In the continuous time signal $s(t)$ this is

$$s(t) = \sqrt{2}p(t)|\mathbf{a}_k| \cos(\omega_0 t + \angle \mathbf{a}_k)$$

24.5 Average Energy in M-QAM

Recall that the average energy is calculated as:

$$\begin{aligned} \mathcal{E}_s &= \frac{1}{M} \sum_{i=0}^{M-1} |\mathbf{a}_i|^2 \\ \mathcal{E}_b &= \frac{1}{M \log_2 M} \sum_{i=0}^{M-1} |\mathbf{a}_i|^2 \end{aligned}$$

where $\mathcal{E}_s$ is the average energy per symbol and $\mathcal{E}_b$ is the average energy per bit. We'll work in class some examples of finding $\mathcal{E}_b$ in different constellation diagrams.

24.6 Phase-Shift Keying

Some implementations of QAM limit the constellation to include only signal space vectors with equal magnitude, *i.e.*,

$$|\mathbf{a}_0| = |\mathbf{a}_1| = \cdots = |\mathbf{a}_{M-1}|$$

The points $\mathbf{a}_i$ for $i = 0, \ldots, M-1$ are uniformly spaced on the unit circle. Some examples are shown in Figure 35.

BPSK For example, binary phase-shift keying (BPSK) is the case of $M = 2$. Thus BPSK is the same as bipolar (2-ary) PAM. What is the probability of error in BPSK? The same as in bipolar PAM, *i.e.*, for equally probable symbols,

$$P[\text{error}] = \mathrm{Q}\left(\sqrt{\frac{2\mathcal{E}_b}{N_0}}\right).$$

QPSK $M = 4$ PSK is also called quadrature phase shift keying (QPSK), and is shown in Figure 35(a). Note that the rotation of the signal space diagram doesn't matter, so both 'versions' are identical in concept (although would be a slightly different implementation). Note how QPSK is the same as $M = 4$ square QAM.

24.7 Systems which use QAM

See [Couch 2007], wikipedia, and the Rice book:

- Digital Microwave Relay, various manufacturer-specific protocols. 6 GHz, and 11 GHz.
- Dial-up modems: use a $M = 16$ or $M = 8$ QAM constellation.
- DSL. G.DMT uses multicarrier (up to 256 carriers) methods (OFDM), and on each narrowband (4.3kHz) carrier, it can send up to 2^{15} QAM (32,768 QAM). G.Lite uses up to 128 carriers, each with up to $2^8 = 256$ QAM.

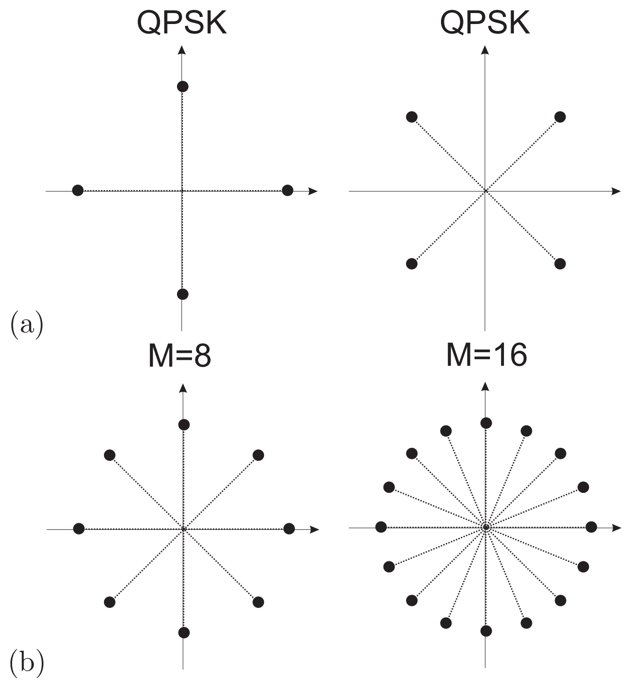


Figure 35: Signal constellations for (a) $M = 4$ PSK and (b) $M = 8$ and $M = 16$ PSK.

- Cable modems. Upstream: 6 MHz bandwidth channel, with 64 QAM or 256 QAM. Downstream: QPSK or 16 QAM.
- 802.11a and 802.11g: Adaptive modulation method, uses up to 64 QAM.
- Digital Video Broadcast (DVB): APSK used in ETSI standard.

25 QAM Probability of Error

25.1 Overview of Future Discussions on QAM

Before we get into details about the probabilities of error, you should realize that there are two main considerations when a constellation is designed for a particular M:

1. Points with equal distances separating them and their neighboring points tend to reduce probability of error.
2. The highest magnitude points (furthest from the origin) strongly (negatively) impact average bit energy.

There are also some practical considerations that we will discuss

1. Some constellations have more complicated decision regions which require more computation to implement in a receiver.
2. Some constellations are more difficult (energy-consuming) to amplify at the transmitter.

25.2 Options for Probability of Error Expressions

In order of preference:

1. *Exact formula.* In a few cases, there is an exact expression for $P[\text{error}]$ in an AWGN environment.
2. *Union bound.* This is a provable upper bound on the probability of error. It is not an approximation. It can be used for "worst case" analysis which is often very useful for engineering design of systems.
3. *Nearest Neighbor Approximation.* This is a way to get a solution that is analytically easier to handle. Typically these approximations are good at high $\frac{\mathcal{E}_b}{N_0}$.

25.3 Exact Error Analysis

Exact probability of error formulas in N-dimensional modulations can be very difficult to find. This is because our decisions regions are more complex than one threshold test. They require an integration of a N-D Gaussian pdf across an area. We needed a Q function to get a tail probability for a 1-D Gaussian pdf. Now we need not just a tail probability... but more like a section probability which is the volume under some part of the N-dimensional Gaussian pdf.

For example, consider M-ary PSK. Essentially, we must find calculate the probability of symbol error as 1

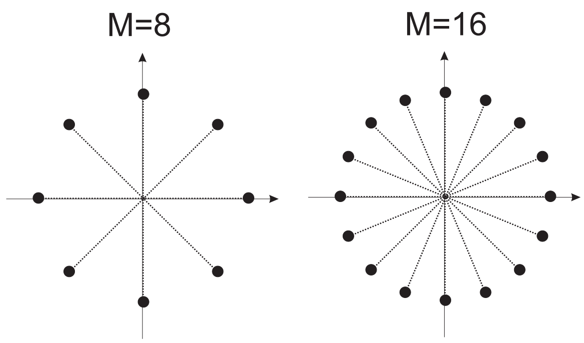


Figure 36: Signal space diagram for M-ary PSK for $M = 8$ and $M = 16$.

minus the area in the sector within $\pm\frac{\pi}{M}$ of the correct angle ϕ_i. This is,

$$P(\text{symbol error}) = 1 - \int_{r \in R_i} \frac{1}{2\pi\sigma^2} e^{-\frac{\|\mathbf{r}-\boldsymbol{\alpha}_i\|^2}{2\sigma^2}} \tag{40}$$

This integral is a double integral, and we don't generally have any exact expression to use to express the result in general.

25.4 Probability of Error in QPSK

In QPSK, the probability of error is analytically tractable. Consider the QPSK constellation diagram, when Gray encoding is used. You have already calculated the decision regions for each symbol; now consider the decision region for the first bit.

The decision is made using *only* one dimension, of the received signal vector $\mathbf{x}$, specifically x_1. Similarly, the second bit decision is made using only x_2. Also, the noise contribution to each element is independent. The decisions are decoupled – x_2 has no impact on the decision about bit one, and x_1 has no impact on the decision on bit two. Since we know the bit error probability for each bit decision (it is the same as bipolar PAM) we can see that the bit error probability is also

$$P[\text{error}] = \text{Q}\left(\sqrt{\frac{2\mathcal{E}_b}{N_0}}\right) \tag{41}$$

This is an extraordinary result – the bit rate will double in QPSK, but in theory, the bit error rate does not increase. As we will show later, the bandwidth of QPSK is identical to that of BPSK.

25.5 Union Bound

From 5510 or an equivalent class (or a Venn diagram) you may recall the probability formula, that for two events E and F that

$$P[E \cup F] = P[E] + P[F] - P[E \cap F]$$

You can prove this from the three axioms of probability. (This holds for *any* events E and F!) Then, using the above formula, and the first axiom of probability, we have that

$$P[E \cup F] \leq P[E] + P[F]. \tag{42}$$

Furthermore, from (42) it is straightforward to show that for *any* list of sets $E_1, E_2, \ldots E_n$ we have that

$$P\left[\bigcup_{i=1}^{n} E_i\right] \leq \sum_{i=1}^{n} P[E_i] \tag{43}$$

This is called the *union bound*, and it is very useful across communications. If you know one inequality, know this one. It is useful when the overlaps $E_i \cap E_j$ are small but difficult to calculate.

25.6 Application of Union Bound

In this class, our events are typically decision events. The event E_i may represent the event that we decide H_i, when some other symbol not equal to i was actually sent. In this case,

$$P[\text{symbol error}] \leq \sum_{i=1}^{n} P[E_i] \tag{44}$$

The union bound gives a conservative estimate. This can be useful for quick initial study of a modulation type.

Example: QPSK

First, let's study the union bound for QPSK, as shown in Figure 37(a). Assume $s_1(t)$ is sent. We know that (from previous analysis):

$$P[\text{error}] = \text{Q}\left(\sqrt{\frac{2\mathcal{E}_b}{N_0}}\right)$$

So the probability of symbol error is one minus the probability that there were no error in either bit,

$$P[\text{symbol error}] = 1 - \left(1 - \text{Q}\left(\sqrt{\frac{2\mathcal{E}_b}{N_0}}\right)\right)^2 \tag{45}$$

In contrast, let's calculate the union bound on the probability of error. We define the two error events as:

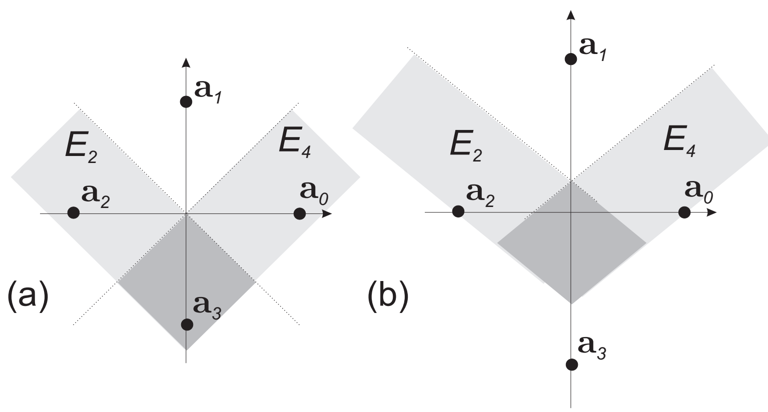


Figure 37: Union bound examples. (a) is QPSK with symbols of equal energy $\sqrt{\mathcal{E}_s}$. In (b) $\mathbf{a}_1 = -\mathbf{a}_3 = [0, \sqrt{3\mathcal{E}_s/2}]^T$ and $\mathbf{a}_2 = -\mathbf{a}_0 = [\sqrt{\mathcal{E}_s/2}, 0]^T$.

- E_2, the event that $\mathbf{r}$ falls on the wrong side of the decision boundary with $\mathbf{a}_2$.
- E_0, the event that $\mathbf{r}$ falls on the wrong side of the decision boundary with $\mathbf{a}_0$.

Then we write

$$P\left[\text{symbol error}|H_1\right] = P\left[E_2 \cup E_0 \cup E_3\right]$$

Questions:

1. Is this equation exact?
2. Is $E_2 \cup E_0 \cup E_3 = E_2 \cup E_0$? Why or why not?

Because of the answer to the question (2.),

$$P\left[\text{symbol error}|H_1\right] = P\left[E_2 \cup E_0\right]$$

But don't be confused. The Rice book will not tell you to do this! His strategy is to ignore redundancies, because we are only looking for an upper bound any way. Either way produces an upper bound, and in fact, both produce nearly identical numerical results. However, it is perfectly acceptable (and in fact a better bound) to remove redundancies and then apply the bound.

Then, we use the union bound.

$$P\left[\text{symbol error}|H_1\right] \leq P\left[E_2\right] + P\left[E_0\right]$$

These two probabilities are just the probability of error for a binary modulation, and both are identical, so

$$P\left[\text{symbol error}|H_1\right] \leq 2\text{Q}\left(\sqrt{\frac{2\mathcal{E}_b}{N_0}}\right)$$

What is missing / What is the difference in this expression compared to (45)?

See Figure 38 to see the union bound probability of error plot, compared to the exact expression. Only at very low E_b/N_0 is there any noticeable difference!

Example: 4-QAM with two amplitude levels
This is shown (poorly) in Figure 37(b). The amplitudes of the top and bottom symbols are $\sqrt{3}$ times the

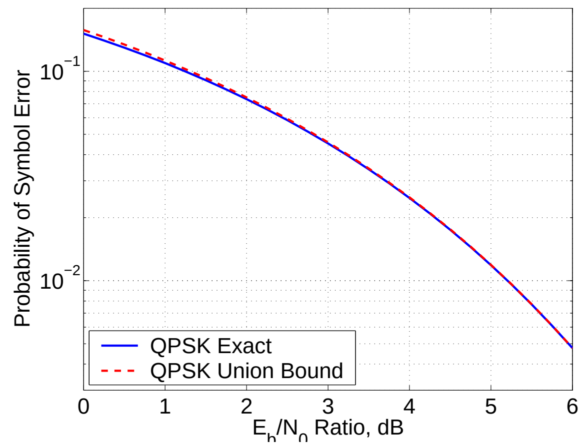


Figure 38: For QPSK, the exact probability of symbol error expression vs. the union bound.

amplitude of the symbols on the right and left. (They are positioned to keep the distance between points in the signal space equal to $\sqrt{2\mathcal{E}_s/N_0}$.) I am calling this "2-amplitude 4-QAM" (I made it up).

What is the union bound on the probability of symbol error, given H_1? **Solution:** Given symbol 1, the probability is the same as above. Defining E_2 and E_0 as above, these two distances between symbol $\mathbf{a}_1$ and $\mathbf{a}_2$ or $\mathbf{a}_0$ are the same: $\sqrt{2\mathcal{E}_s/N_0}$. Thus the formula for the union bound is the same.

What is the union bound on the probability of symbol error, given H_2? **Solution:** Now, it is

$$P\left[\text{symbol error}|H_2\right] \leq 3\text{Q}\left(\sqrt{\frac{2\mathcal{E}_b}{N_0}}\right)$$

So, overall, the union bound on probability of symbol error is

$$P\left[\text{symbol error}|H_2\right] \leq \frac{3\cdot 2 + 2\cdot 2}{4}\text{Q}\left(\sqrt{\frac{2\mathcal{E}_b}{N_0}}\right)$$

How about average energy? For QPSK, the symbol energies are all equal. Thus $\mathcal{E}_{av} = \mathcal{E}_s$. For the two-amplitude 4-QAM modulation,

$$\mathcal{E}_{av} = \mathcal{E}_s\frac{2(0.5) + 2(1.5)}{4} = \mathcal{E}_s$$

Thus there is no advantage to the two-amplitude QAM modulation in terms of average energy.

25.6.1 General Formula for Union Bound-based Probability of Error

This is given in Rice 6.76:

$$P\left[\text{symbol error}\right] \quad \leq \quad \frac{1}{M}\sum_{m=0}^{M-1}\sum_{\substack{n=0\\n\neq m}}^{M-1} P\left[\text{decide } H_n|H_m\right]$$

Note the $\leq$ sign. This means the actual $P\left[\text{symbol error}\right]$ will be *at most* this value. It is probably less than this value. This is a general formula and *is not necessarily the best* upper bound. What do we mean? If I draw any function that is always above the actual P [symbol error] formula, I have drawn an upper bound. But I could draw lots of functions that are upper bounds, some higher than others.

In particular, for some of the error events, decide $H_n|H_m$ may be redundant, and do not need to be included. We will talk about the concept of "neighboring" symbols to symbol i. These neighbors are the ones that are *necessary* to include in the union bound.

Note that the Rice book uses E_{avg} where I use $\mathcal{E}_s$ to denote average symbol energy. The Rice book uses E_b where I use $\mathcal{E}_b$ to denote average bit energy. I prefer $\mathcal{E}_s$ to make clear we are talking about Joules per *symbol*.

Given this, the *pairwise probability of error*, P [decide $H_n|H_m$], is given by:

$$P\left[\text{decide } H_n|H_m\right] = \mathrm{Q}\left(\sqrt{\frac{d_{m,n}^2}{2N_0}}\right) = \mathrm{Q}\left(\sqrt{\frac{d_{m,n}^2}{2\mathcal{E}_s}\frac{\mathcal{E}_s}{N_0}}\right) \tag{46}$$

where $d_{m,n}$ is the Euclidean distance between the symbols m and n in the signal space diagram. If we use a constant A when describing the signal space vectors (as we usually do), then, since $\mathcal{E}_s$ will be proportional to A^2 and $d_{m,n}^2$ will be proportional to A^2, the factors A will cancel out of the expression.

Proof of (46)?

Lecture 14

Today: (1) QAM Probability of Error, (2) Union Bound and Approximations

---

Lecture 15

Today: (1) M-ary PSK and QAM Analysis

26 QAM Probability of Error

26.1 Nearest-Neighbor Approximate Probability of Error

As it turns out, the probability of error is often well approximated by the terms in the Union Bound (Lecture 14, Section 2.6.1) with the smallest $d_{m,n}$. This is because higher $d_{m,n}$ means a higher argument in the Q-function, which in turn means a lower value of the Q-function. A little extra distance means a much lower value of the Q-function. So approximately,

$$\begin{aligned} P\left[\text{symbol error}\right] &\approx \frac{N_{min}}{M}\mathrm{Q}\left(\sqrt{\frac{d_{min}^2}{2N_0}}\right) \\ P\left[\text{symbol error}\right] &\approx \frac{N_{min}}{M}\mathrm{Q}\left(\sqrt{\frac{d_{min}^2}{2\mathcal{E}_s}\frac{\mathcal{E}_s}{N_0}}\right) \end{aligned} \tag{47}$$

where

$$d_{min} = \min_{m\neq n} d_{m,n}, \quad m = 0, \ldots, M-1, \text{ and } n = 0, \ldots, M-1.$$

and N_{min} is the number of pairs of symbols which are separated by distance d_{min}.

26.2 Summary and Examples

We've been formulating in class a pattern or step-by-step process for finding the probability of bit or symbol error in arbitrary M-ary PSK and QAM modulations. These steps include:

1. **Calculate the average energy per symbol**, $\mathcal{E}_s$, as a function of any amplitude parameters (typically we've been using A).

2. **Calculate the average energy per bit**, $\mathcal{E}_b$, using $\mathcal{E}_b = \mathcal{E}_s / \log_2 M$.

3. **Draw decision boundaries.** It is not necessary to include redundant boundary information, that is, an error region which is completely contained in a larger error region.

4. **Calculate pair-wise distances** between pairs of symbols which contribute to the decision boundaries.

5. **Convert distances** to be in terms of $\mathcal{E}_b/N_0$, if they are not already, using the answer to step (1.).

6. **Calculate the union bound** on probability of symbol error using the formula derived in the previous lecture,

$$\begin{aligned} P\left[\text{symbol error}\right] &\leq \frac{1}{M}\sum_{m=0}^{M-1}\sum_{\substack{n=0\\ n\neq m}}^{M-1} P\left[\text{decide } H_n|H_m\right] \\ P\left[\text{decide } H_n|H_m\right] &= \mathrm{Q}\left(\sqrt{\frac{d_{m,n}^2}{2N_0}}\right) \end{aligned}$$

7. **Approximate** using the nearest neighbor approximation if needed:

$$P\left[\text{symbol error}\right] \approx \frac{N_{min}}{M}\mathrm{Q}\left(\sqrt{\frac{d_{min}^2}{2N_0}}\right)$$

   where d_{min} is the shortest pairwise distance in the constellation diagram, and N_{min} is the number of ordered pairs (i,j) which are that distance apart, $d_{i,j} = d_{min}$ (don't forget to count both (i,j) and (j,i)!).

8. **Convert probability** of symbol error into probability of bit error if Gray encoding can be assumed,

$$P\left[\text{error}\right] \approx \frac{1}{\log_2 M} P\left[\text{symbol error}\right]$$

Example: Additional QAM / PSK Constellations

Solve for the probability of symbol error in the signal space diagrams in Figure 39. You should calculate:

- An exact expression if one should happen to be available,
- The Union bound,
- The nearest-neighbor approximation.

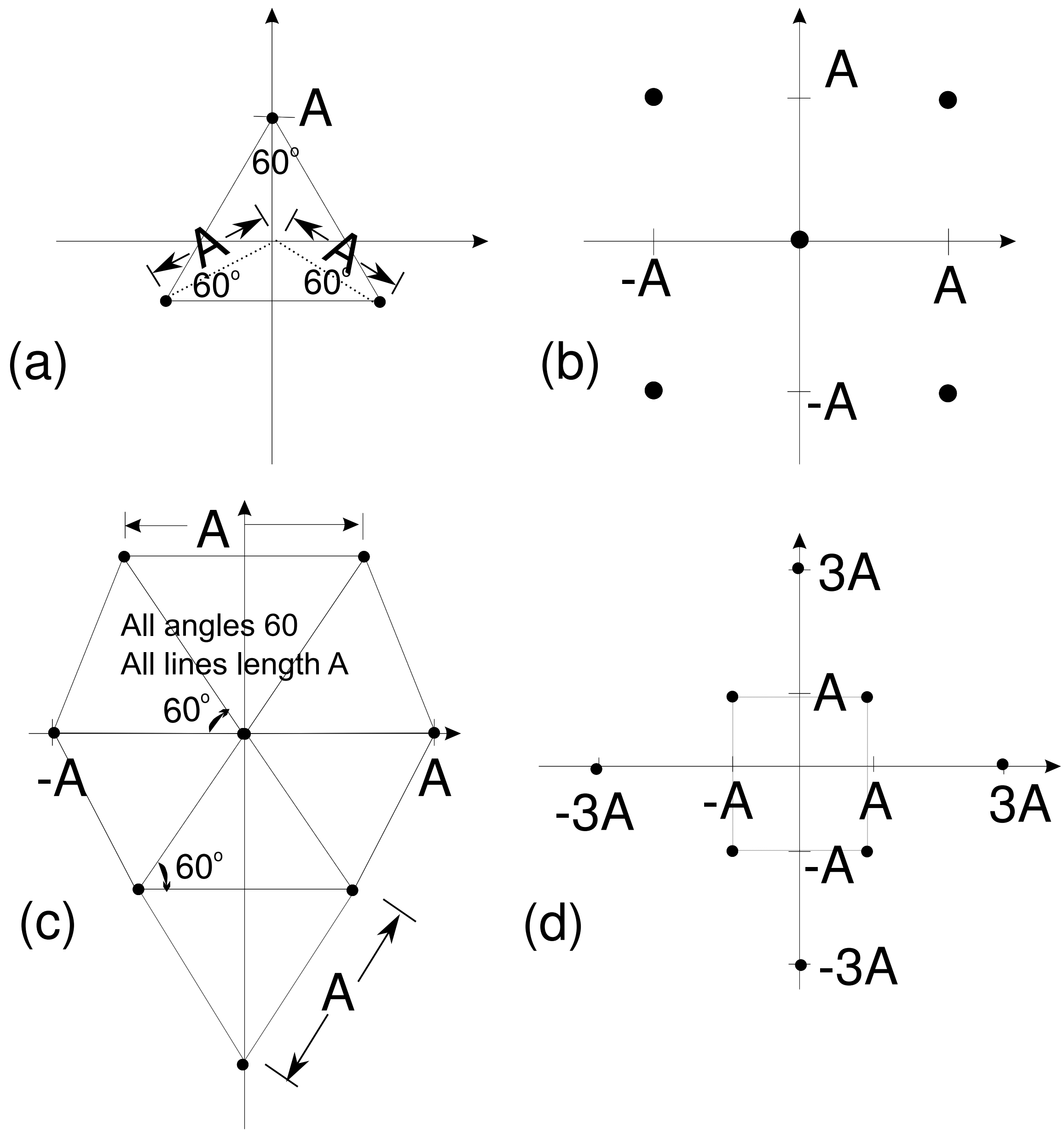


Figure 39: Signal space diagram for some example (made up) constellation diagrams.

on, first, probability of symbol error, and then also probability of bit error. The plots are in terms of amplitude A, but all probability of error expressions should be written in terms of $\mathcal{E}_b/N_0$.

Lecture 16

Today: (1) QAM Examples (Lecture 15) , (2) FSK

27 Frequency Shift Keying

Frequency modulation in general changes the center frequency over time:

$$x(t) = \cos(2\pi f(t)t).$$

Now, in frequency shift keying, symbols are selected to be sinusoids with frequency selected among a set of M different frequencies $\{f_0, f_1, \ldots f_{M-1}\}$.

27.1 Orthogonal Frequencies

We will want our cosine wave at these M different frequencies to be orthonormal. Consider $f_k = f_c + k\Delta f$, and thus

$$\phi_k(t) = \sqrt{2}p(t)\cos(2\pi f_c t + 2\pi k\Delta f t) \tag{48}$$

where $p(t)$ is a pulse shape, which for example, could be a rectangular pulse:

$$p(t) = \begin{cases} 1/\sqrt{T_s}, & 0 \leq t \leq T_s \\ 0, & o.w. \end{cases}$$

- Does this function have unit energy?
- Are these functions orthogonal?

Solution:

$$\begin{aligned} \langle \phi_k(t), \phi_m(t) \rangle &= \int_{t=-\infty}^{\infty} (2/T_s)\cos(2\pi f_c t + 2\pi k\Delta f t)\cos(2\pi f_c t + 2\pi k\Delta f t) \\ &= 1/T_s \int_{t=0}^{T_s} \cos(2\pi(k-m)\Delta f t)dt + \\ &\quad 1/T_s \int_{t=0}^{T_s} \cos(4\pi f_c t + 2\pi(k+m)\Delta f t)dt \\ &= 1/T_s \left[\frac{\sin(2\pi(k-m)\Delta f t)}{2\pi(k-m)\Delta f} \right|_{t=0}^{T_s} \\ &= \frac{\sin(2\pi(k-m)\Delta f T_s)}{2\pi(k-m)\Delta f T_s} \end{aligned}$$

Yes, they are orthogonal if $2\pi(k-m)\Delta f T_s$ is a multiple of π. (They are also approximately orthogonal if Δf is realy big, but we don't want to waste spectrum.) For general $k \neq m$, this requires that $\Delta f T_s = n/2$, *i.e.*,

$$\Delta f = n\frac{1}{2T_s} = n\frac{f_{sy}}{2}$$

for integer n. (Otherwise, no they're not.)

Thus we need to plug into (48) for $\Delta f = n\frac{1}{2T_s}$ for some integer n in order to have an orthonormal basis. What n? We will see that we either use an n of 1 or 2.

Signal space vectors $\mathbf{a}_i$ are given by

$$\begin{aligned} \mathbf{a}_0 &= [A, 0, \ldots, 0] \\ \mathbf{a}_1 &= [0, A, \ldots, 0] \\ &\vdots \\ \mathbf{a}_{M-1} &= [0, 0, \ldots, A] \end{aligned}$$

What is the energy per symbol? Show that this means that $A = \sqrt{\mathcal{E}_s}$.

For $M = 2$ and $M = 3$ these vectors are plotted in Figure 40.

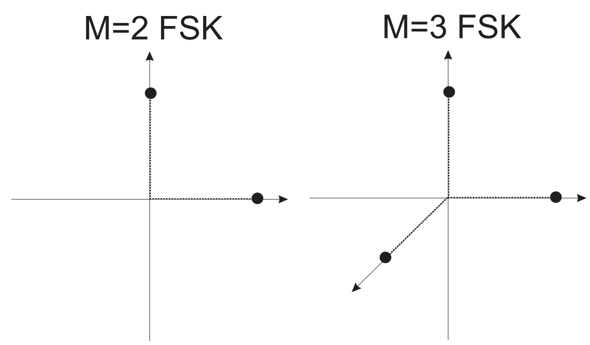


Figure 40: Signal space diagram for $M = 2$ and $M = 3$ FSK modulation.

27.2 Transmission of FSK

At the transmitter, FSK can be seen as a switch between M different carrier signals, as shown in Figure 41.

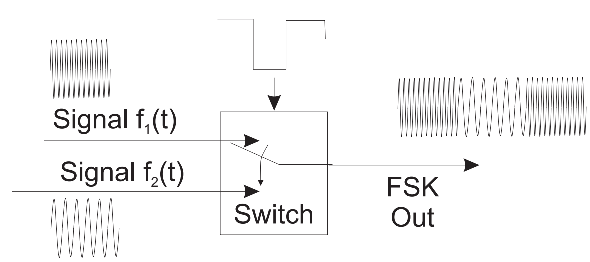


Figure 41: Block diagram of a binary FSK transmitter.

But usually it is generated from a single VCO, as seen in Figure 42.

Def'n: *Voltage Controlled Oscillator (VCO)*
A sinusoidal generator with frequency that linearly proportional to an input voltage.

Note that we don't need to sent square wave input into the VCO. In fact, bandwidth will be lower when we send less steep transitions, for example, SRRC pulses.

Def'n: *Continuous Phase Frequency Shift Keying*
FSK with no phase discontinuity between symbols. In other words, the phase of the output signal $\phi_k(t)$ does not change instantaneously at symbol boundaries iT_s for integer i, and thus $\phi(t^- + iT_s) = \phi(t^+ + iT_s)$ where t^- and t^+ are the limiting times just to the left and to the right of 0, respectively.

There are a variety of flavors of CPFSK, which are beyond the scope of this course.

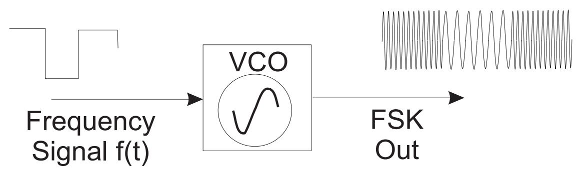


Figure 42: Block diagram of a binary FSK transmitter.

27.3 Reception of FSK

FSK reception is either phase coherent or phase non-coherent. Here, there are M possible carrier frequencies, so we'd need to know and be synchronized to M different phases θ_i, one for each symbol frequency:

$$\begin{aligned} &\cos(2\pi f_c t + \theta_0) \\ &\cos(2\pi f_c t + 2\pi \Delta f t + \theta_1) \\ &\vdots \\ &\cos(2\pi f_c t + 2\pi (M-1) \Delta f t + \theta_{M-1}) \end{aligned}$$

27.4 Coherent Reception

FSK reception can be done via a correlation receiver, just as we've seen for previous modulation types.

Each phase θ_k is estimated to be $\hat{\theta}_k$ by a separate phase-locked loop (PLL). In one flavor of CPFSK (called Sunde's FSK), the carrier $\cos(2\pi f_k t + \theta_k)$ is sent with the transmitted signal, to aid in demodulation (at the expense of the additional energy). This is the only case where I've heard of using coherent FSK reception.

As M gets high, coherent detection becomes difficult. These M PLLs must operate even though they can only synchronize when their symbol is sent, $1/M$ of the time (assuming equally-probable symbols). Also, having M PLLs is a drawback.

27.5 Non-coherent Reception

Notice that in Figure 40, the sign or phase of the sinusoid is not very important – only one symbol exists in each dimension. In non-coherent reception, we just measure the energy in each frequency.

This is more difficult than it sounds, though – we have a fundamental problem. As we know, for every frequency, there are two orthogonal functions, cosine and sine (see QAM and PSK). Since we will not know the phase of the received signal, we don't know whether or not the energy at frequency f_k correlates highly with the cosine wave or with the sine wave. If we only correlate it with one (the sine wave, for example), and the phase makes the signal the other (a cosine wave) we would get a inner product of zero!

The solution is that we need to correlate the received signal with both a sine and a cosine wave at the frequency f_k. This will give us two inner products, lets call them x_k^I using the capital I to denote in-phase and x_k^Q with Q denoting quadrature.

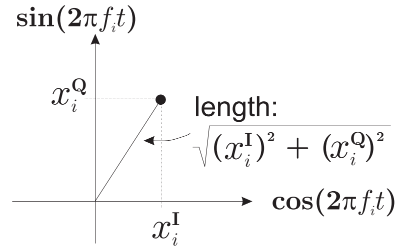


Figure 43: The energy in a non-coherent FSK receiver at one frequency f_k is calculated by finding its correlation with the cosine wave (x_k^I) and sine wave (x_k^Q) at the frequency of interest, f_k, and calculating the squared length of the vector $[x_k^I, x_k^Q]^T$.

The energy at frequency f_k, that is,

$$\mathcal{E}_{f_k} = (x_k^I)^2 + (x_k^Q)^2$$

is calculated for each frequency f_k, $k = 0, 1, \ldots, M-1$. You could prove this, but the optimal detector (Maximum likelihood detector) is then to decide upon the frequency with the maximum energy $\mathcal{E}_{f_k}$. Is this a threshold test?

We would need new analysis of the FSK non-coherent detector to find its analytical probability of error.

27.6 Receiver Block Diagrams

The block diagram of the the non-coherent FSK receiver is shown in Figures 45 and 46 (copied from the Proakis & Salehi book). Compare this to the coherent FSK receiver in Figure 44.

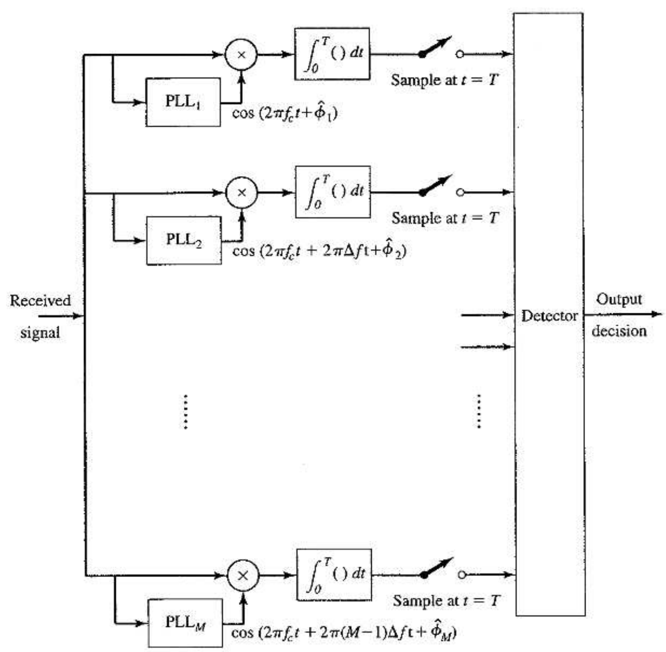


Figure 44: (Proakis & Salehi) Phase-coherent demodulation of M-ary FSK signals.

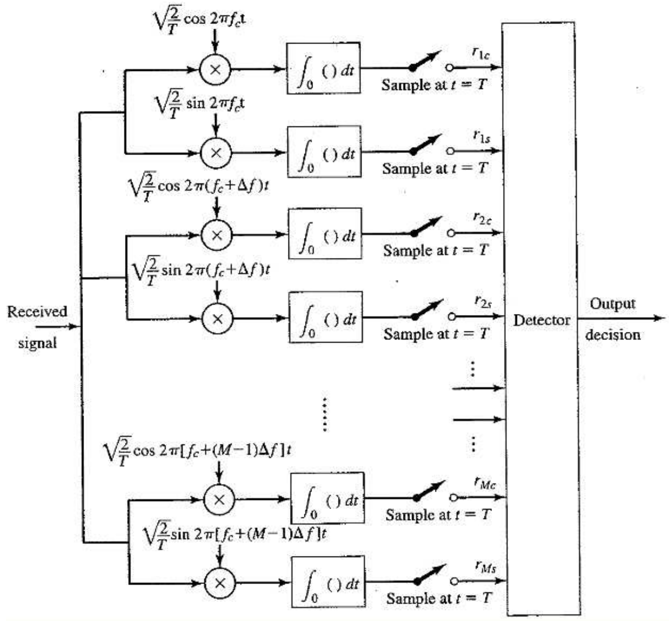


Figure 45: (Proakis & Salehi) Demodulation of M-ary FSK signals for non-coherent detection.

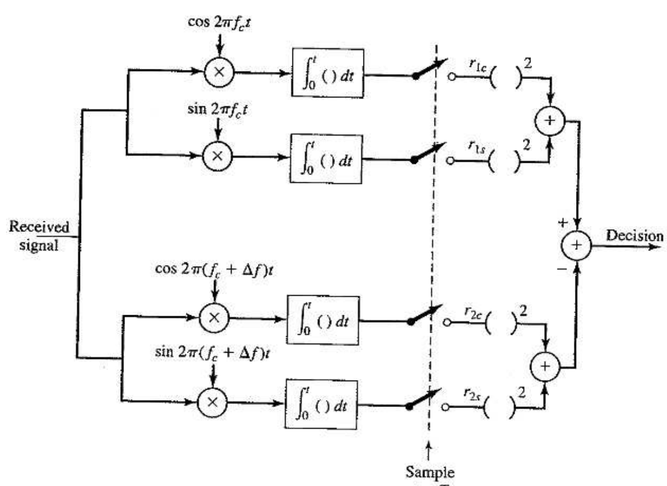


Figure 46: (Proakis & Salehi) Demodulation and square-law detection of binary FSK signals.

27.7 Probability of Error for Coherent Binary FSK

First, let's look at coherent detection of binary FSK.

1. What is the detection threshold line separating the two decision regions?
2. What is the distance between points in the Binary FSK signal space?

What is the probability of error for coherent binary FSK? It is the same as bipolar PAM, but the symbols are spaced differently (more closely) as a function of $\mathcal{E}_b$. We had that

$$P\left[\text{error}\right]_{2-ary} = \text{Q}\left(\sqrt{\frac{d_{0,1}^2}{2N_0}}\right)$$

Now, the spacing between symbols has reduced by a factor of $\sqrt{2}/2$ compared to bipolar PAM, to $d_{0,1} = \sqrt{2\mathcal{E}_b}$. So

$$P\left[\text{error}\right]_{2-Co-FSK} = \text{Q}\left(\sqrt{\frac{\mathcal{E}_b}{N_0}}\right)$$

For the same probability of bit error, binary FSK is about 1.5 dB better than OOK (requires 1.5 dB less energy per bit), but 1.5 dB worse than bipolar PAM (requires 1.5 dB more energy per bit).

27.8 Probability of Error for Noncoherent Binary FSK

The energy detector shown in Figure 46 uses the energy in each frequency and selects the frequency with maximum energy.

This energy is denoted r_m^2 in Figure 46 for frequency m and is

$$r_m^2 = r_{mc}^2 + r_{ms}^2$$

This energy measure is a statistic which measures how much energy was in the signal at frequency f_m. The 'envelope' is a term used for the square root of the energy, so r_m is termed the envelope.

Question: What will r_m^2 equal when the noise is very small?

As it turns out, given the non-coherent receiver and r_{mc} and r_{ms}, the envelope r_m is an optimum (sufficient) statistic to use to decide between $s_1 \ldots s_M$.

What do they do to prove this in Proakis & Salehi? They prove it for *binary* non-coherent FSK. It takes quite a bit of 5510 to do this proof.

1. Define the received vector $\mathbf{r}$ as a 4 length vector of the correlation of $r(t)$ with the sin and cos at each frequency f_1, f_2.
2. They formulate the prior probabilities $f_{\mathbf{r}|H_i}(\mathbf{r}|H_i)$. Note that this depends on θ_m, which is assumed to be uniform between 0 and 2π, and independent of the noise.

$$\begin{aligned} f_{\mathbf{r}|H_i}(\mathbf{r}|H_i) &= \int_0^{2\pi} f_{\mathbf{r},\theta_m|H_i}(\mathbf{r},\phi|H_i)d\phi \\ &= \int_0^{2\pi} f_{\mathbf{r}|\theta_m,H_i}(\mathbf{r}|\phi,H_i)f_{\theta_m|H_i}(\phi|H_i)d\phi \end{aligned} \tag{49}$$

Note that $f_{\mathbf{r}|\theta_m,H_0}(\mathbf{r}|\phi,H_0)$ is a 2-D Gaussian random vector with i.i.d. components.

3. They formulate the joint probabilities $f_{\mathbf{r}\cap H_0}(\mathbf{r}\cap H_0)$ and $f_{\mathbf{r}\cap H_1}(\mathbf{r}\cap H_1)$.
4. Where the joint probability $f_{\mathbf{r}\cap H_0}(\mathbf{r}\cap H_0)$ is greater than $f_{\mathbf{r}|H_1}(\mathbf{r}|H_1)$, the receiver decides H_0. Otherwise, it decides H_1.
5. The decisions in this last step, after manipulation of the pdfs, are shown to reduce to this decision (given that $P[H_0] = P[H_1]$):

$$\sqrt{r_{1c}^2 + r_{1s}^2} \underset{H_1}{\overset{H_0}{\gtrless}} \sqrt{r_{2c}^2 + r_{2s}^2}$$

The "envelope detector" can equally well be called the "energy detector", and it often is.

The above proof is simply FYI, and is presented since it does not appear in the Rice book.

An exact expression for the probability of error can be derived, as well. The proof is in Proakis & Salehi, Section 7.6.9, page 430, which is posted on WebCT. The expression for probability of error in binary non-coherent FSK is given by,

$$P[\text{error}]_{2-NC-FSK} = \frac{1}{2}\exp\left[-\frac{\mathcal{E}_b}{2N_0}\right] \tag{50}$$

The expressions for probability of error in binary FSK (both coherent and non-coherent) are important, and you should make note of them. You will use them to be able to design communication systems that use FSK.

Lecture 17

Today: (1) FSK Error Probabilities (2) OFDM

27.9 FSK Error Probabilities, Part 2

27.9.1 M-ary Non-Coherent FSK

For M-ary non-coherent FSK, the derivation in the Proakis & Salehi book, section 7.6.9 shows that

$$P[\text{symbol error}] = \sum_{n=1}^{M-1} (-1)^{n+1} \binom{M-1}{n} \frac{1}{n+1} e^{-\log_2 M \frac{n}{n+1}\frac{\mathcal{E}_b}{N_0}}$$

and

$$P[\text{error}]_{M-nc-FSK} = \frac{M/2}{M-1} P[\text{symbol error}]$$

See Figure 47.

Proof Summary: Our non-coherent receiver finds the energy in each frequency. These energy values no longer have a Gaussian distribution (due to the squaring of the amplitudes in the energy calculation). They instead are either Ricean (for the transmitted frequency) or Rayleigh distributed (for the "other" $M-1$ frequencies). The probability that the correct frequency is selected is the probability that the Ricean random variable is larger than all of the other random variables measured at the other frequencies.

Example: Probability of Error for Non-coherent $M = 2$ case
Use the above expressions to find the $P[\text{symbol error}]$ and $P[\text{error}]$ for binary non-coherent FSK.

Solution:

$$P[\text{symbol error}] = P[\text{bit error}] = \frac{1}{2} e^{-\frac{1}{2}\frac{\mathcal{E}_b}{N_0}}$$

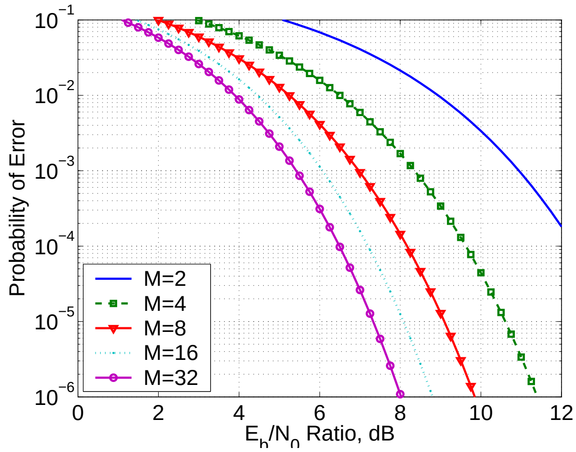


Figure 47: Probability of bit error for non-coherent reception of M-ary FSK.

27.9.2 Summary

The key insight is that probability of error decreases for increasing M. As $M \to \infty$, the probability of error goes to zero, for the case when $\frac{\mathcal{E}_b}{N_0} > -1.6dB$. Remember this for later - it is related to the Shannon-Hartley theorem on the limit of reliable communication.

27.10 Bandwidth of FSK

Carson's rule is used to calculate the bandwidth of FM signals. For M-ary FSK, it tells us that the approximate bandwidth is,

$$B_T = (M-1)\Delta f + 2B$$

where B is the one-sided bandwidth of the digital baseband signal. For the null-to-null bandwidth of raised-cosine pulse shaping, $B = (1+\alpha)/T_s$. Note for square wave pulses, $B = 1/T_s$.

28 Frequency Multiplexing

Frequency multiplexing is the division of the total bandwidth (call it B_T) into many smaller frequency bands, and sending a lower bit rate on each one.

Specifically, divide your B_T bandwidth into K subchannels, you now have B_T/K bandwidth on each subchannel. Note that in the multiplexed system, each subchannel has a symbol period of T_sK, longer by a factor of K so that the bandwidth of that subchannel has narrower bandwidth. In HW 7, you show that the total bit rate is the same in the multiplexed system, so you don't lose anything by this division. Furthermore, your transmission energy is the same. The energy would be divided by K in each subchannel, so the sum of the energy is constant.

For each band, we might send a PSK or PAM or QAM signal on that band. Our choice is arbitrary.

28.1 Frequency Selective Fading

Frequency selectivity is primarily a problem in wireless channels. But, frequency dependent gains are also experienced in DSL, because phone lines were not designed for higher frequency signals. For example, consider Figure 48, which shows an example frequency selective fading pattern, $10\log_{10}|H(f)|^2$, where $H(f)$ is an

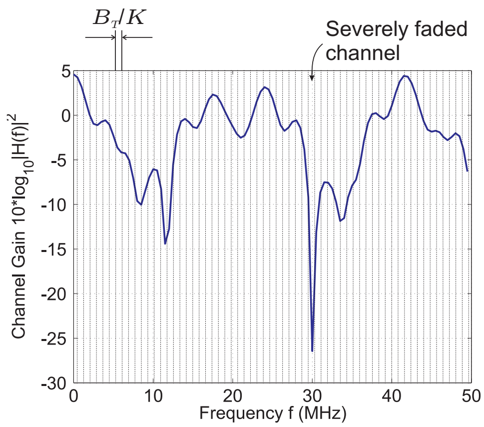


Figure 48: An example frequency selective fading pattern. The whole bandwidth is divided into K subchannels, and each subchannel's channel filter is mostly constant.

example wireless channel. This $H(f)$ might be experienced in an average outdoor channel. (The f is relative, *i.e.*, take the actual frequency $\tilde{f} = f + f_c$ for some center frequency f_c.)

- You can see that some channels experience severe attenuation, while most experience little attenuation. Some channels experience gain.
- For stationary transmitter and receiver and stationary f, the channel stays constant over time. If you put down the transmitter and receiver and have them operate at a particular f', then the loss $10 \log_{10} |H(f')|^2$ will not change throughout the communication. If the channel loss is too great, than the error rate will be too high.

Problems:

1. Wideband: The bandwidth is used for one channel. The $H(f)$ acts as a filter, which introduces ISI.
2. Narrowband: Part of the bandwidth is used for one narrow channel, at a lower bit rate. The power is increased by a *fade margin* which makes the $\frac{\mathcal{E}_b}{N_0}$ high enough for low BER demodulation except in the most extreme fading situations.

When designing for frequency selective fading, designers may use a wideband modulation method which is robust to frequency selective fading. For example,

- Frequency multiplexing methods (including OFDM),
- Direct-sequence spread spectrum (DS-SS), or
- Frequency-hopping spread spectrum (FH-SS).

We're going to mention frequency multiplexing.

28.2 Benefits of Frequency Multiplexing

Now, bit errors are not an 'all or nothing' game. In frequency multiplexing, there are K parallel bitstreams, each of rate R/K, where R is the total bit rate. As a first order approximation, a subchannel either experiences a high SNR and makes no errors; or is in a severe fade, has a very low SNR, and experiences a BER of 0.5 (the worst bit error rate!). If β is the probability that a subchannel experiences a severe fade, the overall probability of error will be 0.5β.

Compare this to a single narrowband channel, which has probability β that it will have a 0.5 probability of error. Similarly, a wideband system with very high ISI might be completely unable to demodulate the received signal.

Frequency multiplexing is typically combined with channel coding designed to correct a small percentage of bit errors.

28.3 OFDM as an extension of FSK

This is section 5.5 in the Rice book.

In FSK, we use a single basis function at each of different frequencies. In QAM, we use two basis functions at the same frequency. OFDM is the combination:

$$
\begin{aligned}
\phi_{0,I}(t) &= \begin{cases} \sqrt{2/T_s}\cos(2\pi f_c t), & 0 \le t \le T_s \\ 0, & o.w. \end{cases} \\
\phi_{0,Q}(t) &= \begin{cases} \sqrt{2/T_s}\sin(2\pi f_c t), & 0 \le t \le T_s \\ 0, & o.w. \end{cases} \\
\phi_{1,I}(t) &= \begin{cases} \sqrt{2/T_s}\cos(2\pi f_c t + 2\pi\Delta f t), & 0 \le t \le T_s \\ 0, & o.w. \end{cases} \\
\phi_{1,Q}(t) &= \begin{cases} \sqrt{2/T_s}\sin(2\pi f_c t + 2\pi\Delta f t), & 0 \le t \le T_s \\ 0, & o.w. \end{cases} \\
&\vdots \\
\phi_{M-1,I}(t) &= \begin{cases} \sqrt{2/T_s}\cos(2\pi f_c t + 2\pi(M-1)\Delta f t), & 0 \le t \le T_s \\ 0, & o.w. \end{cases} \\
\phi_{M-1,Q}(t) &= \begin{cases} \sqrt{2/T_s}\sin(2\pi f_c t + 2\pi(M-1)\Delta f t), & 0 \le t \le T_s \\ 0, & o.w. \end{cases}
\end{aligned}
$$

where $\Delta f = \frac{1}{2T_s}$. These are all orthogonal functions! We can transmit much more information than possible in M-ary FSK. (Note we have $2M$ basis functions here!)

The signal on subchannel k might be represented as:

$$x_k(t) = \sqrt{2/T_s}\left[a_{k,I}(t)\cos(2\pi f_k t) + a_{k,Q}(t)\sin(2\pi f_k t)\right]$$

The complex baseband signal of the sum of all K signals might then be represented as

$$
\begin{aligned}
x_l(t) &= \sqrt{2/T_s}\mathbb{R}\left\{\sum_{k=1}^{K}(a_{k,I}(t) + j a_{k,Q}(t))e^{j2\pi k\Delta f t}\right\} \\
x_l(t) &= \sqrt{2/T_s}\mathbb{R}\left\{\sum_{k=1}^{K} A_k(t) e^{j2\pi k\Delta f t}\right\} \qquad (51)
\end{aligned}
$$

where $A_k(t) = a_{k,I}(t) + j a_{k,Q}(t)$. Does this look like an inverse discrete fourier transform? If yes, than you can see why it is possible to use an IFFT and FFT to generate the complex baseband signal.

1. FFT implementation: There is a particular implementation of the transmitter and receiver that use FFT/IFFT operations. This avoids having K independent transmitter chains and receiver chains. The

FFT implementation (and the speed and ease of implementation of the FFT in hardware) is why OFDM is popular.

Since the K carriers are orthogonal, the signal is like K-ary FSK. But, rather than transmitting on one of the K carriers at a given time (like FSK) we transmit information in parallel on all K channels simultaneously. An example state space diagram for $K = 3$ and PAM on each channel is shown in Figure 49.

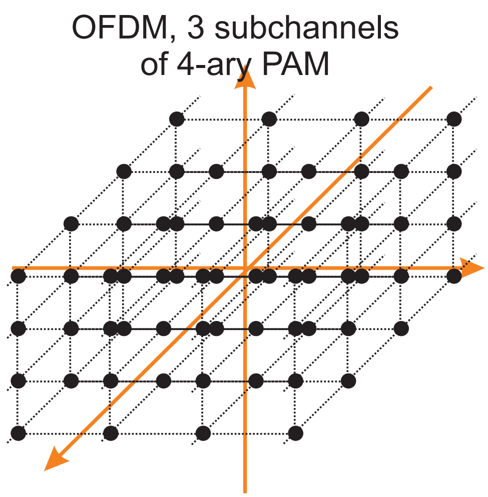


Figure 49: Signal space diagram for $K = 3$ subchannel OFDM with 4-PAM on each channel.

Example: 802.11a
IEEE 802.11a uses OFDM with 52 subcarriers. Four of the subcarriers are reserved for pilot tones, so effectively 48 subcarriers are used for data. Each data subcarrier can be modulated in different ways. One example is to use 16 square QAM on each subcarrier (which is 4 bits per symbol per subcarrier). The symbol rate in 802.11a is 250k/sec. Thus the bit rate is

$$250 \times 10^3 \frac{\text{OFDM symbols}}{\text{sec}} 48 \frac{\text{subcarriers}}{\text{OFDM symbol}} 4 \frac{\text{coded bits}}{\text{subcarrier}} = 48 \frac{\text{Mbits}}{\text{sec}}$$

Lecture 18

Today: Comparison of Modulation Methods

29 Comparison of Modulation Methods

This section first presents some miscellaneous information which is important to real digital communications systems but doesn't fit nicely into other lectures.

29.1 Differential Encoding for BPSK

Def'n: *Coherent Reception*
The reception of a signal when its carrier phase is explicitly determined and used for demodulation.

For coherent reception of PSK, will always need some kind of phase synchronization in BPSK. Typically, this means transmitting a training sequence.

For non-coherent reception of PSK, we use differential encoding (at the transmitter) and decoding (at the receiver).

29.1.1 DPSK Transmitter

Now, consider the bit sequence $\{b_n\}$, where b_n is the nth bit that we want to send. The sequence b_n is a sequence of 0's and 1's. How do we decide which phase to send? Prior to this, we've said, send $\mathbf{a}_0$ if $b_n = 0$, and send $\mathbf{a}_1$ if $b_n = 1$.

Instead of setting k for $\mathbf{a}_k$ only as a function of b_n, in differential encoding, we also include k_{n-1}. Now,

$$k_n = \begin{cases} k_{n-1}, & b_n = 0 \\ 1 - k_{n-1}, & b_n = 1 \end{cases}$$

Note that $1 - k_{n-1}$ is the complement or negation of k_{n-1} – if $k_{n-1} = 1$ then $1 - k_{n-1} = 0$; if $k_{n-1} = 0$ then $1 - k_{n-1} = 1$. Basically, for differential BPSK, a switch in the angle of the signal space vector from 0^o to 180^o or vice versa indicates a bit 1; while staying at the same angle indicates a bit 0.

Note that we have to just agree on the "zero" phase. Typically $k_0 = 0$.

Example: Differential encoding

Let $\mathbf{b} = [1, 0, 1, 0, 1, 1, 1, 0, 0]$. Assume $b_0 = 0$. What symbols $\mathbf{k} = [k_0, \ldots, k_8]^T$ will be sent? **Solution:**

$$\mathbf{k} = [1, 1, 0, 0, 1, 0, 1, 1, 1]^T$$

These values of k_n correspond to a symbol stream with phases:

$$\angle \mathbf{a}_k = [\pi, \pi, 0, 0, \pi, 0, \pi, \pi, \pi]^T$$

29.1.2 DPSK Receiver

Now, at the receiver, we find b_n by comparing the phase of x_n to the phase of x_{n-1}. What our receiver does, is to measure the statistic

$$\Re\{x_n x_{n-1}^*\} = |x_n||x_{n-1}| \cos(\angle x_n - \angle x_{n-1})$$

as the statistic – if it less than zero, decide $b_n = 1$, and if it is greater than zero, decide $b_n = 0$.

Example: Differential decoding

1. Assuming no phase shift in the above encoding example, show that the receiver will decode the original bitstream with differential decoding. **Solution:** Assuming $\phi_{i_0} = 0$,

$$\hat{b}_n = [1, 0, 1, 0, 1, 1, 1, 0, 0]^T.$$

2. Now, assume that all bits are shifted π radians and we receive

$$\angle \mathbf{x}' = [0, 0, \pi, \pi, 0, \pi, 0, 0, 0].$$

What will be decoded at the receiver? **Solution:**

$$\hat{b}_n = [0, 0, 1, 0, 1, 1, 1, 0, 0].$$

Rotating *all* symbols by π radians only causes one bit error.

29.1.3 Probability of Bit Error for DPSK

The probability of bit error in DPSK is slightly worse than that for BPSK:

$$P[\text{error}] = \frac{1}{2}\exp\left(-\frac{\mathcal{E}_b}{N_0}\right)$$

For a constant probability of error, DPSK requires about 1 dB more $\frac{\mathcal{E}_b}{N_0}$ than BPSK, which has probability of bit error $Q\left(\sqrt{\frac{2\mathcal{E}_b}{N_0}}\right)$. Both are plotted in Figure 50.

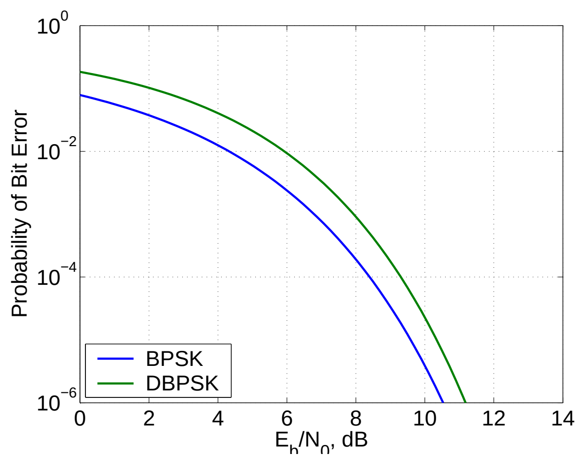


Figure 50: Comparison of probability of bit error for BPSK and Differential BPSK.

29.2 Points for Comparison

- Linear amplifiers (transmitter complexity)
- Receiver complexity
- Fidelity ($P[\text{error}]$) vs. $\frac{\mathcal{E}_b}{N_0}$
- Bandwidth efficiency

η	$\alpha = 0$	$\alpha = 0.5$	$\alpha = 1$
BPSK	1.0	0.67	0.5
QPSK	2.0	1.33	1.5
16-QAM	4.0	2.67	2.0
64-QAM	6.0	4.0	3.0

Table 2: Bandwidth efficiency of PSK and QAM modulation methods using raised cosine filtering as a function of α.

29.3 Bandwidth Efficiency

We've talked about measuring data rate in bits per second. We've also talked about Hertz, *i.e.*, the quantity of spectrum our signal will use. Typically, we can scale a system, to increase the bit rate by decreasing the symbol period, and correspondingly increase the bandwidth. This relationship is typically linearly proportional.

Def'n: *Bandwidth efficiency*
The bandwidth efficiency, typically denoted η, is the ratio of bits per second to bandwidth:

$$\eta = R_b/B_T$$

Bandwidth efficiency depends on the definition of "bandwidth". Since it is usually used for comparative purposes, we just make sure we use the same definition of bandwidth throughout a comparison.

The key figure of merit: bits per second / Hertz, *i.e.*, bps/Hz.

29.3.1 PSK, PAM and QAM

In these three modulation methods, the bandwidth is largely determined by the pulse shape. For root raised cosine filtering, the null-null bandwidth is $1 + \alpha$ times the bandwidth of the case when we use pure sinc pulses. The transmission bandwidth (for a bandpass signal) is

$$B_T = \frac{1+\alpha}{T_s}$$

Since T_s is seconds per symbol, we divide by $\log_2 M$ bits per symbol to get $T_b = T_s/\log_2 M$ seconds per bit, or

$$B_T = \frac{(1+\alpha)R_b}{\log_2 M}$$

where $R_b = 1/T_b$ is the bit rate.

Bandwidth efficiency is then

$$\eta = R_b/B_T = \frac{\log_2 M}{1+\alpha}$$

See Table 2 for some numerical examples.

29.3.2 FSK

We've said that the bandwidth of FSK is,

$$B_T = (M-1)\Delta f + 2B$$

where B is the one-sided bandwidth of the digital baseband signal. For the null-to-null bandwidth of raised-cosine pulse shaping, $2B = (1+\alpha)/T_s$. So,

$$B_T = (M-1)\Delta f + (1+\alpha)/T_s = \frac{R_b}{\log_2 M}\left\{(M-1)\Delta f T_s + (1+\alpha)\right\}$$

since $R_b = 1/T_s$ for

$$\eta = R_b/B_T = \frac{\log_2 M}{(M-1)\Delta f T_s + (1+\alpha)}$$

If $\Delta f = 1/T_s$ (required for non-coherent reception),

$$\eta = \frac{\log_2 M}{M+\alpha}$$

29.4 Bandwidth Efficiency vs. $\frac{\mathcal{E}_b}{N_0}$

For each modulation format, we have quantities of interest:

- Bandwidth efficiency, and
- Energy per bit ($\frac{\mathcal{E}_b}{N_0}$) requirements to achieve a given probability of error.

Example: Bandwidth efficiency vs. $\frac{\mathcal{E}_b}{N_0}$ for $M = 8$ PSK
What is the required $\frac{\mathcal{E}_b}{N_0}$ for 8-PSK to achieve a probability of bit error of 10^{-6}? What is the bandwidth efficiency of 8-PSK when using 50% excess bandwidth?

Solution: Given in Rice Figure 6.3.5 to be about 14 dB, and 2.

We can plot these (required $\frac{\mathcal{E}_b}{N_0}$, bandwidth efficiency) pairs. See Rice Figure 6.3.6.

29.5 Fidelity (P [error]) vs. $\frac{\mathcal{E}_b}{N_0}$

Main modulations which we have evaluated probability of error vs. $\frac{\mathcal{E}_b}{N_0}$:

1. M-ary PAM, including Binary PAM or BPSK, OOK, DPSK.
2. M-ary PSK, including QPSK.
3. Square QAM
4. Non-square QAM constellations
5. FSK, M-ary FSK

In this part of the lecture we will break up into groups and derive: (1) the probability of error and (2) probability of symbol error formulas for these types of modulations.

See also Rice pages 325-331.

Name	P [**symbol error**]	P [**error**]
BPSK	$= \text{Q}\left(\sqrt{\frac{2\mathcal{E}_b}{N_0}}\right)$	same
OOK	$= \text{Q}\left(\sqrt{\frac{\mathcal{E}_b}{N_0}}\right)$	same
DPSK	$= \frac{1}{2}\exp\left(-\frac{\mathcal{E}_b}{N_0}\right)$	same
M-PAM	$= \frac{2(M-1)}{M}\text{Q}\left(\sqrt{\frac{6\log_2 M}{M^2-1}\frac{\mathcal{E}_b}{N_0}}\right)$	$\approx \frac{1}{\log_2 M}P\left[\text{symbol error}\right]$
QPSK		$= \text{Q}\left(\sqrt{\frac{2\mathcal{E}_b}{N_0}}\right)$
M-PSK	$\leq 2\text{Q}\left(\sqrt{2(\log_2 M)\sin^2(\pi/M)\frac{\mathcal{E}_b}{N_0}}\right)$	$\approx \frac{1}{\log_2 M}P\left[\text{symbol error}\right]$
Square M-QAM		$\approx \frac{4}{\log_2 M}\frac{(\sqrt{M}-1)}{\sqrt{M}}\text{Q}\left(\sqrt{\frac{3\log_2 M}{M-1}\frac{\mathcal{E}_b}{N_0}}\right)$
2-non-co-FSK	$= \frac{1}{2}\exp\left[-\frac{\mathcal{E}_b}{2N_0}\right]$	same
M-non-co-FSK	$= \sum_{n=1}^{M-1}\binom{M-1}{n}\frac{(-1)^{n+1}}{n+1}\exp\left[-\frac{n\log_2 M}{n+1}\frac{\mathcal{E}_b}{N_0}\right]$	$= \frac{M/2}{M-1}P\left[\text{symbol error}\right]$
2-co-FSK	$= \text{Q}\left(\sqrt{\frac{\mathcal{E}_b}{N_0}}\right)$	same
M-co-FSK	$\leq (M-1)\text{Q}\left(\sqrt{\log_2 M\frac{\mathcal{E}_b}{N_0}}\right)$	$= \frac{M/2}{M-1}P\left[\text{symbol error}\right]$

Table 3: Summary of probability of bit and symbol error formulas for several modulations.

29.6 Transmitter Complexity

What makes a transmitter more complicated?

- Need for a linear power amplifier
- Higher clock speed
- Higher transmit powers
- Directional antennas

29.6.1 Linear / Non-linear Amplifiers

An amplifier uses DC power to take an input signal and increase its amplitude at the output. If P_{DC} is the input power (*e.g.*, from battery) to the amplifier, and P_{out} is the output signal power, the power efficiency is rated as $\eta_P = P_{out}/P_{DC}$.

Amplifiers are separated into 'classes' which describe their configuration (circuits), and as a result of their configuration, their linearity and power efficiency.

- *Class A*: linear amplifiers with maximum power efficiency of 50%. Output signal is a scaled up version of the input signal. Power is dissipated at all times.
- *Class B*: linear amplifiers turn on for half of a cycle (conduction angle of 180^o) with maximum power efficiency of 78.5%. Two in parallel are used to amplify a sinusoidal signal, one for the positive part and one for the negative part.
- *Class AB*: Like class B, but each amplifier stays on slightly longer to reduce the "dead zone" at zero. That is, the conduction angle is slightly higher than 180^o.

- *Class C*: A class C amplifier is closer to a switch than an amplifier. This generates high distortion, but then is band-pass filtered or tuned to the center frequency to force out spurious frequencies. Class C non-linear amplifiers have power efficiencies around 90%. Can only amplify signals with a nearly constant envelope.

In order to double the power efficiency, battery-powered transmitters are often willing to use Class C amplifiers. They can do this if their output signal has constant envelope. This means that, if you look at the constellation diagram, you'll see a circle. The signal must never go through the origin (envelope of zero) or even near the origin.

29.7 Offset QPSK

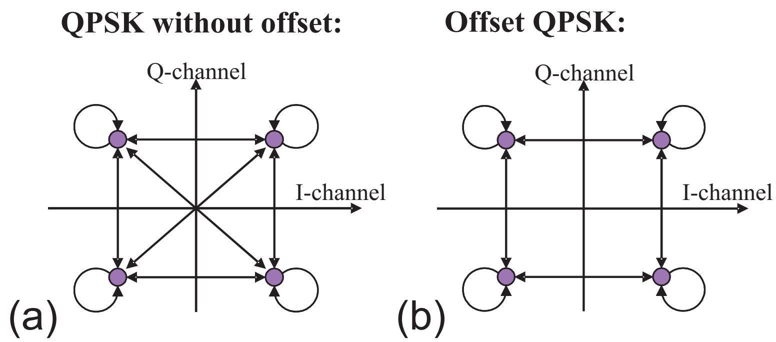


Figure 51: Constellation Diagram for (a) QPSK and (b) O-QPSK.

For QPSK, we wrote the modulated signal as

$$s(t) = \sqrt{2}p(t)\cos(\omega_0 t + \angle\mathbf{a}(t))$$

where $\angle\mathbf{a}(t)$ is the angle of the symbol chosen to be sent at time t. It is in a discrete set,

$$\angle\mathbf{a}(t) \in \{\frac{\pi}{4}, \frac{3\pi}{4}, \frac{5\pi}{4}, \frac{7\pi}{4}\}$$

and $p(t)$ is the pulse shape. We could have also written $s(t)$ as

$$s(t) = \sqrt{2}p(t)\left[a_0(t)\cos(\omega_0 t) + a_1(t)\sin(\omega_0 t)\right]$$

The problem is that when the phase changes 180 degrees, the signal $s(t)$ will go through zero, which precludes use of a class C amplifier. See Figure 52(b) and Figure 51 (a) to see this graphically.

For offset QPSK (OQPSK), we delay the quadrature $T_s/2$ with respect to the in-phase. Rewriting $s(t)$

$$s(t) = \sqrt{2}p(t)a_0(t)\cos(\omega_0 t) + \sqrt{2}p(t - T_s/2)a_1(t - T_s/2)\sin(\omega_0(t - T_s/2))$$

At the receiver, we just need to delay the sampling on the quadrature half of a sample period with respect to the in-phase signal. The new transmitted signal takes the same bandwidth and average power, and has the same E_b/N_0 vs. probability of bit error performance. However, the envelope $|s(t)|$ is largely constant. See Figure 52 for a comparison of QPSK and OQPSK.

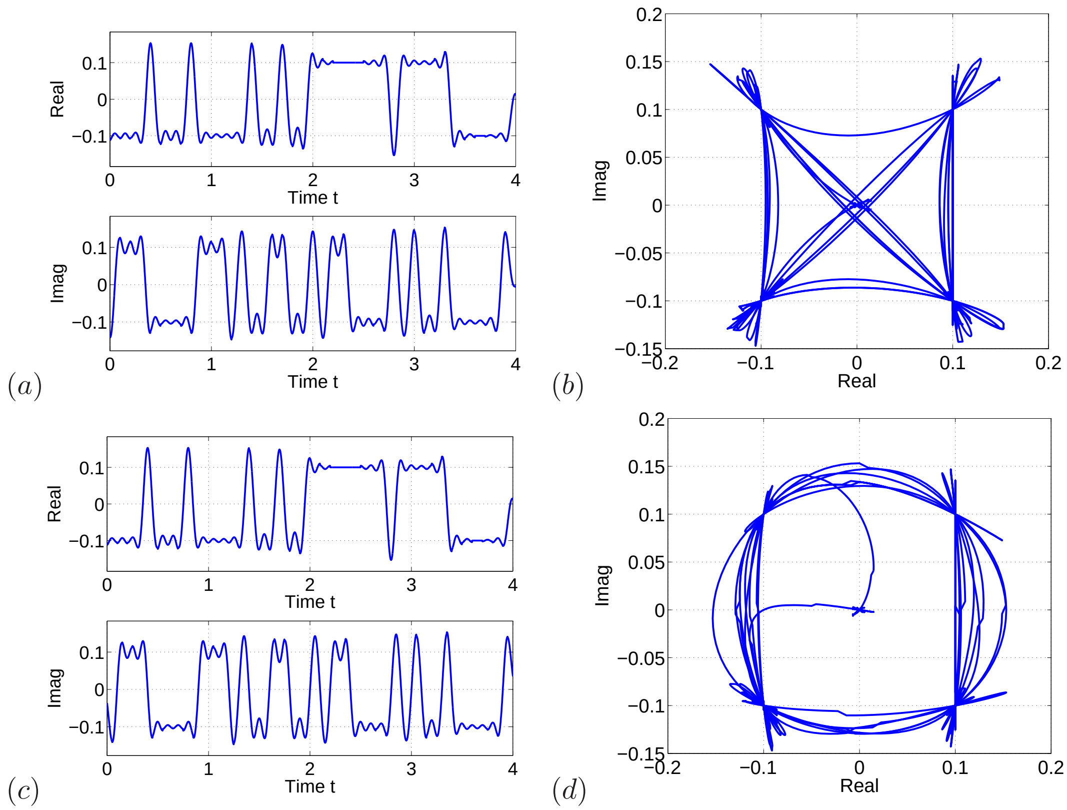


Figure 52: Matlab simulation of (a-b) QPSK and (c-d) O-QPSK, showing the (d) largely constant envelope of OQPSK, compared to (b) that for QPSK.

29.8 Receiver Complexity

What makes a receiver more complicated?

- Synchronization (carrier, phase, timing)
- Multiple parallel receiver chains

Lecture 19

Today: (1) Link Budgets and System Design

30 Link Budgets and System Design

As a digital communication system designer, your mission (if you choose to accept it) is to achieve:

1. High data rate
2. High fidelity (low bit error rate)
3. Low transmit power

4. Low bandwidth
5. Low transmitter/receiver complexity

But this is truly a mission impossible, because you can't have everything at the same time. So the system design depends on what the desired system really needs and what are the acceptable tradeoffs. Typically some subset of requirements are given; for example, given the bandwidth limits, the received signal power and noise power, and bit error rate limit, what data rate can be achieved? Using which modulation?

To answer this question, it is just a matter of setting some system variables (at the given limits) and determining what that means about the values of other system variables. See Figure 53. For example, if I was given the bandwidth limits and C/N_0 ratio, I'd be able to determine the probability of error for several different modulation types.

In this lecture, we discuss this procedure, and define each of the variables we see in Figure 53, and to what they are related. This lecture is about system design, and it cuts across ECE classes; in particular circuits, radio propagation, antennas, optics, and the material from this class. You are expected to apply what you have learned in other classes (or learn the functional relationships that we describe here).

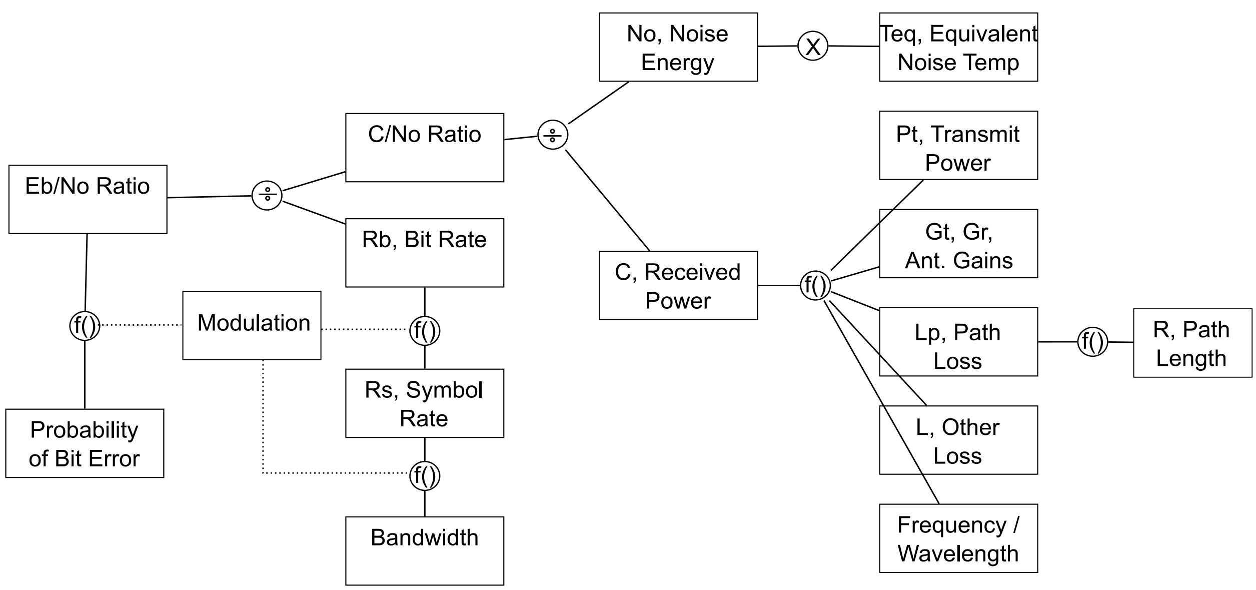


Figure 53: Relationships between important system design variables (rectangular boxes). Functional relationships between variables are given as circles – if the relationship is a single operator (*e.g.*, division), the operator is drawn in, otherwise it is left as an unnamed function $f()$. For example, C/N_0 is shown to have a divide by relationship with C and N_0. The effect of the choice of modulation impacts several functional relationships, *e.g.*, the relationship between probability of bit error and $\frac{\mathcal{E}_b}{N_0}$, which is drawn as a dotted line.

30.1 Link Budgets Given C/N_0

The received power is denoted C, it has units of Watts. What is C/N_0? It is received power divided by noise energy. It is an odd quantity, but it summarizes what we need to know about the signal and the noise for the purposes of system design.

- We often describe both the received power at a receiver P_R, but in the Rice book it is typically denoted C.

- We know the probability of bit error is typically written as a function of $\frac{\mathcal{E}_b}{N_0}$. The noise energy is N_0. The bit energy is $\mathcal{E}_b$. We can write $\mathcal{E}_b = CT_b$, since energy is power times time. To separate the effect of T_b, we often denote:
$$\frac{\mathcal{E}_b}{N_0} = \frac{C}{N_0}T_b = \frac{C/N_0}{R_b}$$
where $R_b = 1/T_b$ is the bit rate. In other words, $C/N_0 = \frac{\mathcal{E}_b}{N_0}R_b$ What are the units of C/N_0? *Answer: Hz, 1/s.*

- Note that people often report C/N_0 in dB Hz, which is
$$10\log_{10}\frac{C}{N_0}$$

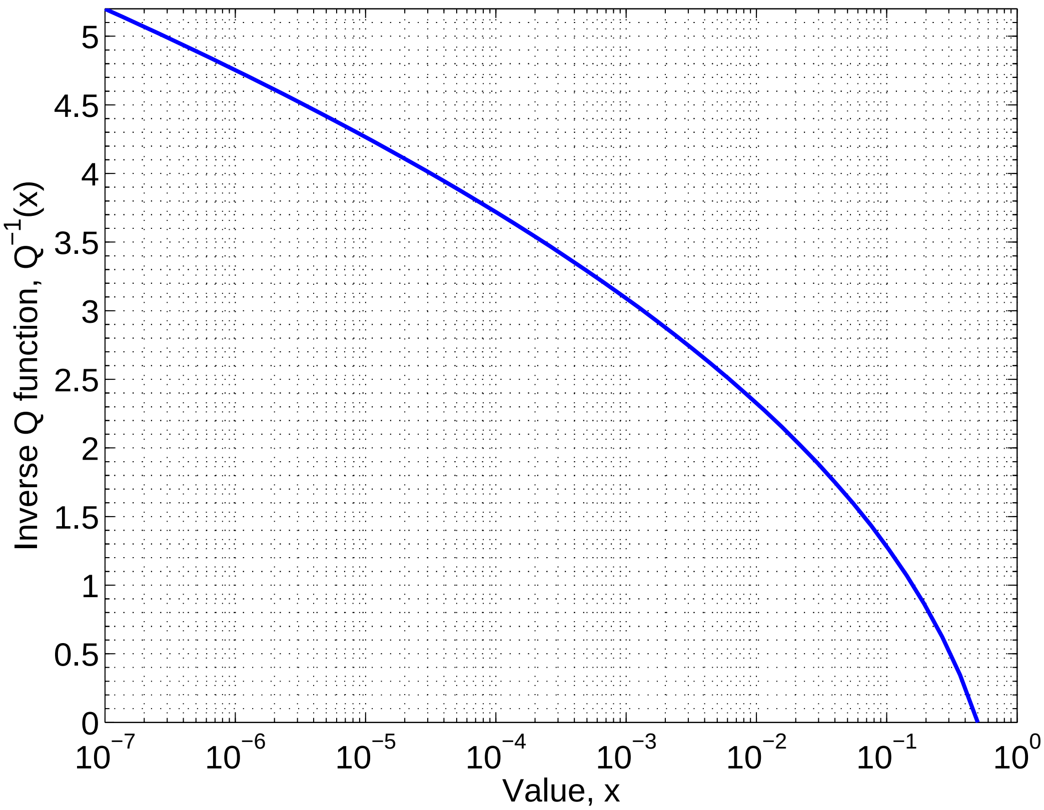
- Be careful of Bytes per sec vs bits per sec. Commonly, CS people use Bps (kBps or MBps) when describing data rate. For example, if it takes 5 seconds to transfer a 1MB file, then software often reports that the data rate is $1/5 = 0.2$ MBps or 200 kBps. But the bit rate is 8/5 Mbps or 1.6×10^6 bps.

Given C/N_0, we can now relate bit error rate, modulation, bit rate, and bandwidth.

Note: We typically use $\mathrm{Q}(\cdot)$ and $\mathrm{Q}^{-1}(\cdot)$ to relate BER and $\frac{\mathcal{E}_b}{N_0}$ in each direction. While you have Matlab, this is easy to calculate. If you can program it into your calculator, great. Otherwise, it's really not a big deal to pull it off of a chart or table. For your convenience, the following tables/plots of $\mathrm{Q}^{-1}(x)$ will appear on Exam 2. I am not picky about getting lots of correct decimal places.

TABLE OF THE $\mathrm{Q}^{-1}(\cdot)$ FUNCTION:

$\mathrm{Q}^{-1}(1\times10^{-6}) = 4.7534$	$\mathrm{Q}^{-1}(1\times10^{-4}) = 3.719$	$\mathrm{Q}^{-1}(1\times10^{-2}) = 2.3263$
$\mathrm{Q}^{-1}(1.5\times10^{-6}) = 4.6708$	$\mathrm{Q}^{-1}(1.5\times10^{-4}) = 3.6153$	$\mathrm{Q}^{-1}(1.5\times10^{-2}) = 2.1701$
$\mathrm{Q}^{-1}(2\times10^{-6}) = 4.6114$	$\mathrm{Q}^{-1}(2\times10^{-4}) = 3.5401$	$\mathrm{Q}^{-1}(2\times10^{-2}) = 2.0537$
$\mathrm{Q}^{-1}(3\times10^{-6}) = 4.5264$	$\mathrm{Q}^{-1}(3\times10^{-4}) = 3.4316$	$\mathrm{Q}^{-1}(3\times10^{-2}) = 1.8808$
$\mathrm{Q}^{-1}(4\times10^{-6}) = 4.4652$	$\mathrm{Q}^{-1}(4\times10^{-4}) = 3.3528$	$\mathrm{Q}^{-1}(4\times10^{-2}) = 1.7507$
$\mathrm{Q}^{-1}(5\times10^{-6}) = 4.4172$	$\mathrm{Q}^{-1}(5\times10^{-4}) = 3.2905$	$\mathrm{Q}^{-1}(5\times10^{-2}) = 1.6449$
$\mathrm{Q}^{-1}(6\times10^{-6}) = 4.3776$	$\mathrm{Q}^{-1}(6\times10^{-4}) = 3.2389$	$\mathrm{Q}^{-1}(6\times10^{-2}) = 1.5548$
$\mathrm{Q}^{-1}(7\times10^{-6}) = 4.3439$	$\mathrm{Q}^{-1}(7\times10^{-4}) = 3.1947$	$\mathrm{Q}^{-1}(7\times10^{-2}) = 1.4758$
$\mathrm{Q}^{-1}(8\times10^{-6}) = 4.3145$	$\mathrm{Q}^{-1}(8\times10^{-4}) = 3.1559$	$\mathrm{Q}^{-1}(8\times10^{-2}) = 1.4051$
$\mathrm{Q}^{-1}(9\times10^{-6}) = 4.2884$	$\mathrm{Q}^{-1}(9\times10^{-4}) = 3.1214$	$\mathrm{Q}^{-1}(9\times10^{-2}) = 1.3408$
$\mathrm{Q}^{-1}(1\times10^{-5}) = 4.2649$	$\mathrm{Q}^{-1}(1\times10^{-3}) = 3.0902$	$\mathrm{Q}^{-1}(1\times10^{-1}) = 1.2816$
$\mathrm{Q}^{-1}(1.5\times10^{-5}) = 4.1735$	$\mathrm{Q}^{-1}(1.5\times10^{-3}) = 2.9677$	$\mathrm{Q}^{-1}(1.5\times10^{-1}) = 1.0364$
$\mathrm{Q}^{-1}(2\times10^{-5}) = 4.1075$	$\mathrm{Q}^{-1}(2\times10^{-3}) = 2.8782$	$\mathrm{Q}^{-1}(2\times10^{-1}) = 0.84162$
$\mathrm{Q}^{-1}(3\times10^{-5}) = 4.0128$	$\mathrm{Q}^{-1}(3\times10^{-3}) = 2.7478$	$\mathrm{Q}^{-1}(3\times10^{-1}) = 0.5244$
$\mathrm{Q}^{-1}(4\times10^{-5}) = 3.9444$	$\mathrm{Q}^{-1}(4\times10^{-3}) = 2.6521$	$\mathrm{Q}^{-1}(4\times10^{-1}) = 0.25335$
$\mathrm{Q}^{-1}(5\times10^{-5}) = 3.8906$	$\mathrm{Q}^{-1}(5\times10^{-3}) = 2.5758$	$\mathrm{Q}^{-1}(5\times10^{-1}) = 0$
$\mathrm{Q}^{-1}(6\times10^{-5}) = 3.8461$	$\mathrm{Q}^{-1}(6\times10^{-3}) = 2.5121$	
$\mathrm{Q}^{-1}(7\times10^{-5}) = 3.8082$	$\mathrm{Q}^{-1}(7\times10^{-3}) = 2.4573$	
$\mathrm{Q}^{-1}(8\times10^{-5}) = 3.775$	$\mathrm{Q}^{-1}(8\times10^{-3}) = 2.4089$	
$\mathrm{Q}^{-1}(9\times10^{-5}) = 3.7455$	$\mathrm{Q}^{-1}(9\times10^{-3}) = 2.3656$	



30.2 Power and Energy Limited Channels

Assume the C/N_0, the maximum bandwidth, and the maximum BER are all given. Sometimes power is the limiting factor in determining the maximum achievable bit rate. Such links (or channels) are called *power limited* channels. Sometimes bandwidth is the limiting factor in determining the maximum achievable bit rate. In this case, the link (or channel) is called a *bandwidth limited* channel. You just need to try to solve the problem and see which one limits your system.

Here is a step-by-step version of what you might need do in this case:

Method A: Start with power-limited assumption:

1. Use the probability of error constraint to determine the $\frac{\mathcal{E}_b}{N_0}$ constraint, given the appropriate probability of error formula for the modulation.

2. Given the C/N_0 constraint and the $\frac{\mathcal{E}_b}{N_0}$ constraint, find the maximum bit rate. Note that $R_b = 1/T_b = \frac{C/N_0}{\frac{\mathcal{E}_b}{N_0}}$, but be sure to express both in linear units.

3. Given a maximum bit rate, calculate the maximum symbol rate $R_s = \frac{R_b}{\log_2 M}$ and then compute the required bandwidth using the appropriate bandwidth formula.

4. Compare the bandwidth at maximum R_s to the bandwidth constraint: If BW at R_s is too high, then the system is bandwidth limited; reduce your bit rate to conform to the BW constraint. Otherwise, your system is power limited, and your R_b is achievable.

Method B: Start with a bandwidth-limited assumption:

1. Use the bandwidth constraint and the appropriate bandwidth formula to find the maximum symbol rate R_s and then the maximum bit rate R_b.
2. Find the $\frac{\mathcal{E}_b}{N_0}$ at the given bit rate by computing $\frac{\mathcal{E}_b}{N_0} = \frac{C/N_0}{R_b}$. (Again, make sure that everything is in linear units.)
3. Find the probability of error at that $\frac{\mathcal{E}_b}{N_0}$, using the appropriate probability of error formula.
4. If the computed $P[\text{error}]$ is greater than the BER constraint, then your system is power limited. Use the previous method to find the maximum bit rate. Otherwise, your system is bandwidth-limited, and you have found the correct maximum bit rate.

Example: Rice 6.33

Consider a bandpass communications link with a bandwidth of 1.5 MHz and with an available $C/N_0 = 82$ dB Hz. The maximum bit error rate is 10^{-6}.

1. If the modulation is 16-PSK using the SRRC pulse shape with $\alpha = 0.5$, what is the maximum achievable bit rate on the link? Is this a power limited or bandwidth limited channel?
2. If the modulation is square 16-QAM using the SRRC pulse shape with $\alpha = 0.5$, what is the maximum achievable bit rate on this link? Is this a power limited or bandwidth limited channel?

Solution:

1. Try Method A. For $M = 16$ PSK, we can find $\frac{\mathcal{E}_b}{N_0}$ for the maximum BER:

$$\begin{aligned}
10^{-6} &= P[\text{error}] = \frac{2}{\log_2 M} \mathrm{Q}\left(\sqrt{2(\log_2 M)\sin^2(\pi/M)\frac{\mathcal{E}_b}{N_0}}\right) \\
10^{-6} &= \frac{2}{4}\mathrm{Q}\left(\sqrt{2(4)\sin^2(\pi/16)\frac{\mathcal{E}_b}{N_0}}\right) \\
\frac{\mathcal{E}_b}{N_0} &= \frac{1}{8\sin^2(\pi/16)}\left[\mathrm{Q}^{-1}\left(2\times 10^{-6}\right)\right]^2 \\
\frac{\mathcal{E}_b}{N_0} &= 69.84
\end{aligned} \tag{52}$$

Converting C/N_0 to linear, $C/N_0 = 10^{82/10} = 1.585 \times 10^8$. Solving for R_b,

$$R_b = \frac{C/N_0}{\frac{\mathcal{E}_b}{N_0}} = \frac{1.585 \times 10^8}{69.84} = 2.27\times 10^6 = 2.27 \text{ Mbits/s}$$

and thus $R_s = R_b/\log_2 M = 2.27\times 10^6/4 = 5.67\times 10^5$ Msymbols/s. The required bandwidth for this system is

$$B_T = \frac{(1+\alpha)R_b}{\log_2 M} = 1.5(2.27\times 10^6)/4 = 851 \text{ kHz}$$

This is clearly lower than the maximum bandwidth of 1.5 MHz. So, the system is power limited, and can operate with bit rate 2.27 Mbits/s. (If B_T had come out > 1.5 MHz, we would have needed to reduce R_b to meet the bandwidth limit.)

2. Try Method A. For $M = 16$ (square) QAM, we can find $\frac{\mathcal{E}_b}{N_0}$ for the maximum BER:

$$\begin{aligned} 10^{-6} &= P[\text{error}] = \frac{4}{\log_2 M}\frac{(\sqrt{M}-1)}{\sqrt{M}}\mathrm{Q}\left(\sqrt{\frac{3\log_2 M}{M-1}\frac{\mathcal{E}_b}{N_0}}\right) \\ 10^{-6} &= \frac{4}{4}\frac{(4-1)}{4}\mathrm{Q}\left(\sqrt{\frac{3(4)}{15}\frac{\mathcal{E}_b}{N_0}}\right) \\ \frac{\mathcal{E}_b}{N_0} &= \frac{15}{12}\left[\mathrm{Q}^{-1}\left((4/3)\times 10^{-6}\right)\right]^2 \\ \frac{\mathcal{E}_b}{N_0} &= 27.55 \end{aligned} \quad (53)$$

Solving for R_b,

$$R_b = \frac{C/N_0}{\frac{\mathcal{E}_b}{N_0}} = \frac{1.585\times 10^8}{27.55} = 5.75\times 10^6 = 5.75 \text{ Mbits/s}$$

The required bandwidth for this bit rate is:

$$B_T = \frac{(1+\alpha)R_b}{\log_2 M} = 1.5(5.75\times 10^6)/4 = 2.16 \text{ MHz}$$

This is greater than the maximum bandwidth of 1.5 MHz, so we must reduce the bit rate to

$$R_b = \frac{B_T \log_2 M}{1+\alpha} = 1.5 \text{ MHz}\frac{4}{1.5} = 4 \text{ MHz}$$

In summary, we have a bandwidth-limited system with a bit rate of 4 MHz.

30.3 Computing Received Power

We need to design our system for the power which we will receive. How can we calculate how much power will be received? We can do this for both wireless and wired channels. Wireless propagation models are required; this section of the class provides two examples, but there are others.

30.3.1 Free Space

'Free space' is the idealization in which nothing exists except for the transmitter and receiver, and can really only be used for deep space communications. But, this formula serves as a starting point for other radio propagation formulas. In free space, the received power is calculated from the Friis formula,

$$C = P_R = \frac{P_T G_T G_R \lambda^2}{(4\pi R)^2}$$

where

- G_T and G_R are the antenna gains at the transmitter and receiver, respectively.
- λ is the wavelength at signal frequency. For narrowband signals, the wavelength is nearly constant across the bandwidth, so we just use the center frequency f_c. Note that $\lambda = c/f_c$ where $c = 3\times 10^8$ meters per second is the speed of light.

- P_T is the transmit power.

Here, everything is in linear terms. Typically people use decibels to express these numbers, and we will write $[P_R]_{\text{dBm}}$ or $[G_T]_{\text{dB}}$ to denote that they are given by:

$$\begin{aligned} [P_R]_{\text{dBm}} &= 10\log_{10} P_R \\ [G_T]_{\text{dB}} &= 10\log_{10} G_T \end{aligned} \tag{54}$$

In lecture 2, we showed that the Friis formula, given in dB, is

$$[C]_{\text{dBm}} = [G_T]_{\text{dB}} + [G_R]_{\text{dB}} + [P_T]_{\text{dBm}} + 20\log_{10}\frac{\lambda}{4\pi} - 20\log_{10} R$$

This says the received power, C, is linearly proportional to the log of the distance R. The Rice book writes the Friis formula as

$$[C]_{\text{dBm}} = [G_T]_{\text{dB}} + [G_R]_{\text{dB}} + [P_T]_{\text{dBm}} + [L_p]_{\text{dB}}$$

where

$$[L_p]_{\text{dB}} = +20\log_{10}\frac{\lambda}{4\pi} - 20\log_{10} R$$

where $[L_p]_{\text{dB}}$ is called the "channel loss". (This name that Rice chose is unfortunate because if it was positive, it would be a "gain", but it is typically very negative. My opinion is, it should be called "channel gain" and have a negative gain value.)

There are also typically other losses in a transmitter / receiver system; losses in a cable, other imperfections, etc. Rice lumps these in as the term L and writes:

$$[C]_{\text{dBm}} = [G_T]_{\text{dB}} + [G_R]_{\text{dB}} + [P_T]_{\text{dBm}} + [L_p]_{\text{dB}} + [L]_{\text{dB}}$$

30.3.2 Non-free-space Channels

We don't study radio propagation path loss formulas in this course. But, a very short summary is that radio propagation on Earth is different than the Friis formula suggests. Lots of other formulas exist that approximate the received power as a function of distance, antenna heights, type of environment, etc.

For example, whenever path loss is linear with the log of distance,

$$[L_p]_{\text{dB}} = L_0 - 10n\log_{10} R.$$

for some constants n and L_0. Effectively, because of shadowing caused by buildings, trees, etc., the average loss may increase more quickly than $1/R^2$, instead, it may be more like $1/R^n$.

30.3.3 Wired Channels

Typically wired channels are lossy as well, but the loss is modeled as linear in the length of the cable. For example,

$$[C]_{\text{dBm}} = [P_T]_{\text{dBm}} - R\,[L_{1m}]_{\text{dB}}$$

where P_T is the transmit power and R is the cable length in meters, and L_{1m} is the loss per meter.

30.4 Computing Noise Energy

The noise energy N_0 can be calculated as:

$$N_0 = kT_{eq}$$

where $k = 1.38 \times 10^{-23}$ J/K is Boltzmann's constant and T_{eq} is called the equivalent noise temperature in Kelvin. This is a topic covered in another course, and so you are not responsible for that material here. But in short, the equivalent temperature is a function of the receiver design. T_{eq} is always going to be higher than the temperature of your receiver. Basically, all receiver circuits add noise to the received signal. With proper design, T_{eq} can be kept low.

30.5 Examples

Example: Rice 6.36

Consider a "point-to-point" microwave link. (Such links were the key component in the telephone company's long distance network before fiber optic cables were installed.) Both antenna gains are 20 dB and the transmit antenna power is 10 W. The modulation is 51.84 Mbits/sec 256 square QAM with a carrier frequency of 4 GHz. Atmospheric losses are 2 dB and other incidental losses are 2 dB. A pigeon in the line-of-sight path causes an additional 2 dB loss. The receiver has an equivalent noise temperature of 400 K and an im- plementation loss of 1 dB. How far away can the two towers be if the bit error rate is not to exceed 10^{-8}? Include the pigeon.

Neal's hint: Use the dB version of Friis formula and subtract these mentioned dB losses: atmospheric losses, incidental losses, implementation loss, and the pigeon.

Solution: Starting with the modulation, $M = 256$ square QAM ($\log_2 M = 8$, $\sqrt{M} = 16$), to achieve $P[\text{error}] = 10^{-8}$,

$$\begin{aligned}
10^{-8} &= \frac{4}{\log_2 m}\frac{(\sqrt{M}-1)}{\sqrt{M}}\mathrm{Q}\left(\sqrt{\frac{3\log_2 M}{M-1}\frac{\mathcal{E}_b}{N_0}}\right) \\
10^{-8} &= \frac{4}{8}\frac{15}{16}\mathrm{Q}\left(\sqrt{\frac{3(8)}{255}\frac{\mathcal{E}_b}{N_0}}\right) \\
10^{-8} &= \frac{15}{32}\mathrm{Q}\left(\sqrt{\frac{24}{255}\frac{\mathcal{E}_b}{N_0}}\right). \\
\frac{\mathcal{E}_b}{N_0} &= \frac{255}{24}\left[\mathrm{Q}^{-1}\left(\frac{32}{15}10^{-8}\right)\right]^2 = 319.0
\end{aligned}$$

The noise power $N_0 = kT_{eq} = 1.38 \times 10^{-23}$(J/K)400(K) $= 5.52 \times 10^{-21}$ J. So $\mathcal{E}_b = 319.0 \times 5.52 \times 10^{-21} = 1.76 \times 10^{-18}$ J. Since $\mathcal{E}_b = C/R_b$ and the bit rate $R_b = 51.84 \times 10^6$ bits/sec, $C = (51.84 \times 10^6)J(1.76 \times 10^{-18})1/sec = 9.13 \times 10^{-11}$ W, or -100.4 dBW.

Switching to finding an expression for C, the wavelength is $\lambda = 3 \times 10^8\text{m/s}/4 \times 10^9 1/\text{s} = 0.075m$, so:

$$\begin{aligned}
[C]_{\text{dBW}} &= [G_T]_{\text{dB}} + [G_R]_{\text{dB}} + [P_T]_{\text{dBW}} + 20\log_{10}\frac{\lambda}{4\pi} - 20\log_{10} R - 2\text{dB} - 2\text{dB} - 2\text{dB} - 1\text{dB} \\
&= 20\text{dB} + 20\text{dB} + 10\text{dBW} + 20\log_{10}\frac{0.075m}{4\pi} - 20\log_{10} R - 7dB \\
&= -1.48\text{dBW} - 20\log_{10} R \qquad (55)
\end{aligned}$$

Plugging in $[C]_{\text{dBm}} = -100.4$ dBW $= -1.48$dBW $- 20\log_{10} R$ and solving for R, we find $R = 88.3$ km. Thus microwave towers should be placed at most 88.3 km (about 55 miles) apart.

Lecture 20

Today: (1) Timing Synchronization Intro, (2) Interpolation Filters

31 Timing Synchronization

At the receiver, a transmitted signal arrives with an unknown delay τ. The received complex baseband signal (for QAM, PAM, or QPSK) can be written (assuming carrier synchronization) as

$$r(t) = \sum_k \sum_m \mathbf{a}_m(k)\phi_m(t - kT_s - \tau) \tag{56}$$

where $\mathbf{a}_k(m)$ is the mth transmitted symbol.

As a start, let's compare receiver implementations for a (mostly) continuous-time and a (more) discrete-time receiver. Figure 54 has a timing synchronization loop which controls the sampler (the ADC).

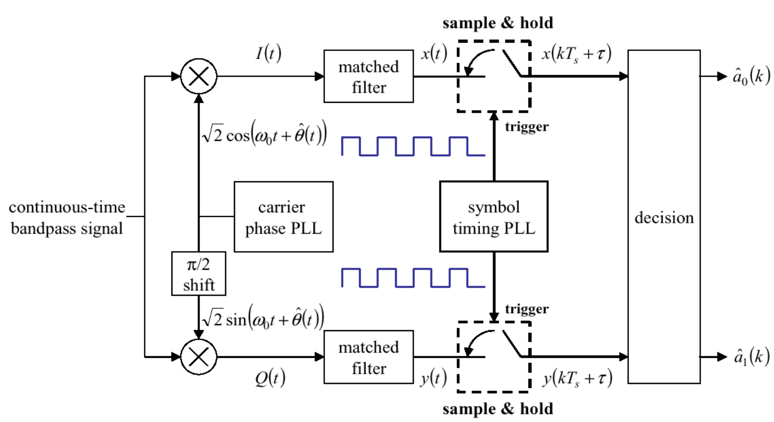


Figure 54: Block diagram for a continuous-time receiver, including analog timing synchronization (from Rice book, Figure 8.3.1).

The input signal is downconverted and then run through matched filters, which correlate the signal with $\phi_n(t - t_k)$ for each n, and for some delay t_k. For the correlation with n,

$$\begin{aligned} x_n(k) &= \langle r(t), \phi_n(t) \rangle \\ x_n(k) &= \sum_k \sum_m \mathbf{a}_m(k) \langle \phi_n(t - t_k), \phi_n(t - kT_s - \tau) \rangle \end{aligned} \tag{57}$$

Note that if $t_k = kT_s + \tau$, then the correlation $\langle \phi_n(t - t_k), \phi_n(t - kT_s - \tau) \rangle$ is highest and closest to 1. This t_k is the correct timing delay at each correlator for the kth symbol. But, these are generally unknown to the receiver until timing synchronization is complete.

Figure 55 shows a receiver with an ADC immediately after the downconverter. Here, note the ADC has nothing controlling it. Instead, after the matched filter, an interpolator corrects the sampling time problems using discrete-time processing. This interpolation is the subject of this section.

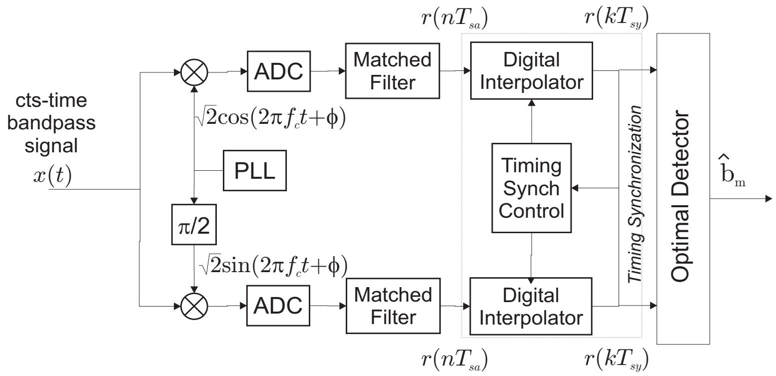


Figure 55: Block diagram for a digital receiver for QAM/PSK, including discrete-time timing synchronization.

The industry trend is more and more towards digital implementations. A 'software radio' follows the idea that as much of the radio is done in digital, after the signal has been sampled. The idea is to "bring the ADC to the antenna" for purposes of reconfigurability, and reducing part counts and costs.

Another implementation is like Figure 55 but instead of the interpolator, the timing synch control block is fed back to the ADC. But again, this requires a DAC and feedback to the analog part of the receiver, which is not preferred. Also, because of the processing delay, this digital and analog feedback loop can be problematic.

First, we'll talk about interpolation, and then, we'll consider the control loop.

32 Interpolation

In this class, we will emphasize digital timing synchronization using an interpolation filter. For example, consider Figure 56. In this figure, a BPSK receiver samples the matched filter output at a rate of twice per symbol, unsynchronized with the symbol clock, resulting in samples $r(nT)$.

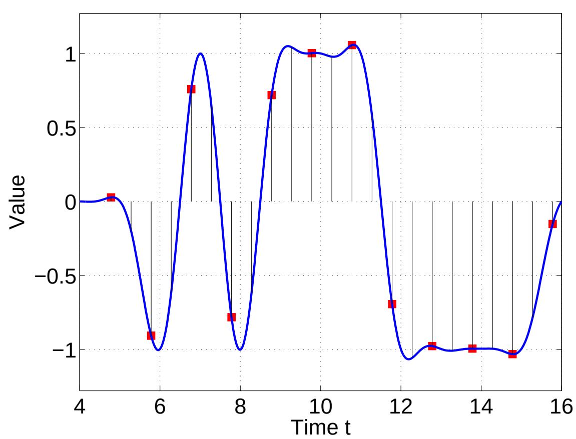


Figure 56: Samples of the matched filter output (BPSK, RRC with $\alpha = 1$) taken at twice the correct symbol rate (vertical lines), but with a timing error. If down-sampling (by 2) results in the symbol samples $\mathbf{r}_k$ given by red squares, then sampling sometimes reduces the magnitude of the desired signal.

Some example sampling clocks, compared to the actual symbol clock, are shown in Figure 57. These are shown in degrees of severity of correction for the receiver. When we say 'synchronized in rate', we mean within an integer multiple, since the sampling clock must operate at (at least) double the symbol rate.

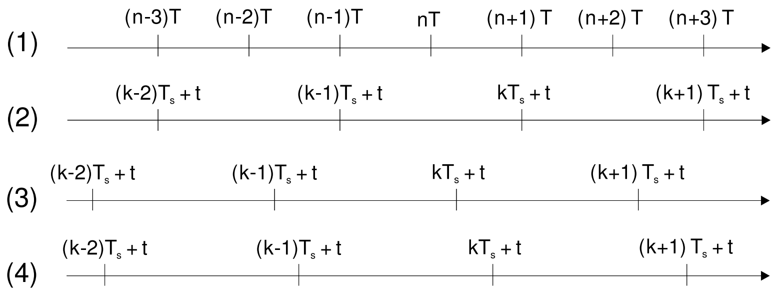


Figure 57: (1) Sampling clock and (2-4) possible actual symbol clocks. Symbol clock may be (2) synchronized in rate and phase, (3) synchronized in rate but not in phase, (4) synchronized neither in rate nor phase, with sample clock.

In general, our receivers always deal with type (4) sampling clock error as drawn in Figure 57. That is, the sampling clock has neither the same exact rate nor the same phase as the actual symbol clock.

Def'n: *Incommensurate*
Two clocks with rates T and T_s are *incommensurate* if the ratio T/T_s is irrational. In contrast, two clocks are commensurate if the ratio T/T_s can be written as n/m where n, m are integers.

For example, $T/T_s = 1/2$, the two clocks are commensurate and we sample exactly twice per symbol period. As another example, if $T/T_s = 2/5$, we sample exactly 2.5 times per symbol, and every 5 samples the delay until the next correct symbol sample will repeat. Since clocks are generally incommensurate, we cannot count on them ever repeating.

The situation shown in Figure 56 is case (3), where $T/T_s = 1/2$ (the clocks are commensurate), but the sampling clock does not have the correct phase (τ is not equal to an integer multiple of T).

32.1 Sampling Time Notation

In general, for both cases (3) and (4) in Figure 57, the correct sampling times should be $kT_s + \tau$, but no samples were taken at those instants. Instead, $kT_s + \tau$ is always $\mu(k)T$ after the most recent sampling instant, where $\mu(k)$ is called the *fractional interval*. We can write that

$$kT_s + \tau = [m(k) + \mu(k)]T \tag{58}$$

where $m(k)$ is an integer, the highest integer such that $nT < kT_s + \tau$, and $0 \leq \mu(k) < 1$. In other words,

$$m(k) = \left\lfloor \frac{kT_s + \tau}{T} \right\rfloor$$

where $\lfloor x \rfloor$ is the greatest integer less than function (the Matlab `floor` function). This means that $\mu(k)$ is given by

$$\mu(k) = \frac{kT_s + \tau}{T} - m(k)$$

Example: Calculation Example
Let $T_s/T = 3.1$ and $\tau/T = 1.8$. Calculate $(m(k), \mu(k))$ for $k = 1, 2, 3$.

Solution:

$$\begin{aligned} m(1) &= \lfloor 3.1 + 1.8 \rfloor = 4; & \mu(1) &= 0.9 \\ m(2) &= \lfloor 2(3.1) + 1.8 \rfloor = 8; & \mu(1) &= 0 \\ m(3) &= \lfloor 3(3.1) + 1.8 \rfloor = 11; & \mu(1) &= 0.1 \end{aligned}$$

Thus your interpolation will be done: in between samples 4 and 5; at sample 8; and in between samples 11 and 12.

32.2 Seeing Interpolation as Filtering

Consider the output of the matched filter, $r(t)$ as given in (57). The analog output of the matched filter could be represented as a function of its samples $r(nT)$,

$$r(t) = \sum_n r(nT) h_I(t - nT) \tag{59}$$

where

$$h_I(t) = \frac{sin(\pi t/T)}{\pi t/T}.$$

Why is this so? What are the conditions necessary for this representation to be accurate?

If we wanted the signal at the correct sampling times, we could have it – we just need to calculate $r(t)$ at another set of times (not nT).

Call the correct symbol sampling times as $kT_s + \tau$ for integer k, where T_s is the actual symbol period used by the transmitter. Plugging these times in for t in (59), we have that

$$r(kT_s + \tau) = \sum_n r(nT) h_I(kT_s + \tau - nT)$$

Now, using the $(m(k), \mu(k))$ notation, since $kT_s + \tau = [m(k) + \mu(k)]T$,

$$r([m(k) + \mu(k)]T) = \sum_n r(nT) h_I([m(k) - n + \mu(k)]T).$$

Re-indexing with $i = m(k) - n$,

$$r([m(k) + \mu(k)]T) = \sum_i r([m(k) - i]T) h_I([i + \mu(k)]T). \tag{60}$$

This is a filter on samples of $r(\cdot)$, where the filter coefficients are dependent on $\mu(k)$.

Note: Good illustrations are given in M. Rice, Figure 8.4.12, and Figure 8.4.13.

32.3 Approximate Interpolation Filters

Clearly, (60) is a filter. The desired sample at $[m(k) + \mu(k)]T$ is calculated by adding the weighted contribution from the signal at each sampling time. The problem is that in general, this requires an infinite sum over i from $-\infty$ to ∞, because the sinc function has infinite support.

Instead, we use polynomial approximations for $h_I(t)$:

- The easiest one we're all familiar with is linear interpolation (a first-order polynomial), in which we draw a straight line between the two nearest sampled values to approximate the values of the continuous signal between the samples. This isn't so great of an approximation.

- A second-order polynomial (*i.e.*a parabola) is actually a very good approximation. Given three points, one can determine a parabola that fits those three points exactly.
- However, the three point fit does *not* result in a linear-phase filter. (To see this, note in the time domain that two samples are on one side of the interpolation point, and one on the other. This is temporal asymmetry.) Instead, we can use four points to fit a second-order polynomial, and get a linear-phase filter.
- Finally, we could use a cubic interpolation filter. Four points determine a 3rd order polynomial, and result in a linear filter.

To see results for different order polynomial filters, see M. Rice Figure 8.24.

32.4 Implementations

Note: These polynomial filters are called *Farrow filters* and are named after Cecil W. Farrow, of AT&T Bell Labs, who has the US patent (1989) for the "Continuously variable digital delay circuit". These Farrow filters started to be used in the 90's and are now very common due to the dominance of digital processing in receivers.

From (60), we can see that the filter coefficients are a function of $\mu(k)$, the fractional interval. Thus we could re-write (60) as

$$r([m(k)+\mu(k)]T) = \sum_i r([m(k)-i]T)h_I(i;\mu(k)). \tag{61}$$

That is, the filter is $h_I(i)$ but its values are a function of $\mu(k)$. The filter coefficients are a polynomial function of $\mu(k)$, that is, they are a weighted sum of $\mu(k)^0, \mu(k)^1, \mu(k)^2, \ldots, \mu(k)^p$ for a pth order polynomial filter.

Example: First order polynomial interpolation
For example, consider the linear interpolator.

$$r([m(k)+\mu(k)]T) = \sum_{i=-1}^{0} r([m(k)-i]T)h_I(i;\mu(k))$$

What are the filter elements h_I for a linear interpolation filter?

Solution:

$$r([m(k)+\mu(k)]T) = \mu(k)r([m(k)+1]T) + [1-\mu(k)]r(m(k)T)$$

Here we have used $h_I(-1;\mu(k)) = \mu(k)$ and $h_I(0;\mu(k)) = 1-\mu(k)$.

Essentially, given $\mu(k)$, we form a weighted average of the two nearest samples. As $\mu(k) \to 1$, we should take the $r([m(k)+1]T)$ sample exactly. As $\mu(k) \to 0$, we should take the $r(m(k)T)$ sample exactly.

32.4.1 Higher order polynomial interpolation filters

In general,

$$h_I(i;\mu(k)) = \sum_{l=0}^{p} b_l(i)\mu(k)^l$$

A full table of $b_l(i)$ is given in Table 8.1 of the M. Rice handout.
Note that the i indices seem backwards.

For the 2nd order Farrow filter, there is an extra degree of freedom – you can select parameter α to be in the range $0 < \alpha < 1$. It has been shown by simulation that $\alpha = 0.43$ is best, but people tend to use $\alpha = 0.5$ because it is only slightly worse, and division by two is extremely easy in digital filters.

Example: 2nd order Farrow filter
What is the Farrow filter for $\alpha = 0.5$ which interpolates exactly half-way between sample points?

Solution: From the problem statement, $\mu = 0.5$. Since $\mu^2 = 0.25, \mu = 0.5, \mu^0 = 1$, we can calculate that

$$\begin{aligned} h_I(-2; 0.5) &= \alpha\mu^2 - \alpha\mu = 0.125 - 0.25 = -0.125 \\ h_I(-1; 0.5) &= -\alpha\mu^2 + (1+\alpha)\mu = -0.125 + 0.75 = 0.625 \\ h_I(0; 0.5) &= -\alpha\mu^2 + (\alpha - 1)\mu + 1 = -0.125 - 0.25 + 1 = 0.625 \\ h_I(1; 0.5) &= \alpha\mu^2 - \alpha\mu = 0.125 - 0.25 = -0.125 \end{aligned} \quad (62)$$

Does this make sense? Do the weights add up to 1? Does it make sense to subtract a fraction of the two more distant samples?

Example: Matlab implementation of Farrow Filter

My implementation is called `ece5520_lec20.m` and is posted on WebCT. Be careful, as my implementation uses a loop, rather than vector processing.

Lecture 21

Today: (1) Timing Synchronization (2) PLLs

33 Final Project Overview

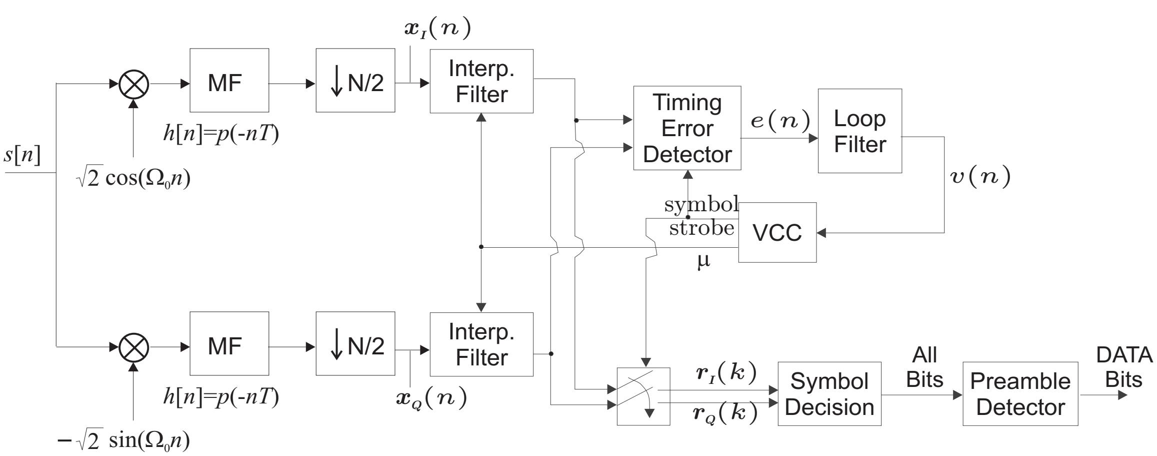


Figure 58: Flow chart of final project; timing synchronization for QPSK.

For your final project, you will implement the above receiver. It adds these blocks to your QPSK implementation:

1. Interpolation Filter
2. Timing Error Detector (TED)
3. Loop Filter
4. Voltage Controlled Clock (VCC)
5. Preamble Detector

These notes address the motivation and overall design of each block.

Compared to last year's class, you have the advantage of having access to the the Rice book, Section 8.4.4, which contains much of the code for a BPSK implementation. When you put these together and implement them for QPSK, you will need to understand how and why they work, in order to fix any bugs.

33.1 Review of Interpolation Filters

Timing synchronization is necessary to know when to sample the matched filter output. We want to sample at times $[n + \mu]T_s$, where n is the integer part and μ is the fractional offset. Often, we leave out the T_s and simply talk about the index n or fractional delay μ.

Implementations may be continuous time, discrete time, or a mix. We focus on the discrete time solutions.

- Problem: After the matched filter, the samples may be at incorrect times, and in modern discrete-time implementations, there may be no analog feedback to the ADC.
- Solution: From samples taken at or above the Nyquist rate, you can interpolate between samples to find the desired sample.

However this solution leads to new problems:

- Problem: True interpolation requires significant calculation – the sinc filter has infinite impulse response.
- Solution: Approximate the sinc function with a 2nd or 3rd order polynomial interpolation filter, it works nearly as well.
- Problem: How do you know when the correct symbol sampling time should be?
- Solution: Use a *timing locked loop*, analogous to a phased locked loop.

33.2 Timing Error Detection

We consider timing error detection blocks that operate after the matched filter and interpolator in the receiver, which has output denoted $x_I(n)$. The job of the timing error detector is to produce a voltage error signal $e(n)$ which is proportional to the correct fractional offset between the current sampling offset ($\hat{\mu}$) and the correct sampling offset.

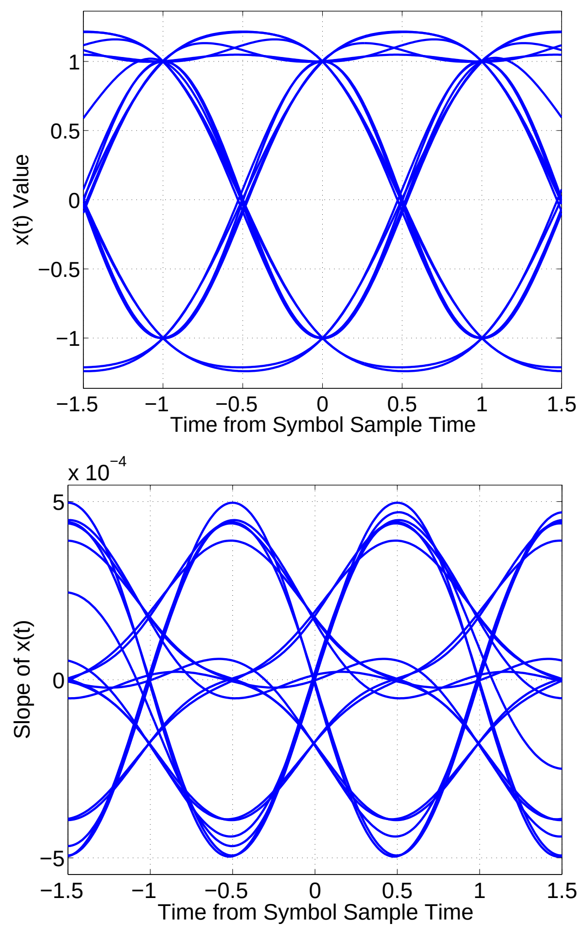
There are several possible timing error detection (TED) methods, as related in Rice 8.4.1. This is a good source for detail on many possible methods. We will talk through two common discrete-time implementations:

1. Early-late timing error detector (ELTED)

2. Zero-crossing timing error detector (ZCTED)

We'll use a continuous-time realization of a BPSK received matched filter output, $x(t)$, to illustrate the operation of timing error detectors. You can imagine that the interpolator output $x_I(n)$ is a sampled version of $x(t)$, hopefully with some samples taken at exactly the correct symbol sampling times.

Figure 59(a) shows an example eye diagram of the signal $x(t)$ and Figure 59(b) shows its derivative $\dot{x}(t)$. In both, time 0 corresponds to a correct symbol sampling time.



(a)

(b)

Figure 59: A sample eye-diagram of a RRC-shaped BPSK received signal (post matched filter), $x(t)$. Here (a) shows the signal $x(t)$ and (b) shows its derivative $\dot{x}(t)$.

33.3 Early-late timing error detector (ELTED)

The early-late TED is a discrete-time implementation of the continuous-time "early-late gate" timing error detector. In general, the error $e(n)$ is determined by the slope and the sign of the sample $x(n)$, at time n. In particular, consider for BPSK the value of

$$\dot{x}(t)\text{sgn}\{x(t)\}$$

Since the derivative $\dot{x}(t)$ is close to zero at the correct sampling time, and increases away from the correct sampling time, we can use it to indicate how far from the sampling time we might be.

Figure 59 shows that the derivative is a good indication of timing error when the bit is changing from -1, to 1, to -1. When the bit is constant (*e.g.*, 1, to 1, to 1) the slope is close to zero. When the bit sequence is from 1, to 1, to -1, the slope will be somewhat negative even when the sample is taken at the right time, and when the bit is changing from -1, to -1, to 1, the slope will be somewhat positive even when the sample is taken at the right time. These are imperfections in the ELTED which (hopefully) average out over many bits. In particular, we can send alternating bits during the synchronization period in order to help the ELTED more quickly converge.

The signum function $\text{sgn}\{x(t)\}$ just corrects for the sign:

- A positive slope would mean we're behind for a +1 symbol, but would mean that we're ahead for a -1 symbol.
- In the opposite manner, a negative slope would mean we're ahead for a +1 symbol, but would mean that we're behind for a -1 symbol.

For a discrete time system, you get only an approximation of the slope unless the sampling rate SPS (or N, as it is called in the Rice book) is high. We'd estimate $\dot{x}(t)$ from the samples out of the interpolator and see that

$$e(n-1) = \{x_I(n) - x_I(n-2)\}\,\text{sgn}\{x_I(n-1)\}$$

Here, we don't divide by the time duration $2T_s$ because it is just a scale factor, and we just need something *proportional to* the timing error. This factor contributes to the gain K_p of this part in the loop. This timing error detector has a theoretical gain K_p which shown in Figure 8.4.7 of the Rice book.

33.4 Zero-crossing timing error detector (ZCTED)

The zero-crossing timing error detector (ZCTED) is also described in detail in Section 8.4.1 of the Rice book. In particular see Figure 8.4.8 of the Rice book. This error detector assumes that the sampling rate is $SPS = 2$. It is described here for BPSK systems.

The error is,

$$e(k) = x((k-1/2)T_s + \hat{\tau})[\hat{a}(k-1) - \hat{a}(k)] \tag{63}$$

where the $\hat{a}(k)$ term is the estimate of the kth symbol (either $+1$ or -1),

$$\hat{a}(k-1) = \text{sgn}\{x((k-1)T_s + \hat{\tau})\}, \tag{64}$$

$$\hat{a}(k) = \text{sgn}\{x(kT_s + \hat{\tau})\}. \tag{65}$$

The error detector is non-zero at symbol sampling times. That is, if the symbol strobe is not activated at sample n, then the error $e(n) = 0$.

In terms of n, because every second sample is a symbol sampling time, (63) can be re-written as

$$e(n) = x_I(n-1)[\text{sgn}\{x_I(n-2)\} - \text{sgn}\{x_I(n)\}] \tag{66}$$

This operation is drawn in Figure 60.

Basically, if the sign changes between $n-2$ and n, that indicates a symbol transition. If there was a symbol transmission, the intermediate sample $n-1$ should be approximately zero, if it was taken at the correct sampling time.

The theoretical gain in the ZCTED is twice that of the ELTED. The factor of two comes from the difference of signs in (66), which will be 2 when the sign changes. In the project, you will need to look up K_p in Figure 8.17 of Rice, which is a function of the excess bandwidth of the RRC filter used, and then multiply it by 2 to find the gain K_p of your ZCTED.

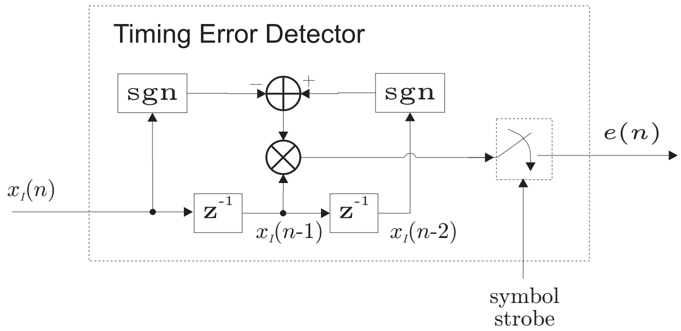


Figure 60: Block diagram of the zero-crossing timing error detector (ZCTED), where sgn indicates the signum function, 1 when positive and -1 when negative. The input strobe signal activates the sampler when it is the correct symbol sampling time. When this happens, and the symbol has switched, the output is $2x_I(n-1)$, which would be approximately zero if the sample clock is synchronized.

33.4.1 QPSK Timing Error Detection

When using QPSK, both the in-phase and quadrature signals are used to calculate the error term. The error is simply the sum of the two errors calculated for each signal. See (8.100) in the Rice book for the exact expression.

33.5 Voltage Controlled Clock (VCC)

We'll use a decrementing VCC in the project. This is shown in Figure 61. (The choice of incrementing or decrementing is arbitrary.) An example trace of the numerically controlled oscillator (NCO) which is the heart of the VCC is shown in Figure 62.

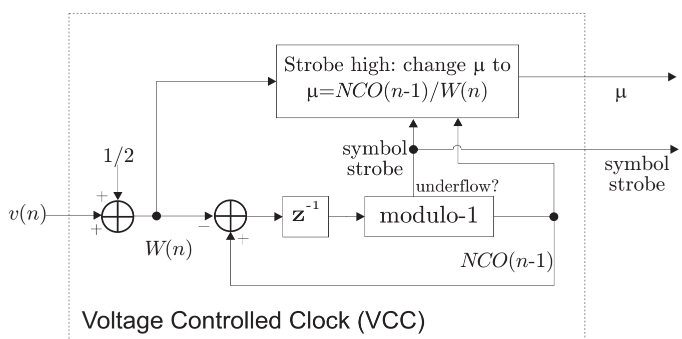


Figure 61: Block diagram of decrementing VCC with control voltage $v(n)$.

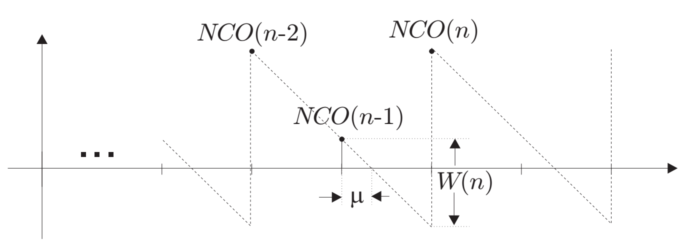


Figure 62: Numerically-controlled oscillator signal $NCO(n)$.

The NCO starts at 1. At each sample n, the NCO decrements by $1/SPS + v(n)$. Once per symbol, (on average) the NCO voltage drops below zero. When this happens, the strobe is set to high, meaning that a sample must be taken.

When the symbol strobe is activated, the fractional interval is also recalculated. It is,

$$\mu(n) = \frac{NCO(n-1)}{W(n)} \tag{67}$$

When the strobe is not activated, $\mu(n)$ is kept the same, *i.e.*, $\mu(n) = \mu(n-1)$. You will prove that (67) is the correct form for $\mu(n)$ in the HW 9 assignment.

33.6 Phase Locked Loops

Carrier synchronization is done using a phase-locked loop (PLL). This section is included for reference. For those of you familiar with PLLs, the operation of the timing synchronization loop may be best understood by analogy to the continuous time carrier synchronization loop.

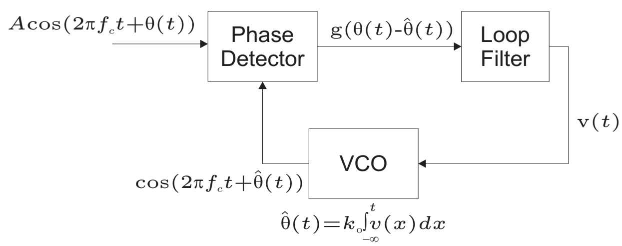


Figure 63: General block diagram of a phase-locked loop for carrier synchronization [Rice].

Consider a generic PLL block diagram in Figure 63. The incoming signal is a carrier wave,

$$A\cos(2\pi f_c t + \theta(t))$$

where $\theta(t)$ is the phase offset. It is a function of time, t, because it may not be constant. For example, it may include a frequency offset, and then $\theta(t) = \Delta f t + \alpha$. In this case, the phase offset is a *ramp*.

The job of the the PLL is to estimate $\phi(t)$. We call this estimate $\hat{\theta}(t)$. The error in the phase estimate is

$$\theta_e(t) = \theta(t) - \hat{\theta}(t)$$

If the PLL is perfect, $\theta_e(t) = 0$.

33.6.1 Phase Detector

If the estimate of the phase is not perfect, the phase detector produces a voltage to correct the phase estimate. The function $g(\theta_e)$ may look something like the one shown in Figure 64.

Initially, let's ignore the loop filter.

1. If the phase estimate is too low, that is, *behind* the carrier, then $g(\theta_e) > 0$ and the VCO increases the phase of the VCO output.

2. If the phase estimate is too high, that is, *ahead of* the carrier, then $g(\theta_e) < 0$ and the VCO decreases the phase of the VCO output.

3. If the phase estimate is exactly correct, then $g(\theta_e) = 0$ and VCO output phase remains constant.

This is the effect of the 'S' curve function $g(\theta_e)$ in Figure 64.

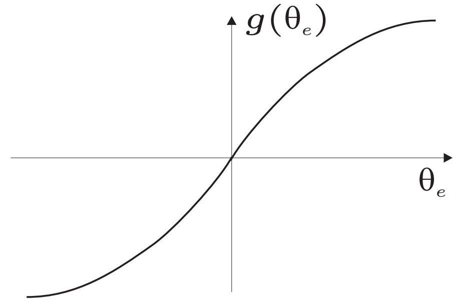


Figure 64: Typical phase detector input-output relationship. This curve is called an 'S' Curve. because the shape of the curve looks like the letter 'S' [Rice].

33.6.2 Loop Filter

The loop filter is just a low-pass filter to reduce the noise. Note that in addition to the $A\cos(2\pi f_c t + \theta(t))$ signal term we also have channel noise coming into the system. We represent the loop filter in the Laplace or frequency domain as $F(s)$ or $F(f)$, respectively, for continuous time. We will be interested in discrete time later, so we could also write the Z-transform response as $F(z)$.

33.6.3 VCO

A voltage-controlled oscillator (VCO) simply integrates the input and uses that integral as the phase of a cosine wave. The input is essentially the gas pedal for a race car driving around a track - increase the gas (voltage), and the engine (VCO) will increase the frequency of rotation around the track.

Note that if you just raise the voltage for a short period and then reduce it back (a constant plus rect function input), you change the phase of the output, but leave the frequency the same.

33.6.4 Analysis

To analyze the loop in Figure 63, we end up making simplifications. In particular, we model the 'S' curve (Figure 64) as a line. This linear approximation is generally fine when the phase error is reasonably small, *i.e.*, $\theta_e = 0$. If we do this, we can re-draw Figure 63 as it is drawn in Figure 65.

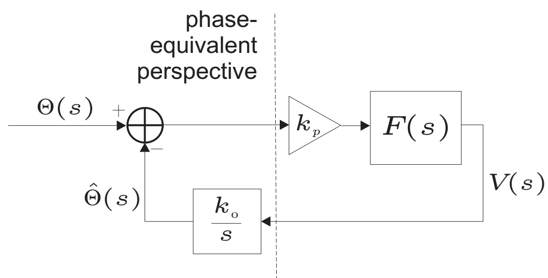


Figure 65: Phase-equivalent and linear model for the continuous-time PLL, in the Laplace domain [Rice].

In the left half of Figure 65, we write just $\Theta(s)$ and $\hat{\Theta}(s)$ even though the physical process, as we know, is the cosine of $2\pi f_c$ *plus* that phase. This phase-equivalent model of the PLL allows us to do analysis.

We can write that

$$\begin{aligned}\Theta_e(s) &= \Theta(s) - \hat{\Theta}(s) \\ &= \Theta(s) - \Theta_e(s) k_p F(s) \frac{k_0}{s}\end{aligned}$$

To get the transfer function of the filter (when we consider the phase estimate to be the output), then some manipulation shows you that

$$\begin{aligned}H_a(s) &= \frac{\hat{\Theta}(s)}{\Theta(s)} = \frac{k_p F(s) \frac{k_0}{s}}{1 + k_p F(s) \frac{k_0}{s}} \\ &= \frac{k_0 k_p F(s)}{s + k_0 k_p F(s)}\end{aligned}$$

You can use $H_a(s)$ to design the loop filter $F(s)$ and gains k_p and k_0 to achieve the desired PLL goals. Here those goals include:

1. Desired bandwidth.
2. Desired response to particular phase error models.

The latter deserves more explanation. For example, phase error models might be: step input; or ramp input. Designing a filter to respond to these, but not to AWGN noise (as much as possible), is a challenge. The Rice book recommends a 2nd order *proportional-plus-integrator* loop filter,

$$F(s) = k_1 + \frac{k_2}{s}$$

This filter has a proportional part (k_1) which is just proportional to the input, which responds to the input during phase offset. The filter also has an integrator part (k_2/s) which integrates the phase input, and in case of a frequency offset, will ramp up the phase at the proper rate.

Note: A accelerating frequency change wouldn't be handled properly by the given filter. For example, if the temperature of the transmitter or receiver varied quickly, the frequency of the crystal will also change quickly. However, this is not typically fast enough to be a problem in most radios.

In this case there are four parameters to set, k_0, k_1, k_2, and k_p. The Rice Section C.2 details how to set these parameters to achieve a desired bandwidth and a desired damping factor (to avoid ringing, but to converge quickly). You will, in Homework 9, design the particular loop filter to use in your final project.

33.6.5 Discrete-Time Implementations

You can alter $F(s)$ to be a discrete time filter using Tustin's approximation,

$$\frac{1}{s} \rightarrow \frac{T}{2} \frac{1 + z^{-1}}{1 - z^{-1}}$$

Because it is only a second-order filter, its implementation is quite efficient. See Figure C.2.1 (p 733, Rice book) for diagrams of this discrete-time PLL. Equations C.56 and C.57 (p 736) show how to calculate the constants K_1 and K_2, given the desired natural frequency θ_n, the normalized bandwidth $B_n T$, damping coefficient ζ, and other gain constants K_0 and K_p.

Note that the bandwidth $B_n T$ is the most critical factor. It is typically much less than one, (*e.g.*, 0.005). This means that the loop filter effectively averages over many samples when setting the input to the VCO.

Lecture 22

Today: (1) Exam 2 Review

- Exam 2 is Tue Apr 14.
- I will be out of town Mon-Wed. The exam will be proctored. Please come to me with questions today or tomorrow.
- Office hours: Today: 2-3:30. Friday 11-12, 3-4.

34 Exam 2 Topics

Where to study:

- Lectures covered: 8 (detection theory), 9, 11-19. Lectures 20,21 are not covered. No material from 1-7 will be directly tested, but of course you need to know some things from the start of the semester to be able to perform well on the second part of this course.
- Homeworks covered: 4-8. Please review these homeworks and know the correct solutions.
- Rice book: Chapters 5 and 6.

What topics:

1. Detection theory
2. Signal space
3. Digital modulation methods:
   - Binary, bipolar vs. unipolar
   - M-ary: PAM, QAM, PSK, FSK
4. Inter-symbol interference, Nyquist filtering, SRRC pulse shaping
5. Differential encoding and decoding for BPSK
6. Energy detection for FSK
7. Gray encoding
8. Probability of Error vs. $\frac{\mathcal{E}_b}{N_0}$.
   - Standard modulation method. Both know the formula and know how it can be derived.
   - A non-standard modulation method, given signal space diagram.
   - Exact expression, union bound, nearest-neighbor approximation.

9. Bandwidth assuming SRRC pulse shaping for PAM/QAM/PSK, FSK, concept of frequency multiplexing
10. Comparison of digital modulation methods:
    - Need for linear amplifiers (transmitter complexity)
    - Receiver complexity
    - Bandwidth efficiency

Types of questions (format):

1. Short answer, with a 1-2 sentence limit. There may be multiple correct answers, but there may be many wrong answers (or right answer / wrong reason combinations).
2. True / false or multiple choice.
3. Work out solution. Example: What is the probability of bit error vs. $\frac{\mathcal{E}_b}{N_0}$ for this signal space diagram?
4. System design: Here are some engineering requirements for this communication system. From this list of possible modulation, which one would you recommend? What bandwidth and bit rate would it achieve?

You will be provided the table of the Qinv function, and the plot of Qinv. You can use two sides of an 8.5 x 11 sheet of paper.

Lecture 23

Today: (1) Source Coding

- HW 9 is due today at 5pm. OH today 2-3pm, Monday 4-5pm.
- Final homework: HW 10, which will be due April 28 at 5pm.

35 Source Coding

This topic is a semester-long course at the graduate level in itself; but the basic ideas can be presented pretty quickly.

One good reference:

- C. E. Shannon, "A mathematical theory of communication," Bell System Technical Journal, vol. 27, pp. 379-423 and 623-656, July and October, 1948. `http://www.cs.bell-labs.com/cm/ms/what/shannonday/paper.html`

We have done a lot of counting of bits as our primary measure of communication systems. Our information source is measured in bits, or in bits per second. Modulation schemes' bandwidth efficiency is measured in bits per Hertz, and energy efficiency is energy per bit over noise PSD. Everything is measured in bits!

But how do we measure the bits of a source (*e.g.*, audio, video, email, ...)? Information can often be represented in many different ways. Images and sound can be encoded in different ways. Text files can be presented in different ways.

Here are two misconceptions:

1. Using the file size tells you how much information is contained within the file.
2. Take the $\log_2$ of the number of different messages you could send.

For example, consider a digital black & white image (not grayscale, but truly black or white).

1. You could store it as a list of pixels. Each pixel has two possibilities (possible messages), thus we could encode it in $\log_2 2$ or one bit per pixel.
2. You could simply send the coordinates of the pixels of one of the colors (*e.g.*all black pixels).

How many bits would be used in these two representations? What would make you decide which one is more efficient?

You can see that equivalent representations can use different number of bits. This is the idea behind *source compression*. For example, .zip or .tar files represent the exact same information that was contained in the original files, but with fewer bits.

What if we had a fixed number of bits to send any image, and we used the sparse B&W image coding scheme (2.) above? Sometimes, the number of bits in the compressed image would exceed what we had allocated. This would introduce errors into the image.

Two types of compression algorithms:

- Lossless: *e.g.*, Zip or compress.
- Lossy: *e.g.*, JPEG, MP3

Note: Both "zip" and the unix "compress" commands use the Lempel-Ziv algorithm for source compression.

So what is the intrinsic measure of bits of text, an image, audio, or video?

35.1 Entropy

Entropy is a measure of the randomness of a random variable. *Randomness* and *information*, in non-technical language, are just two perspectives on the same thing:

- If you are told the value of a r.v. that doesn't vary that much, that telling conveys very little information to you.
- If you are told the value of a very "random" r.v., that telling conveys quite a bit of information to you.

Our technical definition of entropy of a random variable is as follows.

Def'n: *Entropy*
Let X be a discrete random variable with pmf $p_X(x_i) = P[X = x]$. Here, there is a finite or countably infinite set S_X, and $x \in S_X$. We will shorten the notation by using p_i as follows:

$$p_i = p_X(x_i) = P[X = x_i]$$

where $\{x_1, x_2, \ldots\}$ is an ordering of the possible values in S_X. Then the entropy of X, in units of bits, is defined as,

$$H[X] = -\sum_i p_i \log_2 p_i \tag{68}$$

Notes:

- $H[X]$ is an operator on a random variable, not a function of a random variable. It returns a (deterministic) number, not another random variable. This it is like $E[X]$, another operator on a random variable.
- Entropy of a discrete random variable X is calculated using the probability values of the pmf of X, p_i. Nothing else is needed.
- The sum will be from $i = 1 \ldots N$ when $|S_X| = N < \infty$.
- Use that $0 \log 0 = 0$. This is true in the limit of $x \log x$ as $x \to 0^+$.
- All "log" functions are log-base-2 in information theory unless otherwise noted. Keep this in mind when reading a book on information theory. The "reason" the units are bits is because of the base-2 of the log. Actually, when theorists use $\log_e$ or the natural log, they express information in "nats", short for "natural" digits.

Example: Binary r.v.
A binary (Bernoulli) r.v. has pmf,

$$p_X(x) = \begin{cases} s, & x = 1 \\ 1 - s, & x = 0 \\ 0, & o.w. \end{cases}$$

What is the entropy $H[X]$ as a function of s?

Solution: Entropy is given by (68) and is:

$$H[X] = -s \log_2 s - (1 - s) \log_2 (1 - s)$$

The solution is plotted in Figure 66.

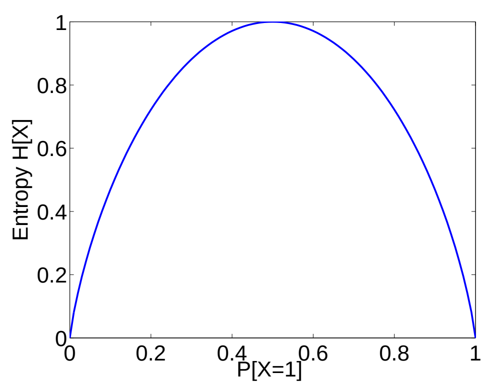


Figure 66: Entropy of a binary r.v.

Example: Non-uniform source with five messages
Some signals are more often close to zero (*e.g.*audio). Model the r.v. X to have pmf

$$p_X(x) = \begin{cases} 1/16, & x = 2 \\ 1/4, & x = 1 \\ 1/2, & x = 0 \\ 1/8, & x = -1 \\ 1/16, & x = -2 \\ 0, & o.w. \end{cases}$$

What is its entropy $H[X]$?

Solution:

$$\begin{aligned} H[X] &= \frac{1}{2}\log_2 2 + \frac{1}{4}\log_2 4 + \frac{1}{8}\log_2 8 + 2\frac{1}{16}\log_2 16 \\ &= \frac{15}{8} \text{ bits} \end{aligned} \tag{69}$$

Other questions:

1. Do you need to know what the symbol set S_X is?
2. Would multiplying X by 2 change its entropy?
3. Would an arbitrary one-to-one function change the entropy of X?

35.2 Joint Entropy

<u>Def'n:</u> *Joint Entropy*
The joint entropy of two random variables X_1, X_2 with event sets S_{X_1} and S_{X_2} is defined as

$$H[X_1, X_2] = -\sum_{x_1 \in S_{X_1}} \sum_{x_2 \in S_{X_2}} p_{X_1,X_2}(x_1, x_2) \log_2 p_{X_1,X_2}(x_1, x_2) \tag{70}$$

For N joint random variables, $X_1, \ldots, X_N$, entropy is

$$H[X_1, \ldots, X_N] = -\sum_{x_1 \in S_{X_1}} \cdots \sum_{x_N \in S_{X_N}} p_{X_1,\ldots,X_N}(x_1, \ldots, x_N) \log_2 p_{X_1,\ldots,X_N}(x_1, \ldots, x_N)$$

What is the entropy for N i.i.d. random variables? You can show that

$$H[X_1, \ldots, X_N] = -N \sum_{x_1 \in S_{X_1}} p_{X_1}(x_1) \log_2 p_{X_1}(x_1) = NH(X_1)$$

The entropy of N i.i.d. random variables has N times the entropy of any one of them. In addition, the entropy of any N independent (but possibly with different distributions) r.v.s is just the sum of the entropy of each individual r.v.

When r.v.s are not independent, the joint entropy of N r.v.s is less than N times the entropy of one of them. Intuitively, if you know some of them, because of the dependence or correlation, the rest that you don't know become less informative. For example, the B&W image, since pixels are correlated in space, the joint r.v. of several neighboring pixels will have less entropy than the sum of the individual pixel entropies.

35.3 Conditional Entropy

How much additional entropy is in the joint random variables X_1, X_2 compared just to one of them? This is often an important question because it answers the question, "How much additional information do I get from both, compared to just one of them?". We call this difference the conditional entropy, $H[X_2|X_1]$:

$$H[X_2|X_1] = H[X_2, X_1] - H[X_1]. \tag{71}$$

What is an equation for $H[X_2|X_1]$ as a function of the joint probabilities $p_{X_1,X_2}(x_1,x_2)$ and the conditional probabilities $p_{X_2|X_1}(x_2|x_1)$.

Solution: Plugging in (68) for $H[X_2,X_1]$ and $H[X_1]$,

$$\begin{aligned}
H[X_2|X_1] &= -\sum_{x_1\in S_{X_1}}\sum_{x_2\in S_{X_2}} p_{X_1,X_2}(x_1,x_2)\log_2 p_{X_1,X_2}(x_1,x_2) + \sum_{x_1\in S_{X_1}} p_{X_1}(x_1)\log_2 p_{X_1}(x_1) \\
&= -\sum_{x_1\in S_{X_1}}\sum_{x_2\in S_{X_2}} p_{X_1,X_2}(x_1,x_2)\log_2 p_{X_1,X_2}(x_1,x_2) + \sum_{x_1\in S_{X_1}}\left[\sum_{x_2\in S_{X_2}} p_{X_1,X_2}(x_1,x_2)\right]\log_2 p_{X_1}(x_1) \\
&= -\sum_{x_1\in S_{X_1}}\sum_{x_2\in S_{X_2}} p_{X_1,X_2}(x_1,x_2)\left(\log_2 p_{X_1,X_2}(x_1,x_2) - \log_2 p_{X_1}(x_1)\right) \\
&= -\sum_{x_1\in S_{X_1}}\sum_{x_2\in S_{X_2}} p_{X_1,X_2}(x_1,x_2)\log_2 \frac{p_{X_1,X_2}(x_1,x_2)}{p_{X_1}(x_1)} \\
&= -\sum_{x_1\in S_{X_1}}\sum_{x_2\in S_{X_2}} p_{X_1,X_2}(x_1,x_2)\log_2 p_{X_2|X_1}(x_2|x_1) \qquad (72)
\end{aligned}$$

Note the asymmetry – there is the joint probability multiplied by the log of the conditional probability. This is not like either the joint or the marginal entropy.

We could also have multi-variate conditional entropy,

$$H[X_N|X_{N-1},\ldots,X_1] = -\sum_{x_{N-1}\in S_{X_{N-1}}}\cdots\sum_{x_1\in S_{X_1}} p_{X_1,\ldots,X_N}(x_1,x_N)\log_2 p_{X_N|X_{N-1},\ldots,X_1}(x_N|x_{N-1},\ldots,x_1)$$

which is the additional entropy (or information) contained in the Nth random variable, given the values of the $N-1$ previous random variables.

35.4 Entropy Rate

Typically, we're interested in discrete-time random processes, in which we have random variables $X_1, X_2, \ldots$. Since there are infinitely many of them, the joint entropy of all of them may go to infinity as $N \to \infty$. For this case, we are more interested in the rate. How many additional bits, in the limit, are needed for the average r.v. as $N \to \infty$?

Def'n: *Entropy Rate*

The entropy rate of a stationary discrete-time random process, in units of bits per random variable (a.k.a. source output), is defined as

$$H = \lim_{N\to\infty} H[X_N|X_{N-1},\ldots,X_1].$$

It can be shown that entropy rate can equivalently be written as

$$H = \lim_{N\to\infty}\frac{1}{N}H[X_1,X_2,\ldots,X_N].$$

Example: Entropy of English text

Let X_i be the ith letter or space in a common English sentence. What is the sample space S_{X_i}? Is X_i uniform on that space?

What is $H[X_i]$? Solution: I had Matlab read in the text of Shakespeare's *Romeo and Juliet*. See Figure 67(a). For this pmf, I calculated an entropy of $H = 4.1199$. The Proakis & Salehi book mentions that this value for general English text is about 4.3.

What is $H[X_i, X_{i+1}]$? Solution: Again, using Matlab on Shakespeare's *Romeo and Juliet*, I calculated the entropy of the joint pmf of each two-letter combination. This gives me the two-dimensional pmf shown in Figure **??**(b). I calculate an entropy of 7.46, which is $2 \cdot 3.73$. For the three-letter combinations, the joint entropy was $10.04 = 3 \cdot 3.35$. For four-letter combinations, the joint entropy was $11.98 = 4 \cdot 2.99$.

You can see that the average entropy in bits per letter is decreasing quickly.

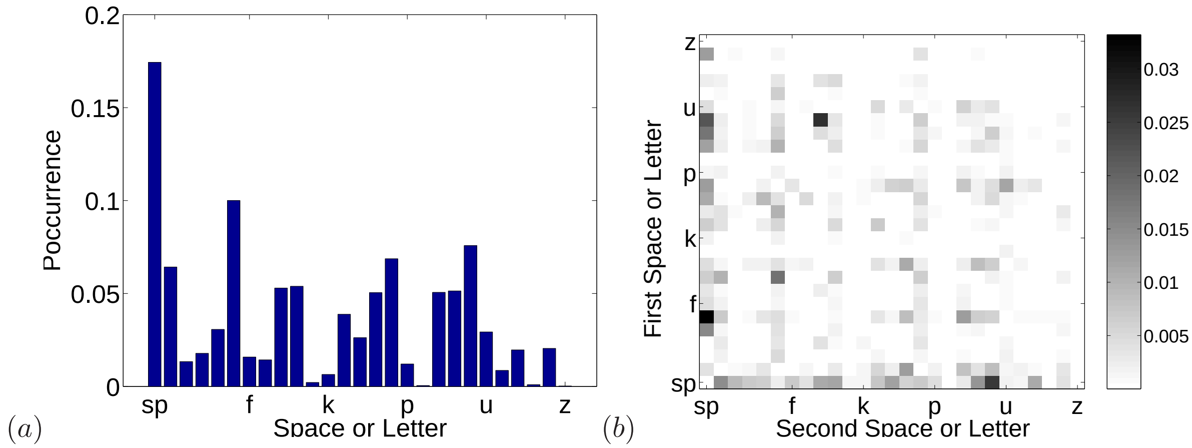


Figure 67: PMF of (a) single letters and (b) two-letter combinations (including spaces) in Shakespeare's *Romeo and Juliet*.

What is the entropy rate, H? Solution: For $N = 10$, we have $H = 1.3$ bits/letter [taken from Proakis & Salehi Section 6.2].

35.5 Source Coding Theorem

The key connection between this mathematical definition of entropy and the bit rate that we've been talking about all semester is given by the source coding theorem. It is one of the two fundamental theorems of information theory, and was introduced by Claude Shannon in 1948.

Theorem: A source with entropy rate H can be encoded with arbitrarily small error probability, at any rate R (bits / source output) as long as $R > H$. Conversely, if $R < H$, the error probability will be bounded away from zero, independent of the complexity of the encoder and the decoder employed.
Proof: Proof: Using *typical sequences*. See Shannon's original 1948 paper.

Notes:

- Here, an 'error' occurs when your compressed version of the data is not exactly the same as the original. Example: B&W images.
- R is our $1/T_b$.
- Theorem fact: Information measure (entropy) gives us a minimum bit rate.
- What is the minimum possible rate to encode English text (if you remove all punctuation)?
- The theorem does not tell us how to do it – just that it can be done.

- The theorem does not tell us how well it can be done if N is not infinite. That is, for a finite source, the rate may need to be higher.

Lecture 24

Today: (1) Channel Capacity

36 Review

Last time, we defined entropy,

$$H[X] = -\sum_i p_i \log_2 p_i$$

and entropy rate,

$$H = \lim_{N\to\infty} \frac{1}{N} H[X_1, X_2, \ldots, X_N].$$

We showed that entropy can be used to quantify information. Given our information source X or $\{X_i\}$, the value of $H[X]$ or H gives us a measure of how many bits we actually need to use to encode, without loss, the source data.

The major result was the Shannon's source coding theorem, which says that a source with entropy rate H can be encoded with arbitrarily small error probability, at any rate R (bits / source output) as long as $R > H$. Any lower rate than H would guarantee loss of information.

37 Channel Coding

Now, we turn to the noisy channel. This discussion of entropy allows us to consider the maximum data rate which can be carried without error on a bandlimited channel, which is affected by additive White Gaussian noise (AWGN).

37.1 R. V. L. Hartley

Ralph V. L. Hartley (born Nov. 30, 1888) received the A.B. degree from the University of Utah in 1909. He worked as a researcher for the Western Electric Company, involved in radio telephony. Here he developed the "Hartley oscillator". Afterwards, at Bell Laboratories, he developed relationships useful for determining the capacity of bandlimited communication channels. In July 1928, he published in the Bell System Technical Journal a paper on "Transmission of Information".

Hartley was particularly influenced by Nyquist's result. When transmitting a sequence of pulses, each of duration T_{sy}, Nyquist determined that the pulse rate was limited to two times the available channel bandwidth B,

$$\frac{1}{T_{sy}} \leq 2B.$$

In Hartley's 1928 paper, he considered digital transmission in pulse-amplitude modulated systems. The pulse rate was limited to $2B$, as described by Nyquist. But, depending on how pulse amplitudes were chosen, each pulse could represent more or less information.

In particular, Hartley assumed that the maximum amplitude available to the transmitter was A. Then, Hartley made the assumption that the communication system could discern between pulse amplitudes, if they were at separated by at least a voltage spacing of A_δ. Given that a PAM system operates from 0 to A in increments of A_δ, the number of different pulse amplitudes (symbols) is

$$M = 1 + \frac{A}{A_\delta}$$

Note that early receivers were modified AM envelope detectors, and did not deal well with negative amplitudes.

Next, Hartley used the 'bit' measure to quantify the data which could be encoded using M amplitude levels,

$$\log_2 M = \log_2 \left(1 + \frac{A}{A_\delta}\right)$$

Finally, Hartley quantified the data rate using Nyquist's relationship to determine the maximum rate C, in bits per second, possible from the digital communication system,

$$C = 2B \log_2 \left(1 + \frac{A}{A_\delta}\right)$$

(Note that the Hartley transform came later in his career in 1942).

37.2 C. E. Shannon

What was left unanswered by Hartley's capacity formula was the relationship between noise and the minimum amplitude separation between symbols. Engineers would have to be conservative when setting A_δ to ensure a low probability of error.

Furthermore, the capacity formula was for a particular type of PAM system, and did not say anything fundamental about the relationship between capacity and bandwidth for arbitrary modulation.

37.2.1 Noisy Channel

Shannon did take into account an AWGN channel, and used statistics to develop a universal bound for capacity, regardless of modulation type. In this AWGN channel, the ith symbol sample at the receiver (after the matched filter, assuming perfect syncronization) is y_i,

$$y_i = x_i + z_i$$

where X is the transmitted signal and Z is the noise in the channel. The noise term z_i is assumed to be i.i.d. Gaussian with variance P_N.

37.2.2 Introduction of Latency

Shannon's key insight was to exchange latency (time delay) for reduced probability of error. In fact, his capacity bound considers simultaneously demodulating sequences of received symbols, $\mathbf{y} = [y_1, \ldots, y_n]$, of length n. All n symbols are received before making a decision. This late decision will decide all values of $\mathbf{y} = [x_1, \ldots, x_n]$ simultaneously. Further, Shannon's proof considers the limiting case as $n \to \infty$.

This asymptotic limit as $n \to \infty$ allows for a proof using the statistical convergence of a sequence of random variables. In particular, we need a law called *the law of large numbers* (ECE 6962 topic). This law says that the following event,

$$\frac{1}{n} \sum_{i=1}^{n} (y_i - x_i)^2 \leq P_N$$

happens with probability one, as $n \to \infty$. In other words, as $n \to \infty$, the measured value $\mathbf{y}$ will be located within an n-dimensional sphere (hypersphere) of radius $\sqrt{nP_N}$ with center $\mathbf{x}$.

37.2.3 Introduction of Power Limitation

Shannon also formulated the problem as a power-limited case, in which the average power in the desired signal x_i was limited to P. That is,

$$\frac{1}{n}\sum_{i=1}^{n} x_i^2 \leq P$$

This combination of signal power limitation and noise power results in the fact that,

$$\frac{1}{n}\sum_{i=1}^{n} y_i^2 \leq \frac{1}{n}\sum_{i=1}^{n} x_i^2 + \frac{1}{n}\sum_{i=1}^{n}(y_i - x_i)^2 \leq P + P_N$$

As a result

$$\|\mathbf{y}\|^2 \leq n(P + P_N)$$

This result says that the vector $\mathbf{y}$, with probability one as $n \to \infty$, is contained within a hypersphere of radius $\sqrt{n(P + P_N)}$ centered at the origin.

37.3 Combining Two Results

The two results, together, show how we many different symbols we could have uniquely distinguished, within a period of n sample times. Hartley asked how many symbol amplitudes could be fit into $[0, A]$ such that they are all separated by A_δ. Shannon's formulation asks us how many multidimensional amplitudes $\mathbf{x}_i$ can be fit into a hypersphere of radius $\sqrt{n(P + P_N)}$ centered at the origin, such that hyperspheres of radius $\sqrt{nP_N}$ do not overlap. This is shown in P&S in Figure 9.9 (pg. 584).

This number M is the number of different messages that could have been sent in n pulses. The result of this geometrical problem is that

$$M = \left(1 + \frac{P}{P_N}\right)^(n/2) \tag{73}$$

37.3.1 Returning to Hartley

Adjusting Hartley's formula, if we could send M messages now in n pulses (rather than 1 pulse) we would adjust capacity to be:

$$C = \frac{2B}{n}\log_2 M$$

Using the M from (76) above,

$$\begin{aligned} C &= \frac{2B}{n}\frac{n}{2}\log_2\left(1 + \frac{P}{P_N}\right) \\ &= B\log_2\left(1 + \frac{P}{P_N}\right) \end{aligned}$$

37.3.2 Final Results

Finally, we can replace the noise variance P_N with the relationship between noise two-sided PSD $N_0/2$, by

$$P_N = N_0 B$$

So finally we have the Shannon-Hartley Theorem,

$$C = B \log_2 \left(1 + \frac{P}{N_0 B}\right) \tag{74}$$

Typically, people also use $W = B$ so you'll see also

$$C = W \log_2 \left(1 + \frac{P}{N_0 W}\right) \tag{75}$$

This result says that **a communication system can operate at bit rate** C (in a bandlimited channel with width W given power limit P and noise value N_0), **with arbitrarily low probability of error.**

Shannon also proved that any system which operates at a bit rate higher than the capacity C will certainly incur a positive bit error rate. Any practical communication system must operate at $R_b < C$, where R_b is the operating bit rate.

Note that the ratio $\frac{P}{N_0 W}$ is the signal power divided by the noise power, or signal to noise ratio (SNR). Thus the capacity bound is also written $C = W \log_2(1 + SNR)$.

37.4 Efficiency Bound

Recall that $\mathcal{E}_b = PT_b$. That is, the energy per bit is the power multiplied by the bit duration. Thus from (74),

$$C = W \log_2 \left(1 + \frac{\mathcal{E}_b/T_b}{N_0 W}\right)$$

or since $R_b = 1/T_b$,

$$C = W \log_2 \left(1 + \frac{R_b}{W}\frac{\mathcal{E}_b}{N_0}\right)$$

Here, C is just a capacity limit. We know that our bit rate $R_b \le C$, so

$$\frac{R_b}{W} \le \log_2 \left(1 + \frac{R_b}{W}\frac{\mathcal{E}_b}{N_0}\right)$$

Defining $\eta = \frac{R_b}{W}$,

$$\eta \le \log_2 \left(1 + \eta\frac{\mathcal{E}_b}{N_0}\right)$$

This expression can't analytically be solved for η. However, you can look at it as a bound on the bandwidth efficiency as a function of the $\frac{\mathcal{E}_b}{N_0}$ ratio. This relationship is shown in Figure 70.

Lecture 25

Today: (1) Channel Capacity

- Lecture Tue April 28, is canceled. Instead, please the Robert Graves lecture 3:05 PM in Room 105 WEB, "Digital Terrestrial Television Broadcasting".
- The final project is due Tue, May 5, 5pm. **No late projects** are accepted, because of the late due date. If you think you might be late, set your deadline to May 4.
- Everyone gets a 100% on the "Discussion Item" which I never implemented.

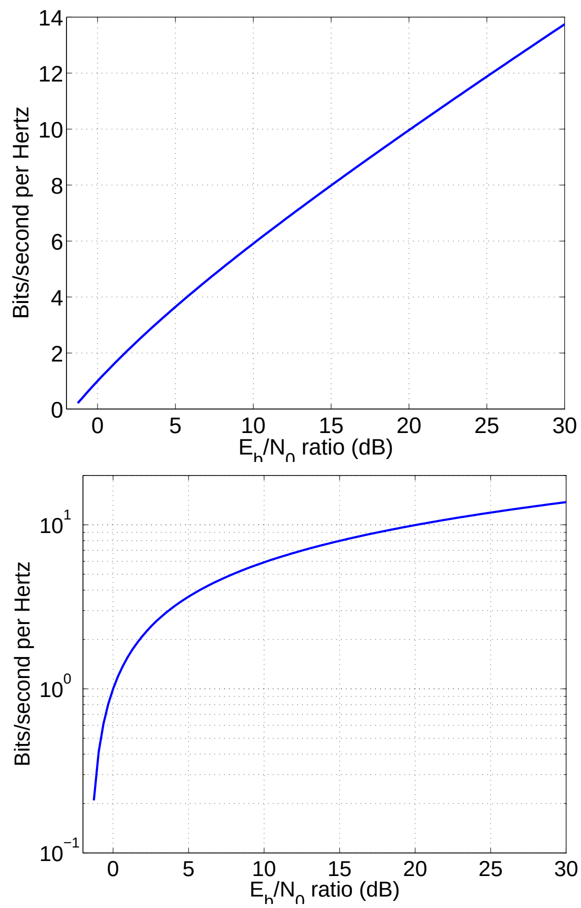


Figure 68: From the Shannon-Hartley theorem, bound on bandwidth efficiency, η, on a (a) linear plot and (b) log plot.

38 Review

Last time, we defined entropy,

$$H[X] = -\sum_i p_i \log_2 p_i$$

and entropy rate,

$$H = \lim_{N\to\infty} \frac{1}{N} H[X_1, X_2, \ldots, X_N].$$

We showed that entropy can be used to quantify information. Given our information source X or $\{X_i\}$, the value of $H[X]$ or H gives us a measure of how many bits we actually need to use to encode, without loss, the source data.

The major result was the Shannon's source coding theorem, which says that a source with entropy rate H can be encoded with arbitrarily small error probability, at any rate R (bits / source output) as long as $R > H$. Any lower rate than H would guarantee loss of information.

39 Channel Coding

Now, we turn to the noisy channel. This discussion of entropy allows us to consider the maximum data rate which can be carried without error on a bandlimited channel, which is affected by additive White Gaussian noise (AWGN).

39.1 R. V. L. Hartley

Ralph V. L. Hartley (born Nov. 30, 1888) received the A.B. degree from the University of Utah in 1909. He worked as a researcher for the Western Electric Company, involved in radio telephony. Afterwards, at Bell Laboratories, he developed relationships useful for determining the capacity of bandlimited communication channels. In July 1928, he published in the Bell System Technical Journal a paper on "Transmission of Information".

Hartley was particularly influenced by Nyquist's sampling theorem. When transmitting a sequence of pulses, each of duration T_s, Nyquist determined that the pulse rate was limited to two times the available channel bandwidth B,

$$\frac{1}{T_s} \leq 2B.$$

In Hartley's 1928 paper, he considered digital transmission in pulse-amplitude modulated systems. The pulse rate was limited to $2B$, as described by Nyquist. But, depending on how pulse amplitudes were chosen, each pulse could represent more or less information.

In particular, Hartley assumed that the maximum amplitude available to the transmitter was A. Then, Hartley made the assumption that the communication system could discern between pulse amplitudes, if they were at separated by at least a voltage spacing of A_δ. Given that a PAM system operates from 0 to A in increments of A_δ, the number of different pulse amplitudes (symbols) is

$$M = 1 + \frac{A}{A_\delta}$$

Note that early receivers were modified AM envelope detectors, and did not deal well with negative amplitudes.

Next, Hartley used the 'bit' measure to quantify the data which could be encoded using M amplitude levels,

$$\log_2 M = \log_2 \left(1 + \frac{A}{A_\delta}\right)$$

Finally, Hartley quantified the data rate using Nyquist's relationship to determine the maximum rate C, in bits per second, possible from the digital communication system,

$$C = 2B \log_2 \left(1 + \frac{A}{A_\delta}\right)$$

39.2 C. E. Shannon

What was left unanswered by Hartley's capacity formula was the relationship between noise and the minimum amplitude separation between symbols. Engineers would have to be conservative when setting A_δ to ensure a low probability of error.

Furthermore, the capacity formula was for a particular type of PAM system, and did not say anything fundamental about the relationship between capacity and bandwidth for arbitrary modulation.

39.2.1 Noisy Channel

Shannon did take into account an AWGN channel, and used statistics to develop a universal bound for capacity, regardless of modulation type. In this AWGN channel, the ith symbol sample at the receiver (after the matched filter, assuming perfect synchronization) is y_i,

$$y_i = x_i + z_i$$

where X is the transmitted signal and Z is the noise in the channel. The noise term z_i is assumed to be i.i.d. Gaussian with variance $E_N = N_0/2$.

39.2.2 Introduction of Latency

Shannon's key insight was to exchange latency (time delay) for reduced probability of error. In fact, his capacity bound considers n-dimensional signaling. So the received vector is $\mathbf{y} = [y_1, \ldots, y_n]$, of length n. These might be truly an n-dimensional signal (*e.g.*, FSK), or they might use multiple symbols over time (recall that symbols at different multiples of T_s are orthogonal). In either case, Shannon uses all n dimensions in the constellation – the detector must use all n samples of $\mathbf{y}$ to make a decision. In the multiple symbols over time, this late decision will decide all values of $\mathbf{x} = [x_1, \ldots, x_n]$ simultaneously. Further, Shannon's proof considers the limiting case as $n \rightarrow \infty$.

This asymptotic limit as $n \rightarrow \infty$ allows for a proof using the statistical convergence of a sequence of random variables. In particular, we need a law called *the law of large numbers.* This law says that the following event,

$$\frac{1}{n} \sum_{i=1}^{n} (y_i - x_i)^2 \leq E_N$$

happens with probability one, as $n \rightarrow \infty$. In other words, as $n \rightarrow \infty$, the measured value $\mathbf{y}$ will be located within an n-dimensional sphere (hypersphere) of radius $\sqrt{nE_N}$ with center $\mathbf{x}$.

39.2.3 Introduction of Power Limitation

Shannon also formulated the problem as a energy-limited case, in which the *maximum* symbol energy in the desired signal x_i was limited to E. That is,

$$\frac{1}{n} \sum_{i=1}^{n} x_i^2 \leq E$$

This combination of signal energy limitation and noise energy results in the fact that,

$$\frac{1}{n}\sum_{i=1}^{n} y_i^2 \leq \frac{1}{n}\sum_{i=1}^{n} x_i^2 + \frac{1}{n}\sum_{i=1}^{n}(y_i - x_i)^2 \leq E + E_N$$

As a result

$$\|\mathbf{y}\|^2 \leq n(E + E_N)$$

This result says that the vector $\mathbf{y}$, with probability one as $n \to \infty$, is contained within a hypersphere of radius $\sqrt{n(E + E_N)}$ centered at the origin.

39.3 Combining Two Results

The two results, together, show how we many different symbols we could have uniquely distinguished, within a period of n sample times. Hartley asked how many symbol amplitudes could be fit into $[0, A]$ such that they are all separated by A_δ. Shannon's formulation asks us how many multidimensional amplitudes $\mathbf{x}_i$ can be fit into a hypersphere of radius $\sqrt{n(E + E_N)}$ centered at the origin, such that hyperspheres of radius $\sqrt{nE_N}$ do not overlap. This is shown in Figure 69.

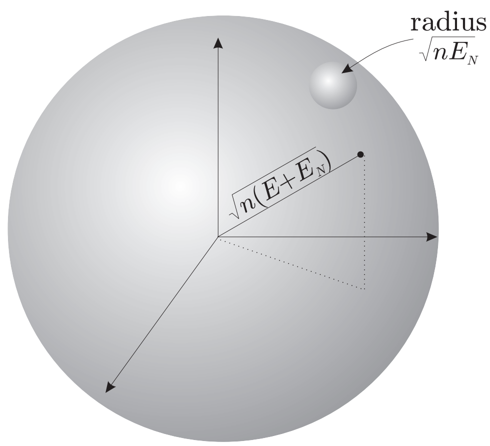


Figure 69: Shannon's capacity formulation simplifies to the geometrical question of: how many hyperspheres of a smaller radius $\sqrt{nE_N}$ fit into a hypersphere of radius $\sqrt{n(E + E_N)}$?

This number M is the number of different messages that could have been sent in n pulses. The result of this geometrical problem is that

$$M = \left(1 + \frac{E}{E_N}\right)^{n/2} \tag{76}$$

39.3.1 Returning to Hartley

Adjusting Hartley's formula, if we could send M messages now in n pulses (rather than 1 pulse) we would adjust capacity to be:

$$C = \frac{2B}{n}\log_2 M$$

Using the M from (76) above,

$$C = \frac{2B}{n}\frac{n}{2}\log_2\left(1 + \frac{E}{E_N}\right) = B\log_2\left(1 + \frac{E}{E_N}\right)$$

39.3.2 Final Results

Since energy is power multiplied by time, $E = PT_s = \frac{P}{2B}$ where P is the maximum signal power and B is the bandwidth, and $E_N = N_0/2$, we have the Shannon-Hartley Theorem,

$$C = B \log_2 \left(1 + \frac{P}{N_0 B}\right). \tag{77}$$

This result says that **a communication system can operate at bit rate** C (in a bandlimited channel with width B given power limit E and noise value N_0), **with arbitrarily low probability of error.**

Shannon also proved that any system which operates at a bit rate higher than the capacity C will certainly incur a positive bit error rate. Any practical communication system must operate at $R_b < C$, where R_b is the operating bit rate.

Note that the ratio $\frac{E}{N_0 B}$ is the signal power divided by the noise power, or signal to noise ratio (SNR). Thus the capacity bound is also written $C = B \log_2(1 + SNR)$.

39.4 Efficiency Bound

Another way to write the maximum signal power P is to multiply it by the bit period and use it as the maximum energy per bit, *i.e.*, $\mathcal{E}_b = PT_b$. That is, the energy per bit is the maximum power multiplied by the bit duration. Thus from (77),

$$C = B \log_2 \left(1 + \frac{\mathcal{E}_b/T_b}{N_0 B}\right)$$

or since $R_b = 1/T_b$,

$$C = B \log_2 \left(1 + \frac{R_b}{B} \frac{\mathcal{E}_b}{N_0}\right)$$

Here, C is just a capacity limit. Be know that our bit rate $R_b \leq C$, so

$$\frac{R_b}{B} \leq \log_2 \left(1 + \frac{R_b}{B} \frac{\mathcal{E}_b}{N_0}\right)$$

Defining $\eta = \frac{R_b}{B}$ (the spectral efficiency),

$$\eta \leq \log_2 \left(1 + \eta \frac{\mathcal{E}_b}{N_0}\right)$$

This expression can't analytically be solved for η. However, you can look at it as a bound on the bandwidth efficiency as a function of the $\frac{\mathcal{E}_b}{N_0}$ ratio. This relationship is shown in Figure 70. Figure 71 is the plot on a log-y axis with some of the modulation types discussed this semester.

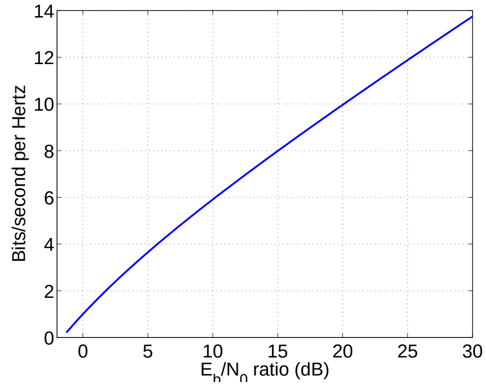


Figure 70: From the Shannon-Hartley theorem, bound on bandwidth efficiency, η.

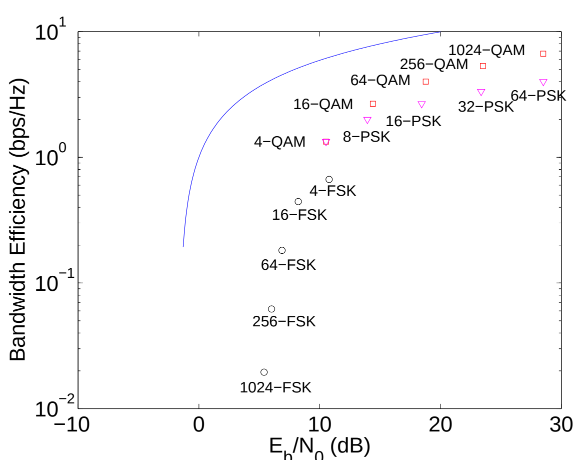


Figure 71: From the Shannon-Hartley theorem bound with achieved bandwidth efficiencies of M-QAM, M-PSK, and M-FSK.