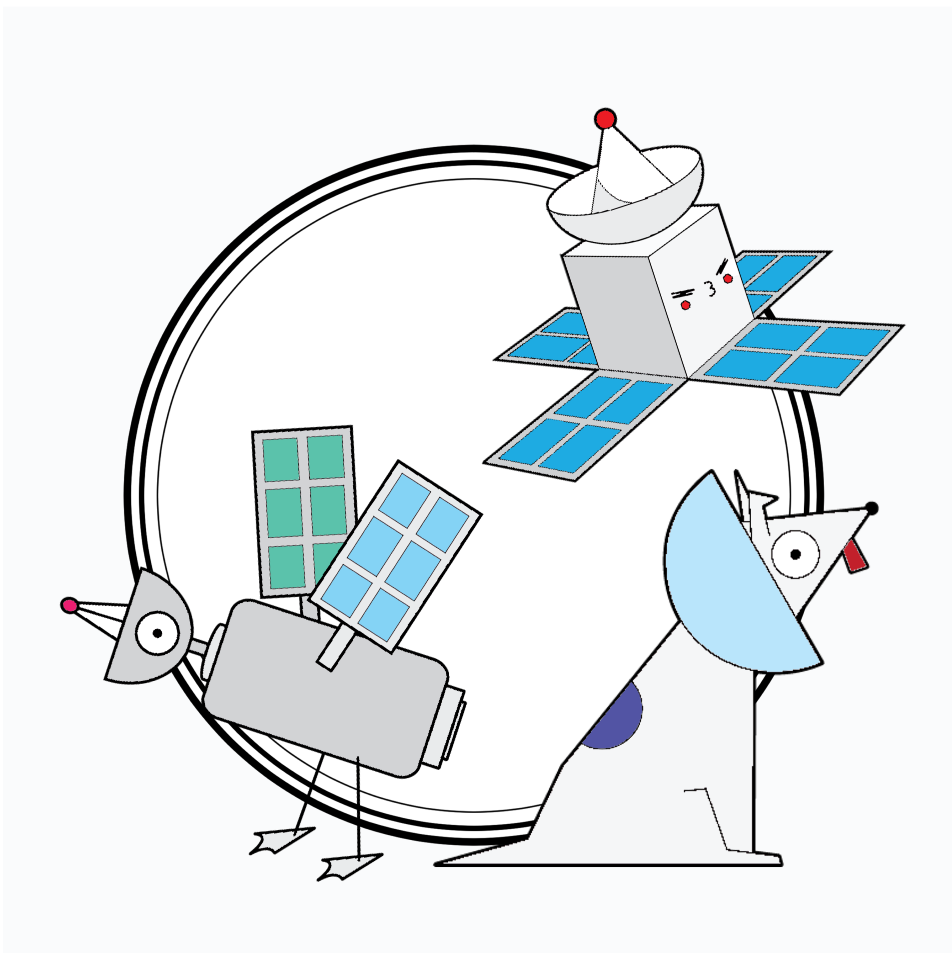
Analog and Digital Communications



Siddhartan Govindasamy, Whitney Lohmeyer, Michael Remley, Sreenidhi Chalimadugu, Kristtiya Guerra

Edition 1 - FA23 Beta

Thank you to the reviewers who helped make this textbook happen:
Dhara Patel
Jillian MacGregor
Charlotte Ramiro

Contents

Part I

Introduction

Chapter 1: Introduction

The year 2019, marked the first year in which more than half the world had access to the Internet according to estimates by the International Telecommunication Union [1]. More than 150 years after the invention of the telephone and forty years after the invention of the Internet, electronic communication is increasingly present in our everyday lives.

Today, we live in an inherently digital world. We spend the majority of our days communicating with others via talk, text, and through Internet applications. When we talk, sound waves are transmitted from our vocal chords and mouth, and are received and processed in our ears. When we call someone, those sound waves are digitized, modulated, converted to an analog wave that is sent "over the air" so that a receiver can take the information, convert it back to the digital domain and process the call as sound that we receive and hear in our ears! The key elements of a communications system are shown below in Figure 1.1.

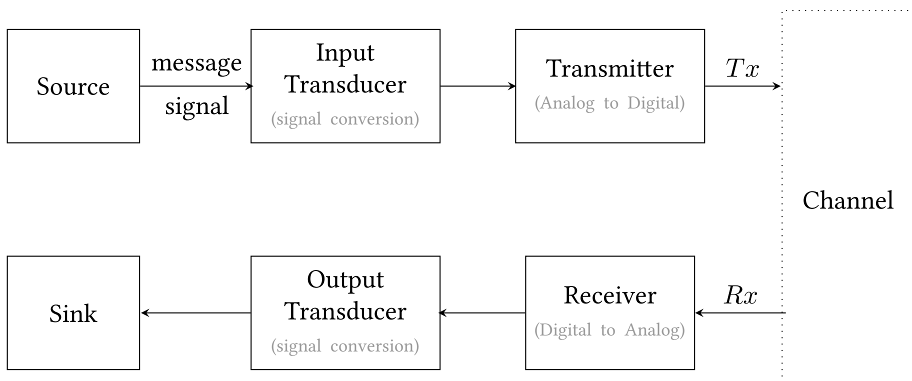


Figure 1.1: Block diagram of the communications system.

A digital communications system used to transmit analog data, such as a digital telephone, can be broken down into the following blocks:

- Source: sound or other information
- Input Transducer: a device that converts the physical quantity to electrical signal (voltage)
- Transmitter (Tx): converts electrical signal into a waveform; analog to digital conversion (ADC) takes place here along with sampling and quantization. The transmitter also handles compression, encoding, modulation and amplification
- Channel: Transmission medium and routing. The channel used is often 1) wired (optical fiber that carries light or coaxial cable and waveguide that carries an electrical signal with a voltage and current) or 2) wireless (radio waves, acoustic waves, optical)
- Receiver (Rx): recovers the original signal passed from the transmitter and converts the analog waveform into a digital signal by reversing the process in the transmitter. This process consists of fre-

quency translation, amplification, filtering, demodulation, demultiplexing, decoding, deciphering, error correction, reconstruction, and digital to analog conversion (DAC)

- Output Transducer: hardware that converts the electrical signal into the intended quantity (e.g. audio)
- Sink: the endpoint of the communications link

The combination of these blocks forms a communications link.

You may also come across the term, transceiver. This is a device that is capable of both transmitting and receiving signals. When we describe a device's capability to transmit and receive in time we use the terminology *duplex*. A full-duplex system can transmit and receive at the same time, whereas a half-duplex system is only capable of either transmitting or receiving at a given moment in time. A system that either only transmits or only receives is called *simplex*.

The objective a communications system is often to send data over a channel as quickly and as efficiently as possible given a set of constraints throughout the link. Data rate, measured in bits per second (bps), and spectral efficiency, measured in bps/Hz quantify how quickly data is sent, which ultimately defines the link performance. Constraints that impact the link performance include bandwidth, or portions of contiguous spectrum, measured in Hz, power, error rate, and latency, which is the time it takes for a signal to travel from one point to another.

Communications systems are impaired by channel distortion and additive noise. This text will primarily focus on linear time invariant channels and additive noise. Although non-linear channels and other forms of noise are prevalent in certain systems, we will not focus on these more complicated scenarios in this text.

1.1 How to Use This Book

This book contains 11 Chapters covering a variety of Analog and Digital Communications concepts.

Chapter 1: **Introduction**: An introduction to analog and digital communication and how to use this textbook to further your studies.

Chapter 2: **Signals and Systems Review**: A review of fundamental signals and systems concepts that are essential for understanding the math behind wireless communications.

Chapter 3: **Amplitude Modulation (AM)**: The fundamentals of amplitude modulation, including different AM techniques that are used in the real world.

Chapter 4: **Frequency Modulation (FM)**: The fundamentals of frequency modulation, an essential technique used for data collection and communication.

Chapter 5: **Probability and Random Variables**: A change in gear to statistics and probability. Probability is an important mathematical tool that allows us to characterize a system and its efficiency. Random variables are important for testing a system and even mapping out natural phenomena.

Chapter 6: **Random Processes**: Random processes are means of dealing with naturally occurring random variables in nature, such as white noise. This chapter will cover important techniques used to recover data in the presence of unwanted noise.

Chapter 7: **Pulse Amplitude Modulation (PAM) and Inter-Symbol Interference (ISI)**: This chapter will discuss pulse amplitude modulation and how it is performed. It will also cover inter-symbol interference, which is interference and data loss caused by the signal itself.

Chapter 8: **Noise Power and Cauchy Schwartz Inequality**: This chapter will teach you how to determine the noise power of a system. This information is useful for then filtering out said noise. It will also cover the Cauchy Schwartz Inequality, which is an essential technique that allows you to derive a filter for a system.

Chapter 9: **M-PAM and System Design**: This chapter will cover how to design an M-PAM system with noise. At this point, you have already learned about PAM. With M-PAM, you can design a system that allows the transmission and retrieval of M symbols.

Chapter 10: **Generalized Modulation Techniques**: This chapter covers additional modulation techniques that are useful for signal processing.

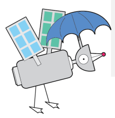
Chapter 11: **Quadrature Amplitude Modulation (QAM)**: This chapter covers the math behind quadrature amplitude modulation, and how to perform quadrature phase shift keying (QPSK) and M-quadrature amplitude modulation (M-QAM).

In addition to our textbook, you may find “Fundamentals of Communications Systems”, by Proakis and Salehi, [2], a helpful resource.

There are six problem sets and a lab that can be conducted independently or in a team.

Throughout this book, our friendly pals Sat, Squit, and Doggo will be stand by your side, ready to support you in your learning adventure! Sat is a mischievous bird that loves trivia and exercises! Whenever you see Sat, a problem is sure to follow.

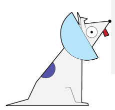
Exercise with Sat



Skwaaa! Nice to meet you, as we say in bird speak. My name is Sat, and I'll be the one who challenges your noggin' to make sure you understand what we introduce.

Next we have Doggo. Doggo is your loyal companion, who will provide helpful tips and treats, so you don't get lost along the way.

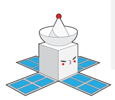
Fun Facts from Doggo



Woof! My name is Doggo. I love treats, belly rubs, and helping you! Just look out for me for a tasty treat.

Last but not least, Squit is our octopod historian that loves nothing more than a good fun fact! Squit will always have something fascinating to share.

History with Squit!



Sploof! Howdy, as we say down under, my name is Squit: your one stop shop for history. We may think of history as just boring dead people, but history is a living thing that we are all making. Look out for me for learning about the past, present, and future!

We actually have one more member of our team: **you**. All of us, including Sat, Squit, and Doggo, want to see you learn and succeed. If you ever feel overwhelmed, don't panic, just take a step back and revisit what you've learned. Space is hard to understand, but we'll be here with you for every step of the way. This text uses styling to emphasize important information, examples, and practice exercises.

Example 1.1

This is an example problem. Example problems come with explained solutions.

Exercise 1.1

This is an exercise containing a practice problem. These come in sets at the end of a select six chapters and do not include solutions.

Chapter 2: Signals and Systems Review

In this chapter, we will dive back into the world of signals and systems, and review fundamental concepts such as: continuous and discrete time, digital processing, sampling, analog to digital conversion, quantization, and channel modeling!

2.1 *Introduction to Signals and Systems*

A signal is a function of one or more variables, typically time, that carries information. Examples of signals include: 1) the voltage produced by an audio device to drive speakers, 2) sound waves, which are variations in sound pressure level, and 3) electromagnetic waves.

Systems process one or more input signal to produce one or more output signal, and can model physical devices such as cell phones, which take electromagnetic signals as inputs and turn them into sound waves. More generally, systems can model wireless communications channels that takes an input, the electromagnetic wave produced by a source device, and output an electromagnetic wave to a destination device.

2.2 *Linear Time Invariant (LTI) Systems*

Communications channels tend to be non-linear and time varying, yet under appropriate assumptions, can be modelled as linear and time varying, or linear, and time invariant. All communications systems are inherently subject to noise, which can be modeled as *additive* noise.

To begin, we will model an LTI communications system without noise, as shown in Figure 2.1. An LTI system is **completely** characterized by its impulse response, denoted $h(t)$, which is the output of a system when the input is an unit impulse (more on this later). These systems can scale signals, delay signals, and change the amplitude of certain frequencies within the signal. An LTI system cannot, however, introduce new frequencies in its output(s), which are not present in its input(s).

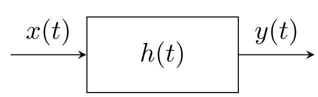


Figure 2.1: A linear time-invariant (LTI) system with input, $x(t)$, impulse response, $h(t)$, and output, $y(t)$.

Examples of systems, which (under the right conditions) can be modeled as LTI include:

- Communications channels over a short enough time scale, such that the impulse response does not change significantly
- Circuits with resistors (R), inductors (L), and capacitors (C), called RLC circuits
- Ideal mass-spring-damper systems, which are mathematically equivalent to RLC circuits
- Certain feedback control systems

To prove a system is LTI, one must show that it is both linear and time-invariant. Let $x(t)$ refer to the input of a system and $y(t)$ refer to the output. Consider two different inputs to the system, $x_1(t)$ and $x_2(t)$. If the input to a system is $x_1(t)$ (i.e. $x(t) = x_1(t)$), then the corresponding output is $y_1(t)$ ($y(t) = y_1(t)$). If the input to a system is $x_2(t)$ (i.e. $x(t) = x_2(t)$), then the corresponding output is $y_2(t)$ ($y(t) = y_2(t)$). This if-then statement is written below in equation form for both inputs and their respective outputs.

$$x(t) = x_1(t) \longrightarrow y(t) = y_1(t)$$
$$x(t) = x_2(t) \longrightarrow y(t) = y_2(t)$$

The system is **linear** if the relationship between inputs and outputs is both

1. Homogeneous: $ax_1(t) \longrightarrow ay_1(t)$ for any constant a
2. Additive: $x(t) = x_1(t) + x_2(t) \longrightarrow y(t) = y_1(t) + y_2(t)$

This combination of properties is also known as **superposition**. If the superposition property holds for a system, and the input to a system is a weighted sum of two signals,

$$x(t) = ax_1(t) + bx_2(t) \tag{2.1}$$

for constants a and b, then the output is also a superposition as follows

$$y(t) = ay_1(t) + by_2(t) \tag{2.2}$$

In other words, when the input is a weighted sum of two (or more) signals, then the output is the weighted sum of the responses to each of those signals.

Note that the idea readily extends to the sum of more than two signals by repeatedly applying the superposition property.

To show that the system is **time invariant**, we need to show that a shift in the input by T time units will result in a shift in the output by T time units. In other words, suppose $x_1(t)$ is the input to the system, which produces an output $y_1(t)$, i.e.

$$x(t) = x_1(t) \longrightarrow y(t) = y_1(t) \tag{2.3}$$

If the input is $x_1(t)$ but delayed by T seconds, then it would be written as $x(t) = x_1(t - T)$. The system is then time invariant if and only if the output is $y_1(t)$ identically delayed

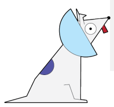
$$x(t) = x_1(t - T) \longrightarrow y(t) = y_1(t - T) \tag{2.4}$$

A system is LTI if both linearity and time invariance hold. To see if a system is LTI, we have to test if a delayed superposition of two input signals produces the correct output implied by linearity and time invariance:

$$\text{If } x(t) = ax_1(t - T_1) + bx_2(t - T_2) \tag{2.5}$$
$$\text{Then } y(t) = ay_1(t - T_1) + by_2(t - T_2)\,, \tag{2.6}$$

These equations must hold for all inputs $x_1(t), x_2(t)$ and any constants a, b, T_1 and T_2. Equation 2.5 will be used to check if a given system is LTI.

Fun Facts from Doggo



Shifting a signal $x(t - t_0)$ corresponds to a delay if t_0 is positive ($t_0 > 0$), or it corresponds to an advance if t_0 is negative ($t_0 < 0$). When plotting a signal, a delay shifts the signal to the right in time, whereas an advance shifts the signal to the left.

The following exercises will offer practice into proving LTI systems.

Example 2.1

Determine if the following system is LTI:

$$y(t) = 3x + 2$$

Solution:

If $x_1(t)$ is an input into the system that produces $y_1(t) = 3x_1(t) + 2$, and $x_2(t)$ is another input into the system that produces $y_2(t) = 3x_2(t) + 2$, then define a new input

$$x_3(t) = ax_1(t) + bx_2(t)$$

which has the output

$$\begin{aligned} y_3(t) &= 3(ax_1(t) + bx_2(t)) + 2 \\ &= 3ax_1(t) + 3bx_2(t) + 2 \end{aligned}$$

Step 1. Checking for Linearity:
In order for the system to be linear, it must be shown that this total output is equivalent to a linear combination of the individual outputs:

$$y_3(t) = ay_1(t) + by_2(t)$$

Plugging y_1 and y_2 into this equation yields

$$\begin{aligned} y_3(t) &= 3ax_1(t) + 3bx_2(t) + 4 \\ 3ax_1(t) + 3bx_2(t) + 2 &\neq 3ax_1(t) + 3bx_2(t) + 4 \end{aligned}$$

which does not equal the expected output if the system were linear. Therefore, the system is **not linear**. Adding a constant which does not scale with the input causes this system to fail the linearity test.
Step 2. Checking for Time-Invariance:
To show the system is time-invariant, show that delaying the input produces a delayed version of the output, with the same delay. Let $x_2(t)$ be a delayed version of $x_1(t)$ by a constant C.

$$x_2(t) = x_1(t - C)$$

x_1 produces $y_1(t) = 3x_1(t) + 2$, so $x_2(t)$ must produce $y_2(t) = y_1(t - C)$ in order to be time-invariant.

$$\begin{aligned} y_2(t) &= 3x_2(t) + 2 \\ &= 3x_1(t - C) + 2 \\ \therefore y_2(t) &= y_1(t - C) \end{aligned}$$

Delaying the input delays the output by the same amount, so the system is **time-invariant**.

In the next example, we will investigate an exponential function.

Example 2.2

Determine if the following system is LTI:

$$y(t) = x^2(t)$$

Solution:

Here we will follow the same two step process as in the previous example.

Step 1. Checking for Linearity:
To check linearity, test two input signals, $x_1(t)$ and $x_2(t)$, like in the previous example. These two inputs produce outputs, $y_1(t)$ and $y_2(t)$

$$\begin{aligned} y_1(t) &= x_1^2(t) \\ y_2(t) &= x_2^2(t) \end{aligned}$$

A linear combination of these inputs is

$$x_3(t) = ax_1(t) + bx_2(t)$$

$x_3(t)$ produces the output

$$y_3(t) = a^2x_1^2(t) + 2abx_1(t)x_2(t) + b^2x_2^2(t)$$

Test the linearity condition that $y_3(t) = ay_1(t) + by_2(t)$

$$\begin{aligned} y_3(t) &= ay_1(t) + by_2(t) \\ a^2x_1^2(t) + 2abx_1(t)x_2(t) + b^2x_2(t) &\neq ax_1^2(t) + bx_2^2(t) \end{aligned}$$

Not only does an additional term appear, but the scale factors a and b do not match. Therefore, the system is **not linear**. Note that we could have also shown this by setting $b = 0$, which would have resulted in shorter steps since the term involving $x_2(t)$ will not be needed.

Step 2. Checking for Time-Invariance:
Let $x_2(t)$ be $x_1(t)$ delayed by a constant C. The outputs produced by $x_1(t)$ and $x_2(t)$ are

$$\begin{aligned} y_1(t) &= x_1^2(t) \\ y_2(t) &= x_2^2(t) \end{aligned}$$

We test time-invariance by checking if $y_2(t) = y_1(t-C)$ satisfies the equation above given $x_2(t) = x_1(t-C)$

$$\begin{aligned} y_2(t) &= x_2^2(t) \\ &= x_1^2(t-C) \\ \therefore y_2(t) &= y_1(t-C) \end{aligned}$$

The delayed input produces a delayed output, so the system is **time-invariant**.

Next, please attempt to investigate whether the sum of a function and a time shifted version of itself satisfies the conditions of LTI.

Example 2.3

Determine if the following system is LTI:

$$y(t) = 0.2x(t) + 0.02x(t-4)$$

Solution:

Step 1. Checking for Linearity:
To check linearity, use the two input signals, $x_1(t)$ and $x_2(t)$, and the linear combination of the two, $x_3(t) = ax_1(t) + bx_2(t)$. The outputs are

$$\begin{aligned} y_1(t) &= 0.2x_1(t) + 0.02x_1(t-4) \\ y_2(t) &= 0.2x_2(t) + 0.02x_2(t-4) \\ y_3(t) &= 0.2(ax_1(t) + bx_2(t)) + 0.02(ax_1(t-4) + bx_2(t-4)) \end{aligned}$$

Evaluate the linearity condition by replacing $y_3(t) = ay_1(t) + by_2(t)$

$$\begin{aligned} &ay_1(t) + by_2(t) \\ &\qquad = 0.2(ax_1(t) + bx_2(t)) + 0.02(ax_1(t-4) + bx_2(t-4)) \\ &0.2ax_1(t) + 0.02ax_1(t-4) + 0.2bx_2(t) + 0.02bx_2(t-4) \\ &\qquad = 0.2ax_2(t) + 0.2bx_2(t) + 0.02ax_1(t-4) + 0.02bx_2(t-4) \end{aligned}$$

Every term on the left is balanced by an equal term on the right, so the system is **linear**.

Step 2. Checking for Time-Invariance:
Test time-invariance with a signal, $x_1(t)$, and a delayed version, $x_2(t) = x_1(t-C)$, where C is a constant. The outputs are

$$\begin{aligned} y_1(t) &= 0.2x_1(t) + 0.02x_1(t-4) \\ y_2(t) &= 0.2x_1(t-C) + 0.02x_1(t-C-4) \end{aligned}$$

Delaying $y_1(t)$ by C will equal $y_2(t)$ if the system is time-invariant

$$\begin{aligned} y_1(t-C) &= y_2(t) \\ 0.2x_1(t-C) + 0.02x_1(t-C-4) &= 0.2x_1(t-C) + 0.02x_1(t-C-4) \end{aligned}$$

The delayed input produced the same delayed output, so the system is **time-invariant**. The system is both linear and time-invariant, so it is also **LTI**. This particular system is an echo, where the echo takes 4 time units to reach output and is much quieter than the original input.

Lastly, we will investigate a discrete trigonometric function!

Example 2.4

Is the following discrete-time system LTI?

$$y[n] = x[n]\cos(0.2\pi n)$$

Solution:

Step 1. Checking for Linearity:

$$\begin{aligned} x[n] &= ax_1[n] + bx_2[n] \\ y[n] &= (ax_1[n] + bx_2[n])\cos(0.2\pi n) \\ &= ax_1[n]\cos(0.2\pi n) + bx_2[n]\cos(0.2\pi n) \\ &= ay_1 + by_2 \end{aligned}$$

Therefore, the system is **linear**.

Step 2. Checking for Time-Invariance:
Test time-invariance with a signal $x_1[n]$ and a delayed version $x_2[n] = x_1[n-n_0]$ where n_0 is a

constant. The outputs are

$$y_1[n] = x_1[n]\cos(0.2\pi n)$$
$$y_2[n] = x_2[n]\cos(0.2\pi n)$$

We test time-invariance by checking if $y_2[n] = y_1[n-n_0]$ satisfies the equation above given $x_2[n] = x_1[n-n_0]$

$$\begin{aligned} y_1[n-n_0] &= x_1[n-n_0]\cos(0.2\pi(n-n_0)) \\ y_2[n] &= x_2[n]\cos(0.2\pi n) \\ &= x_1[n-n_0]\cos(0.2\pi n) \\ y_1[n-n_0] &= y_2[n] \\ x_1[n-n_0]\cos(0.2\pi(n-n_0)) &\neq x_1[n-n_0]\cos(0.2\pi n) \end{aligned}$$

The system is **not time-invariant**, since $y_1[n-n_0] \neq y_2[n]$.

If these examples were challenging to follow, try using the following prompts to find further information:

- How to prove linearity and time invariance
- Examples of proving LTI
- Prove the superposition principle for LTI systems

2.3 Continuous Time (CT) Systems

Continuous-time (CT) systems operate on continuous input signals to produce continuous outputs. Continuous signals are signals whose inputs and outputs are defined at all real-numbered values of time and are typically denoted using parentheses, e.g., $x(-1), x(0), x(1), x(\pi)$. Signals like temperature in the real world and analog signals are examples of continuous time systems because they are defined at any time t.

Unit Impulse

A helpful building block is the CT unit impulse function, or the Dirac Delta, $\delta(t)$. Loosely speaking, the CT unit impulse has the following properties:

- Infinite height at $t = 0$
- Zero-valued everywhere else
- Unit area ($A = 1$)

Consider a rectangular pulse whose width is Δ and height is $\frac{1}{\Delta}$, as depicted in Figure 2.2.

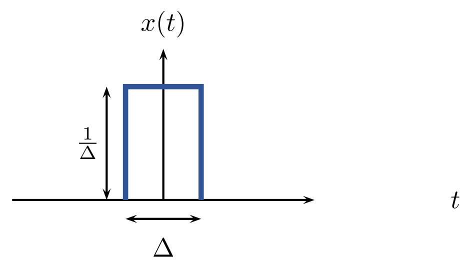


Figure 2.2: Rectangular pulse with width Δ and height $\frac{1}{\Delta}$

We can define $\delta(t)$ as follows

$$\delta(t) = \lim_{\Delta \to 0} x(t) \tag{2.7}$$

The Dirac Delta, shown in Figure 2.3, is drawn as a vertical arrow with its area noted in parenthesis. This is not to be confused with the impulse having a value of 1 at zero, but rather the area under the impulse being 1.

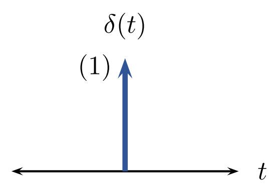


Figure 2.3: Unit impulse, $\delta(t)$

While the Dirac Delta function is not realizable, it can be approximated by a real-world signal, and proves helpful in areas such as spectral analysis because it comprises equal portions of all possible frequencies. This allows LTI systems to be characterized at all frequencies using only a single test input.

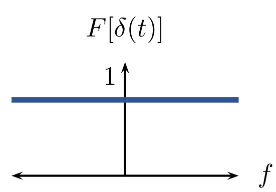


Figure 2.4: A depiction of a unit impulse in the frequency domain, $F[\delta(t)]$

Continuous Time Fourier Transform (CTFT)

The Fourier Transform (FT) uniquely maps signals from the time domain to the frequency domain, and expresses a function of time as a sum of sines and cosines, or complex exponentials.

In general, the FT produces complex values for every frequency, often written in terms of a magnitude and phase. A graph of the magnitude versus frequency of the Fourier Transform of a signal, describes the amplitude of the different sinusoids or complex exponentials that make up the signal. A graph of the phase versus frequency of the Fourier Transform describes the phases of those sinusoids or complex exponentials.

The FT can also be viewed as a change of **basis** where a function of time is represented in a basis of sinusoids, or complex exponentials. To break this down further.

The forward Fourier Transform converts a function from time to frequency and is defined by the integral, X(f) where

$$X(f) = F[x(t)] = \int_{-\infty}^{\infty} x(t)e^{-j2\pi ft}dt \tag{2.8}$$

The inverse Fourier Transform converts a function from frequency to time and is computed as

$$x(t) = F^{-1}[X(f)] = \int_{-\infty}^{\infty} X(f)e^{j2\pi ft}df \tag{2.9}$$

Throughout this text, time-domain functions are written with lowercase letters, such as $x(t)$, and the corresponding frequency domain functions are written with uppercase letters, such as $X(f)$ where f is in units of Hz. The relationship of a time-domain function to its frequency counterpart can be written as:

$$x(t) \xleftrightarrow{F} X(f) \tag{2.10}$$

This relationship between $x(t)$ and $X(f)$ is fully invertible (provided $X(f)$ exists). Hence, one can go from a function of time to a function of frequency and vice versa, using the forward and inverse Fourier Transforms

$$X(f) = F[x(t)] \tag{2.11}$$

$$x(t) = F^{-1}[X(f)] \tag{2.12}$$

To express the Fourier Transforms in angular frequency (ω), substitute variables

$$\omega = 2\pi f \tag{2.13}$$

and

$$d\omega = 2\pi f df \tag{2.14}$$

into

$$X(\omega) = \int_{-\infty}^{\infty} x(t)e^{-j\omega t}dt \tag{2.15}$$

and

$$x(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} X(\omega)e^{j\omega t}d\omega \tag{2.16}$$

In this book, Ω will be used to refer to angular frequency in discrete time. You may wonder why $x(t)$ in Equation 2.16 consists of a $\frac{1}{2\pi}$. Mathematically, this results from the substitution of Equation 2.14 into the Fourier Transform integral.

Continuous Time Fourier Transform Properties

The Fourier Transform has several interesting properties relevant to communications, such as:

1. A shift in the time domain produces a phase shift in the frequency domain and vice-versa. Note that magnitude of the FT is not altered!

$$x(t-t_0) \longleftrightarrow e^{-j2\pi f t_0}X(f) \tag{2.17}$$
$$e^{j2\pi f_0 t}x(t) \longleftrightarrow X(f-f_0) \tag{2.18}$$

2. Convolution in the time domain is multiplication in the frequency domain and vice-versa

$$x(t)*y(t) \longleftrightarrow X(f)Y(f) \tag{2.19}$$
$$x(t)y(t) \longleftrightarrow X(f)*Y(f) \tag{2.20}$$

3. The energy of a signal is the same if measured in time or frequency. This relationship, also known as Parseval's Relation, is as follows:

$$\int_{-\infty}^{\infty} |x(t)|^2 dt = \int_{-\infty}^{\infty} |X(f)|^2 df \tag{2.21}$$

4. A linear scaling in time by a corresponds to a linear scaling in the frequency domain by $|\frac{1}{a}|$. For example, scaling can occur when you have a rectangular pulse with different limits.

$$x(at) \longleftrightarrow \frac{1}{|a|}X\left(\frac{f}{a}\right), \text{ where } f = j\omega \tag{2.22}$$

5. A linear combination of time-domain functions produces the same linear combination of frequency domain functions

$$x_1(t) \longleftrightarrow X_1(f) \tag{2.23}$$
$$x_2(t) \longleftrightarrow X_2(f) \tag{2.24}$$
$$\alpha x_1(t) + \beta x_2(t) \longleftrightarrow \alpha X_1(f) + \beta X_2(f) \tag{2.25}$$

You may notice that the Fourier Transform has values at negative frequencies. The magnitude of a Fourier Transform of a real function is even, i.e., symmetric around $f = 0$. Some functions, such as a sine wave, have purely imaginary Fourier Transforms, and others have non-zero real and imaginary parts. In such cases, the amplitude **and** phase of the transform are needed for a complete description. For an intuitive understanding for this, consider that a cosine and sine of the same frequency and amplitude differ only in their phase. Throughout this text, illustrations of transforms show the real part unless otherwise specified.

Example 2.5

Compute the Fourier Transform of the Dirac Delta Function.

Solution:

Start with Equation 2.8, where $x(t)$ is the the Dirac Delta function

$$F\{\delta(t)\} = \int_{-\infty}^{\infty} \delta(t)e^{-j2\pi ft}dt$$

The integral bounds are for all time, t, but the Dirac Delta is only non-zero at $t = 0$. Thus, replace $t = 0$ in the exponential, and recognize that $e^0 = 1$.

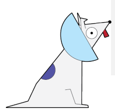
$$\begin{aligned} F\{\delta(t)\} &= \int_{-\infty}^{\infty} \delta(t)e^{0}dt \\ &= \int_{-\infty}^{\infty} \delta(t)dt \end{aligned}$$

Recall that the Dirac Delta has unit area and recognize that the integral now evaluates the area under $\delta(t)$:

$$F\{\delta(t)\} = 1$$

Therefore, the impulse in the time domain is constant in the frequency domain, or in other words contains equal portions of all frequencies!

Fun Facts from Doggo



The Fourier Transform of the CT unit impulse has constant magnitude for all frequencies. Later in this text, we will encounter white noise processes, which are random signals whose average power is constant across frequencies.

Euler's Equation

The Fourier Transform multiplies the input function by $e^{-j2\pi ft}$, which is called a *complex exponential*. According to Euler's formula, complex exponentials are related to sines and cosines by the equation:

$$e^{j\theta} = \cos(\theta) + j\sin(\theta) \tag{2.26}$$

Complex exponents are convenient because they turn trigonometric problems into algebraic problems. Graphically, exponential imaginary growth moves around a circle at a constant rate. In the CTFT, integrating over the product of a signal multiplied by a complex exponential measures how correlated the signal is to a given frequency.

It is sometimes useful to replace cosine or sines with exponentials. Plugging $-\theta$ into Euler's equation leads to Equation 2.28:

$$e^{j(-\theta)} = \cos(-\theta) + j\sin(-\theta) \tag{2.27}$$

which simplifies to

$$e^{-j\theta} = \cos(\theta) - j\sin(\theta) \tag{2.28}$$

To solve for $\cos(\theta)$, add Equation 2.26 to Equation 2.28:

$$e^{j\theta} + e^{-j\theta} = 2\cos(\theta) \tag{2.29}$$

$$\cos(\theta) = \frac{1}{2}[e^{j\theta} + e^{-j\theta}] \tag{2.30}$$

Likewise, an equation for sine can be found. To solve for $\sin(\theta)$, subtract Equation 2.28 from Equation 2.26:

$$e^{j\theta} - e^{-j\theta} = 2j\sin(\theta) \tag{2.31}$$

$$\sin(\theta) = \frac{1}{2j}[e^{j\theta} - e^{-j\theta}] \tag{2.32}$$

We will see the $\frac{1}{2}$ that is distributed becomes the area of the spikes in the Fourier domain, which we can see in Figure 2.36 below.

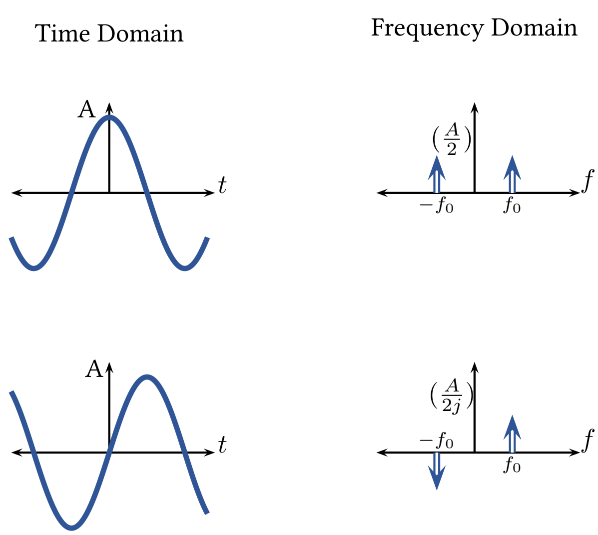
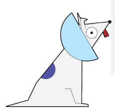


Figure 2.5: CTFT pairs shifted in the time domain

Fun Facts from Doggo



Multiplying by j is a rotation of 90° around 0, and multiplying by -1 is like rotating by 180° counter clockwise. Note that i and j are used interchangeably. Figure 2.6 depicts a Gaussian unit circle along with the components of Euler's equation.

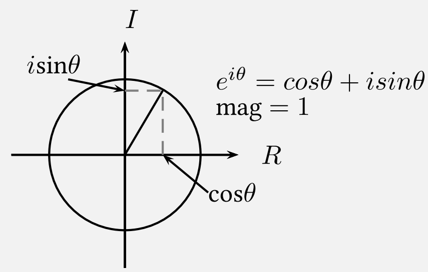


Figure 2.6: Gaussian Unit Circle

Now, we will practice finding the Fourier Transform of various functions.

Example 2.6

Compute the Fourier Transform of a rectangular pulse, $p(t)$.

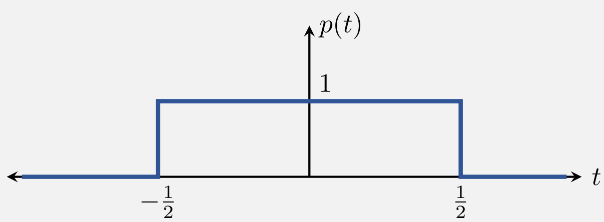


Figure 2.7: Rectangular Pulse, $p(t)$

Solution:

To find the Fourier Transform of a rectangular pulse, $p(t)$, where $p(t)$ is defined as:

$$\sqcap = p(t) = \begin{cases} 1 & |t| < 1/2 \\ 0 & \text{otherwise} \end{cases} \tag{2.33}$$

Plug $p(t)$ into the CTFT, shown in Equation 2.8, and recognize that the integrand is only non-zero between $\pm\frac{1}{2}$

$$\begin{aligned}P(f) &= \int_{-\infty}^{\infty} p(t)e^{-j2\pi ft}dt \\ &= \int_{-1/2}^{1/2} e^{-j2\pi ft}dt, \text{ since } p(t) = 1\end{aligned}$$

Next, evaluate the integral using u-substitution where:

$$\begin{aligned}u &= -j2\pi ft \\ du &= -j2dt \\ P(f) &= \int_{-1/2}^{1/2} \frac{e^{u}du}{-j2\pi ft} \\ &= -\frac{1}{j2\pi f}[e^{-j2\pi ft}]\Big|_{-1/2}^{1/2} \\ &= -\frac{1}{j2\pi f}[e^{-j2\pi f(1/2)} - e^{-j2\pi f(-1/2)}] \\ &= -\frac{1}{j2\pi f}[e^{-j\pi f} - e^{j\pi f}]\end{aligned}$$

Recall Euler's Equation solved for sine

$$\sin(x) = \frac{1}{2j}[e^{-jx} - e^{jx}] \tag{2.34}$$

The Fourier Transform of the box is simplified to the following equation:

$$P(f) = \frac{\sin(\pi f)}{\pi f} \tag{2.35}$$

$$= \text{sinc}(x) \text{ where } x = \pi f, \textit{and} \tag{2.36}$$

$$\text{sinc}(x) = \frac{\sin(\pi x)}{\pi x} \tag{2.37}$$

sinc(x) is called the *normalized sinc function* and is depicted in Figure 2.8.

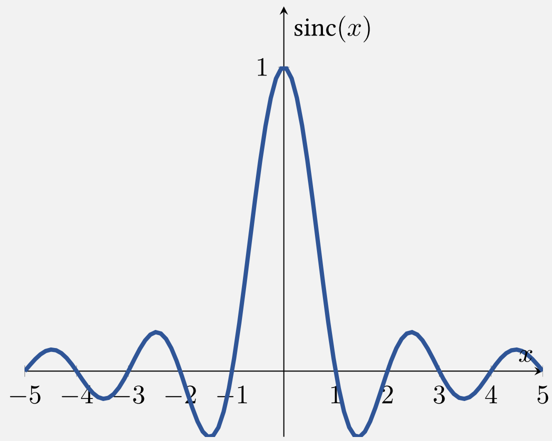


Figure 2.8: Normalized sinc function

This sinc(x) function has a maximum value of 1, as the maximum value occurs at $x = 0$, where both the numerator and denominator of the sinc function in 2.37 are both zero. Taking the limit as $x \to 0$, using, for instance, L'Hopital's rule we find that the value of $sinc(0) = 1$. The zero crossings occur at non-zero values of x for which the numerator in (2.37) equals zero. Since $\sin(\pi x) = 0$ at integer values of x, the zero crossings of the normalized sinc occurs at non-zero integer values of x. The sinc is used to reconstruct a continuous band limited signal from uniformly spaced samples of that signal.

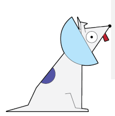
Fun Facts from Doggo

An **unnormalized sinc function** has zero crossings at integer multiples of π, except for zero

$$\text{sinc}(f) = \frac{\sin(f)}{f} = 0 \quad f = k\pi, k \in \mathbb{Z}, k \neq 0 \tag{2.38}$$

A **normalized sinc function** has zero crossings at integers except for zero

$$\text{sinc}(x) = \frac{\sin(\pi x)}{\pi x} = 0 \quad x \in \mathbb{Z}, x \neq 0 \tag{2.39}$$



Be sure to check that the version of sinc is actually the one you are using and need!

Example 2.7

Find the Fourier Transform of the given function:

$$x(t) = \frac{1}{2}x_1(t-2.5) + x_2(t-2.5)$$

Note: x_1 and x_2 are defined as two individual rectangular pulses. Therefore the system can be treated as the sum of two individual pulses in the frequency domain

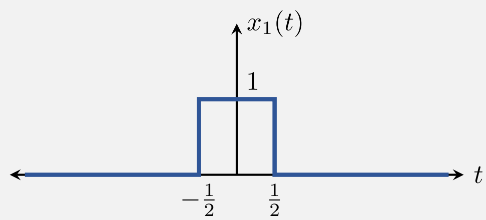


Figure 2.9: Rectangular pulse, $x_1(t)$, in the time domain

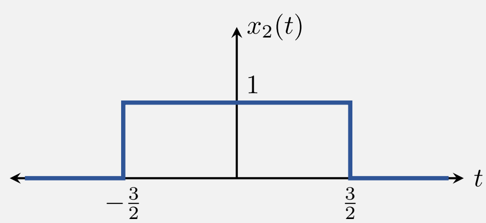


Figure 2.10: The rectangular pulse $x_2(t)$ in time-domain.

Solution:

The Fourier Transforms of $x_1(t)$ and $x_2(t)$, denoted $X_1(f)$ and $X_2(f)$ respectively, are calculated as follows:

$$\begin{aligned}
X_1(f) &= \int_{-\frac{1}{2}}^{\frac{1}{2}} 1e^{-jwt}dt \\
&= \frac{1}{-jw}[e^{-jwt} - e^{-jwt}]\Big|_{-\frac{1}{2}}^{\frac{1}{2}} \\
&= \frac{1}{-jw}[e^{-\frac{jw}{2}} - e^{\frac{jw}{2}}] \\
&= \frac{1}{jw}[e^{\frac{jw}{2}} - e^{-\frac{jw}{2}}] \\
&= \frac{2\sin(\frac{w}{2})}{w}
\end{aligned}$$

$$\begin{aligned}
X_2(f) &= \int_{-\frac{3}{2}}^{\frac{3}{2}} 1e^{-jwt}dt \\
&= \frac{-1}{jw}[e^{-\frac{3}{2}jw} - e^{\frac{3}{2}jw}] \\
&= \frac{1}{jw}[e^{\frac{3}{2}jw} - e^{-\frac{3}{2}jw}] \\
&= \frac{2\sin(\frac{3}{2}w)}{w}
\end{aligned}$$

Thus, the Fourier Transform of $x(t)$, denoted $X(f)$, results in:

$$X(f) = e^{-jw(\frac{5}{2})}(\frac{2\sin(\frac{w}{2})}{w} + \frac{2\sin(\frac{3}{2}w)}{w})$$

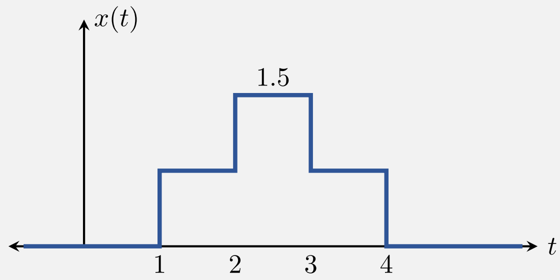


Figure 2.11: The resulting signal $x(t)$ in time-domain.

Example 2.8

Find the Fourier Transform of the given function:

$$y(t) = \text{sinc}(t)\cos(2\pi f t)$$

where $\text{sinc}(t)$ *is defined as*

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

Solution:

Given the Fourier Transform of $\text{sinc}(t)$ is a box in the frequency domain:

$$F[\text{sinc}(t)] = \sqcap(f)$$

the Fourier Transform of $y(t)$ is the rectangular pulse duplicated at $\pm f_0$ whose heights are halved:

$$Y(f) = \frac{\sqcap(f - f_0) + \sqcap(f + f_0)}{2}$$

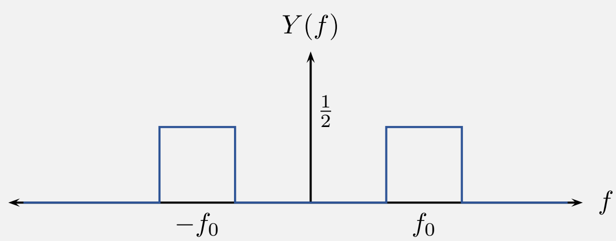


Figure 2.12: Fourier Transform of $y(t) = \text{sinc}(t)\cos(2\pi f t)$

Continuous Time Fourier Transform Pairs

When the Forward and Reverse Fourier Transforms are difficult to evaluate by hand, tables of Fourier Transform pairs can be consulted to save time. A transform pair is a pair of functions that are related by the Forward and Reverse Fourier Transformations. Two interesting pairs, the sinc-box pair and the impulse train pair, are shown in Figure 2.13. Performing the FT on an impulse train with spacing between the impulses of T in the time domain produces an impulse train in the frequency domain, with impulses spaced by $\frac{1}{T}$ Hz and areas scaled by $\frac{1}{T}$. In angular frequency, the spacing instead becomes $\frac{2\pi}{T}$, and the areas of the impulses are scaled by 2π.

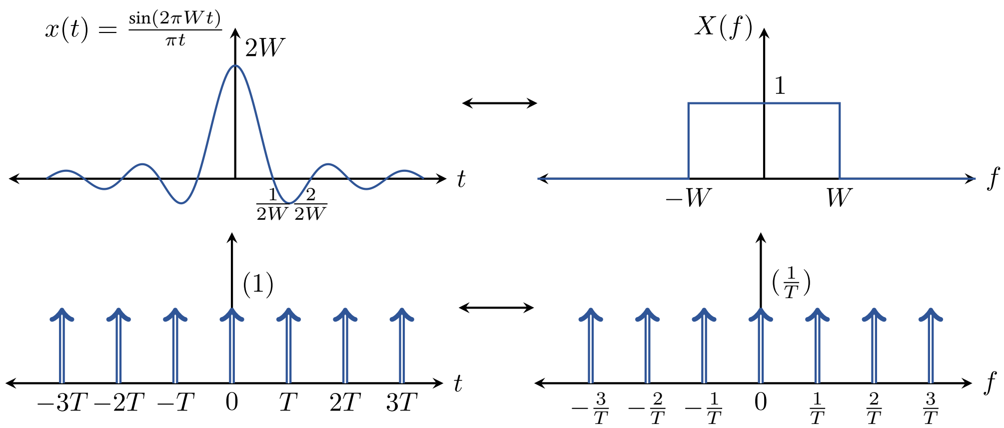


Figure 2.13: The transform pairs of the sinc function in time to a box in frequency (top) and an impulse train (bottom).

The Fourier Coefficients can be calculated using Equation 2.3. Fourier Coefficients describe the signal in the frequency domain and provide a complex amplitude for a signal at a certain frequency, f_k.

$$a_k = \frac{1}{T}\int_{-T/2}^{T/2} \delta(t)e^{-j2\pi f t}dt$$

To show why the heights of the impulse train are scaled by $\frac{1}{T}$, compute the Fourier Coefficient for a single impulse at zero, which corresponds to $k = 0$. **The impulse at $f = 0$ is our DC offset.**

For $k = 0$, let $t = 0$, since $\delta(t)$ is zero when $t \neq 0$. The integral of the impulse evaluates to 1, leaving us with the value for the zeroth Fourier Coefficient

$$\begin{aligned} a_0 &= \frac{1}{T}\int_{-T/2}^{T/2} \delta(t)e^{0}dt \\ &= \frac{1}{T} \end{aligned}$$

Appendix A.2 provides a list of commonly referenced transform pairs including the following three, which you will see frequently in this text. The time-domain expressions on the left map to the frequency-domain expressions on the right through the Fourier Transform

$$e^{j2\pi f_1 t} \longleftrightarrow \delta(f - f_1) \tag{2.40}$$

$$\cos(2\pi f_1 t) \longleftrightarrow \frac{1}{2}\delta(f - f_1) + \frac{1}{2}\delta(f + f_1) \tag{2.41}$$

$$\sin(2\pi f_1 t) \longleftrightarrow \frac{1}{2j}\delta(f - f_1) - \frac{1}{2j}\delta(f + f_1) \tag{2.42}$$

In communications systems, modulation modifies a relatively high frequency carrier signal to convey the content of an message signal. This can be performed by shifting the phase of the carrier wave, as in binary phase shift keying (BPSK) modulation schemes; scaling the amplitude, as in quadrature amplitude modulation (QAM); or shifting the frequency of a carrier signal (frequency modulation). In any case, modulation serves to shift a signal from the baseband (near zero-frequency) to higher frequencies suitable for transmission. A shift in frequency is represented by the following FT pair:

$$x(t)e^{j2\pi f_0 t} \longleftrightarrow X(f - f_0) \tag{2.43}$$

Though the above equation is written as a time-domain signal multiplied by a complex exponential, consider what happens when a message signal $m(t)$ is multiplied by the more familiar cosine.

$$x(t) = m(t)\cos(2\pi f_0 t) \tag{2.44}$$

Remember that cosine functions can be written in terms of complex exponentials:

$$\cos(x) = \frac{e^{jx} + e^{-jx}}{2} \tag{2.45}$$

The input signal, $x(t)$, is equivalently written in terms of complex exponentials by substituting Equation 2.45 to Equation 2.44

$$x(t) = \frac{m(t)e^{j2\pi f_0 t} + m(t)e^{-j2\pi f_0 t}}{2} \tag{2.46}$$

To see how multiplying a message by a cosine moves the message to higher frequency, apply the Fourier Transform pair from Equation 2.43.

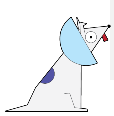
$$X(f) = \frac{M(f - f_0) + M(f + f_0)}{2} \tag{2.47}$$

Note that the time shift does not impact the frequency components in the signal. Rather, it scales the frequency components, which you saw in Figure 2.36 that compares a cosine and sine signal.

Fun Facts from Doggo

In the frequency domain, an impulse at $f = 0$ represents a signal that is constant in time. For example, a constant 5 V electrical signal would be represented by the transform pair

$$x(t) = 5 \longleftrightarrow X(f) = 5\delta(f) \tag{2.48}$$



A constant voltage is also called a **DC offset**, so zero frequency signals and DC offset refer to the same thing.

CT LTI Systems and Convolution

The impulse response, $h(t)$, is a function that completely defines an LTI system, as it defines the output of the LTI system given any input. LTI systems are commonly represented as box with a single input and output, as seen in Figure 2.14.

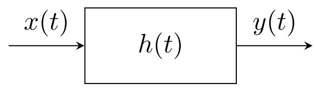


Figure 2.14: A linear time-invariant (LTI) system with input $x(t)$, impulse response $h(t)$, and output $y(t)$.

The impulse response is the reaction (output) of a system to an impulse input as a function of time. The output, $y(t)$, of such an LTI system is related to the input $x(t)$ by the equation:

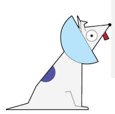
$$y(t) = x * h(t) \tag{2.49}$$

Shifting and time-reversal can be applied to either $x(t)$ or $h(t)$ without changing the overall result, hence the two forms of the equations in Equation 2.50. For completeness, the integral form of the convolution of $x(t)$ and $h(t)$ is

$$y(t) = \int_{-\infty}^{\infty} x(\tau)h(t-\tau)d\tau = \int_{-\infty}^{\infty} h(\tau)x(t-\tau)d\tau \tag{2.50}$$

In Equation 2.49, the new $*$ operator represents convolution. Thus, to determine the output, $y(t)$, of an LTI system in the time domain requires one to convolve the input, $x(t)$, with the impulse response, $h(t)$.

Fun Facts from Doggo



It is common to see convolution written as either $y(t) = x * h(t)$ or $y(t) = x(t) * h(t)$. For this text, we will use the $y(t) = x * h(t)$ convention to emphasize that the convolution occurs for the entire function of x, rather than at a particular time.

If two function are convolved in the time domain, the are multiplied in the frequency domain.

$$Y(f) = X(f)H(f) \tag{2.51}$$

where $Y(f)$, $X(f)$, and $H(f)$ are the frequency-domain representations of $y(t)$, $x(t)$, and $h(t)$, respectively. Convolution is essential because it connects multiplication in the time domain to the corresponding transformation in frequency domain and vice-versa. Without this connection, translating filters and other operations between domains becomes more challenging.

To visualize convolution, first consider the signal, $x(\tau)$, and a **time-reversed** and shifted version of $h(\tau)$, i.e. $h(t-\tau)$. From (2.50), note that the value of $y(t)$ is the area under the graph $x(\tau)h(t-\tau)$. So, to evaluate $y(t)$, we can consider the area beneath the graph $x(\tau)h(t-\tau)$ for different values of t. It is customary in visualizing convolution to shift one of the signals to the far left (i.e. very negative value of t), and slide it to the right along the horizontal axis as t increases. The area of the product for a particular shift is the value of the convolution. If you are having trouble visualizing convolution, try researching animations or interactive demonstrations such as [3].

Figure 2.15: Signals $x(\tau)$ and $h(\tau)$

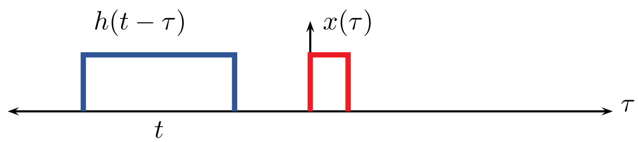


Figure 2.16: Signals $x(\tau)$ and $h(t-\tau)$

Convolution also has several mathematical properties including that it is commutative, associative, and distributive:

$$x * h = h * x$$
$$(x * h) * g = x * (h * g)$$
$$x * (h + g) = x * h + x * g$$

What happens if a signal is convolved with an impulse, $\delta(t)$? Because the impulse has unit area and is zero everywhere except one point, the integral of its product with another equation is that equation's value delayed by some amount t_0. The end result is the signal shifted to the location of the impulse as if we had moved the x-axis by t_0. This phenomenon is known as the sifting property, because the impulse function $\delta(t - t_0)$ "sifts" through the function $x(t)$ and preserves the value $x(t_0)$.

$$x(t) * \delta(t - t_0) = x(t - t_0) \tag{2.52}$$

Imagine an input signal, $x(t)$, that is a rectangular pulse with bandwidth $2W$ and amplitude, A, centered at $t = 0$. When convolution is performed with the impulse response $\delta(t - t_0)$, one can envision the rectangular pulse passing by the impulse response for all time. The sum of the output of the "sifting" is $y(t)$, which is the input signal $x(t)$ shifted to the location of our impulse, t_0.

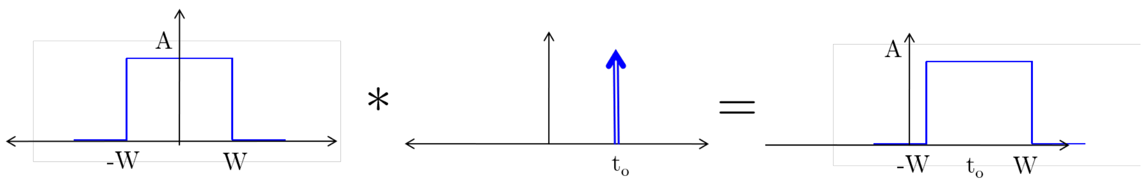


Figure 2.17: Convolving by an impulse copies the other signal to the location of the impulse.

Example 2.9

What is the convolution of two rectangular pulses with width $2W$, one with height A and the other with height B?

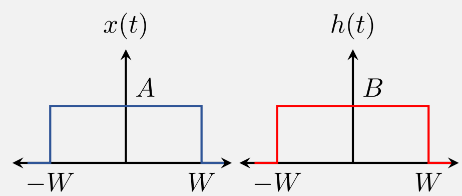


Figure 2.18: Two rectangular pulses with the same width $2W$ but different heights.

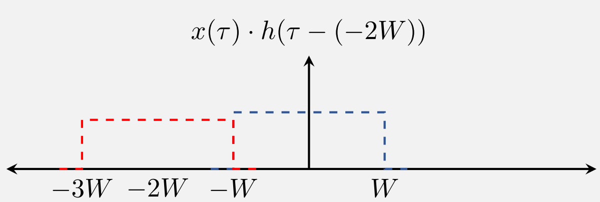
When A equals B, this is the auto-correlation, a fundamental principle in communications theory that we will explore more thoroughly.

Solution:

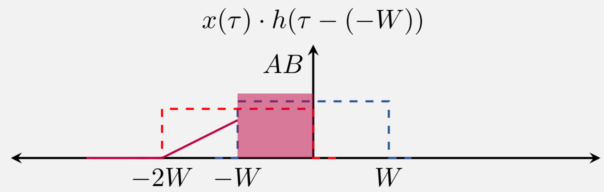
Recalling the definition of convolution from Equation 2.50 and recognizing that $h(t) = h(-t)$, write the convolution equation

$$x * h(t) = \int_{-\infty}^{\infty} x(\tau)h(\tau - t)d\tau$$

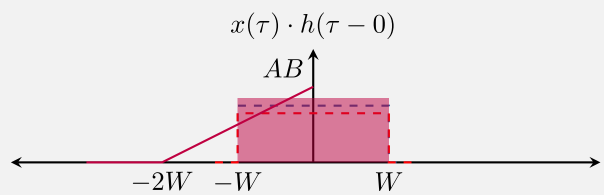
Visualize shifting the second pulse from the left, the product of the two pulses is zero until they begin overlapping. The convolution is the area under this product, so it is also zero. The convolution is zero until the second pulse shifts past $t = -2W$. The convolution $x*h(-2W)$ is illustrated below with $x(t)$ in blue, the shifted $h(t)$ in red, the product $x(\tau)h(\tau - t)$ as a filled purple area, and the convolution for $t \leq -2W$ as a purple line.



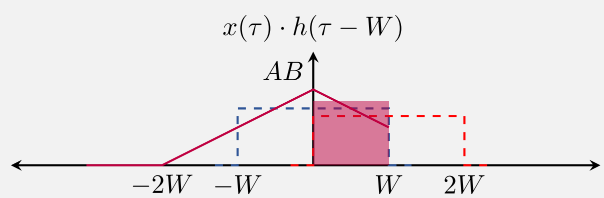
At $t = -W$, the pulse has continued to move right, and the convolution increases linearly, as area of the product increases.

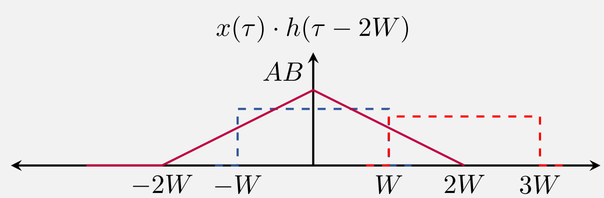


The pulse continues to move right until it is centered at zero. At this moment, the product of the two boxes is a box with width $2W$ and height AB.

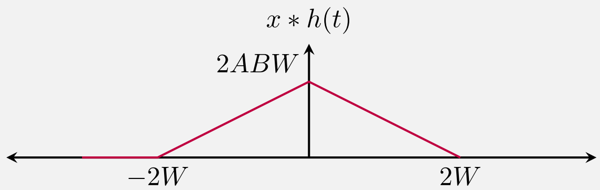


The convolution at zero is the area in purple, $2ABW$. The convolution from $-2W$ to 0 is a line with constant slope because the area of the product, or the overlap, between two pulses linearly increases as the shift, t, approaches zero. After reaching a shift of zero, the previous diagrams progress in reverse as t moves from 0 to $+2W$. The area of the overlap between the two boxes begins to decrease linearly until it reaches zero when the sliding pulse is at $t = 2W$.





The convolution is then the magenta line that tracked the area as t moved from $-2W$ to $+2W$



The output of the convolution operation produces a isosceles triangle with height $2ABW$ and width $4W$.

Example 2.10

What is the output of the LTI system with the following impulse response if the input is $x(t)$?

$$h(t) = 0.1\delta(t) + 0.01\delta(t-4)$$

Solution:

The impulse response, $h(t)$, is the output, $y(t)$, when the input, $x(t)$, is an impulse $\delta(t)$. Substituting $h(t) = y(t)$ and $\delta(t) = x(t)$ gives the output of the LTI system:

$$y(t) = 0.1x(t) + 0.01x(t-4)$$

Example 2.11

What is the impulse response of the following system?

$$y(t) = x(t) - 0.1x(t-5) + 0.05x(t-10)$$

Solution:

The impulse response of $y(t)$ is:

$$h(t) = \delta(t) - 0.1\delta(t-5) + 0.05\delta(t-10)$$

Example 2.12

Given the impulse response, $h(t)$, and input, $x(t)$, find the output, $y(t)$.
Recall that $y(t) = x * h(t)$.

$$h(t) = \frac{\sin(150\pi t)}{\pi t}$$
$$x(t) = \sum_{k=-\infty}^{\infty} \delta(t - 0.02k)$$

Solution:

This problem can be solved in either the time or frequency domain.

Time domain:

$$\begin{aligned} y(t) &= x * h(t) \\ &= \frac{\sin(150\pi(t - 0.02k))}{\pi(t - 0.02k)} \end{aligned}$$

Frequency domain:
$X(f)$ is a train of impulses, where

$$\begin{aligned} X(f) &= \frac{1}{0.02} \sum_{k=-\infty}^{\infty} \delta(f - \frac{k}{0.02}) \\ &= 50 \sum_{k=-\infty}^{\infty} \delta(f - 50k) \end{aligned}$$

$$Y(f) = X(f)H(f), \text{ where } X(f) \text{ is an impulse train and } H(f) \text{ is a box.}$$

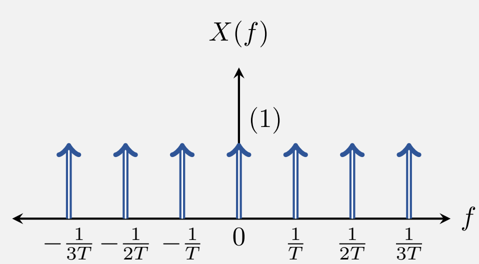


Figure 2.19: The impulse train $X(f)$ in frequency-domain.

Based on the equation of $X(f)$ above, the impulses are separated by 50, meaning $\frac{1}{T} = 50$.

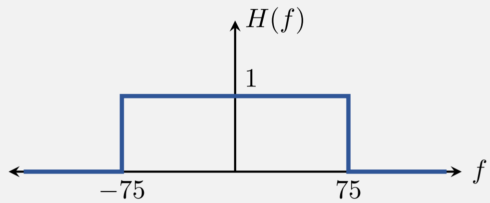
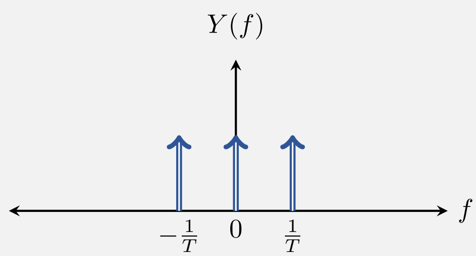


Figure 2.20: The rectangular pulse $H(f)$ in frequency-domain.

$Y(f)$ consists of three impulses within the bounds of $H(f)$ as shown below.



The resulting $y(t)$ is as follows:

$$y(t) = A + \cos(100\pi t)$$

where $A = 1$.

2.4 *Discrete Time (DT) Systems*

Discrete-time (DT) systems operate on DT input signals $x[n]$ and produce DT output signals $y[n]$. $h[n]$ is the unit sample response, and it completely characterizes our system. $h[n]$ is also called the impulse response throughout this text. In DT, the analogue to the unit impulse is the unit sample $\delta[n]$. It is called the Kronecker Delta function, whereas in the continuous time the Dirac Delta function is used.

$h[n]$ **is the output of the system when the input is a DT unit impulse**.

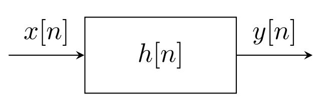


Figure 2.21: A linear time-invariant (LTI) system with input $x[n]$, impulse response $h[n]$, and output $y[n]$.

Discrete-time signals or sequences are only defined for integer values of the argument n. For example, average daily temperatures in Boston are defined for individual days but not at a precise time. The average daily temperature, can thus be viewed as a sampling of the temperature which is varying at every time instant. Similarly, many digital signals can be viewed as DT samples of CT signals, although some are inherently DT.

As such, DT signals or sequences are defined **only** at integer increments of n:

$$..., x[-1], x[0], x[1], x[2], ..., x[n]$$

For these signals, it would not make sense to write $x[1.7]$ because the argument must be an integer.

Difference Equations and Impulse Responses

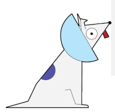
In discrete time, LTI systems are characterized by their **impulse responses or difference equations**. Difference equations provide a mathematical equality involving differences between successive values of a function of a discrete variable, and have the general form of:

$$\sum_{k=-\infty}^{\infty} b_k y[n-k] = \sum_{k=-\infty}^{\infty} a_k x[n-k] \tag{2.53}$$

For example, a specific difference equation might be

$$y[n] = x[n] + 2x[n-1] - 3x[n-2] \tag{2.54}$$

Fun Facts from Doggo



By definition, the impulse response is the output of the system when the input is an impulse. You can always switch between impulse responses and difference equations by letting $x = \delta$ and $y = h$.

The discrete time impulse response $h[n]$ completely characterizes how an LTI system responds to inputs, telling us what output is produced for any input. Setting the input x equal to an impulse δ produces the corresponding impulse response from Equation 2.54:

$$h[n] = \delta[n] + 2\delta[n-1] - 3\delta[n-2] \tag{2.55}$$

The difference equation or impulse response is sometimes written in terms of the sequential coefficients in Equation 2.53. This example would be $a_k = [1, 2, -3]$ from the right-hand side and $b_k = [1]$ from the left-hand side, which implies that the coefficients everywhere else are zero.

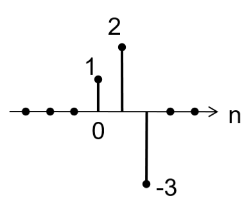


Figure 2.22: The impulse response, $h[n]$, for the difference equation $y[n] = x[n] + 2x[n-1] - 3x[n-2]$ found by setting the input equal to an impulse.

Input-Output for DT LTI Systems

Suppose that we have the following DT system and wish to find its output $y[n]$ for input $x[n]$

$$h[n] = \delta[n] + 2\delta[n-1] - 3\delta[n-2] \tag{2.56}$$
$$x[n] = \delta[n] - \delta[n-1] \tag{2.57}$$

The input is depicted in Figure 2.23.

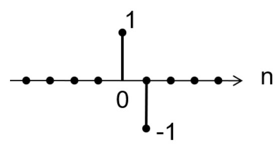


Figure 2.23: Example input $x[n] = \delta[n] - \delta[n-1]$ for a DT system.

Since the system is LTI, we find $y[n]$ by considering superposition of the two impulses of $x[n]$, the input consisting of a basic impulse and a delayed, scaled impulse. If that is confusing, remember that the superposition principle implies linearity and suggests that if we input x_1 into our system it will return y_1, and if we input x_2 our system will return y_2. Furthermore if we input the sum of x_1 and x_2, the output should be the sum of the individual responses, y_1 and y_2.

The basic impulse produces an impulse response as output, and the delayed, scaled impulse produces a delayed, scaled impulse response. We can consider the system as if we pushed the input through each of these impulses separately, then added the results together. This is convolution! In MATLAB, this can be performed with **conv([-1 1], [1, 2, -3])**. This is depicted visually in Figure 2.24.

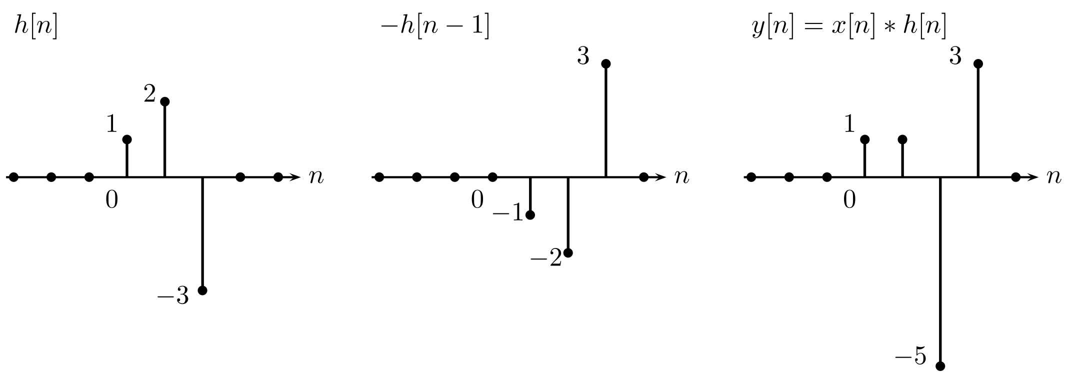


Figure 2.24: The impulse response $h[n]$ and negated, delayed impulse response, which sum to produce output $y[n]$ of a discrete LTI system.

For the input $x[n] = \delta[n] - \delta[n-1]$, the output $y[n]$ is the sum of the regular impulse response and the delayed, negated impulse response. $y[n]$ is obtained by substituting $\delta[n]$ with $h[n]$ and $-\delta[n-1]$ with $-h[n-1]$ and summing the results together.

DT Fourier Transform

The discrete time Fourier Transform (DTFT) is like the continuous time Fourier Transform, but it utilizes a summation instead of an integral due to the discrete nature of the signals.

$$X(e^{j\omega}) = \sum_{k=-\infty}^{\infty} x[k]e^{-j\omega k} \tag{2.58}$$

As previously discussed, it is helpful to note that ω, measured in radians per second, may also be used to represent the angular frequency.

$$\omega = 2\pi f \tag{2.59}$$

Discrete Time Fourier Transform Properties

The DTFT has its own set of properties that mirrors those of the CTFT. In these equations, the notation $< 2\pi >$ indicates that the relationship holds for any integral interval of width 2π. You may sometimes see normalized angular frequency Ω, which is ω over the domain $[-1, 1]$ such that $X(e^{j\omega}) = X(\Omega)$.

1. A shift in the time domain produces a phase shift in the frequency domain and vice-versa

$$x[n - n_0] \longleftrightarrow e^{-j\omega n_0} X(e^{j\omega}) \tag{2.60}$$

$$e^{j\omega_0 n} x[n] \longleftrightarrow X(e^{j(\omega-\omega_0)}) \tag{2.61}$$

2. Convolution in the time domain is multiplication in the frequency domain and vice-versa

$$x[n] * y[n] \longleftrightarrow X(e^{j\omega})Y(e^{j\omega}) \tag{2.62}$$

$$x[n]y[n] \longleftrightarrow \frac{1}{2\pi}\int_{<2\pi>} X(e^{j\theta})Y(e^{j(\theta-\omega)})d\theta \tag{2.63}$$

3. The power content of a signal is conserved

$$\sum_{n=-\infty}^{\infty} |x[n]|^2 \longleftrightarrow \frac{1}{2\pi}\int_{<2\pi>} |X(e^{j\omega})|^2 d\omega \tag{2.64}$$

4. A linear scaling in time by a corresponds to a linear scaling in the frequency domain by $|\frac{1}{a}|$. For example, scaling occurs when an impulse train with period T in time domain is spaced by $\frac{1}{T}$ in the frequency domain.

$$x(an) \longleftrightarrow \frac{1}{|a|} X\left(\frac{f}{a}\right), \text{ where } f = j\omega \tag{2.65}$$

5. A linear combination of time-domain functions produces the same linear combination of frequency domain functions

$$x_1[n] \longleftrightarrow X_1(f) \tag{2.66}$$

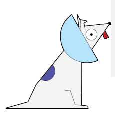
$$x_2[n] \longleftrightarrow X_2(f) \tag{2.67}$$

$$\alpha x_1[n] + \beta x_2[n] \longleftrightarrow \alpha X_1(f) + \beta X_2(f) \tag{2.68}$$

Because **the DTFT is periodic with period** 2π, it only needs to be specified on the interval from $-\pi$ to π, making the domain $\omega \in [-\pi, \pi]$. The periodicity is proven below:

$$\begin{aligned} X(e^{j(\omega+2\pi)}) &= \sum_{k=-\infty}^{\infty} x[k]e^{-j(\omega+2\pi)k} \\ &= \sum_{k=-\infty}^{\infty} x[k]e^{-j\omega k}e^{-j2\pi k} \\ &= \sum_{k=-\infty}^{\infty} x[k]e^{-j\omega k} \\ &= X(e^{j\omega}) \end{aligned}$$

Fun Facts from Doggo



When referring to frequency in **continuous time**, the upper-case omega symbol, Ω, is used. And when referring to frequency in **discrete time**, the lower-case omega symbol, ω, is used.

Example 2.13

Find the DTFT of $x[n] - x[n-1]$ *in terms of* $X(e^{j\omega})$.

Solution:

$$\begin{aligned} &X(e^{j\Omega}) - X(e^{j\Omega})e^{-j\Omega n_0} \text{ where } n_0 = 1 \\ &X(e^{j\Omega}) - X(e^{j\Omega})e^{-j\Omega} \\ &X(e^{j\Omega})(1 - e^{-j\Omega}) \end{aligned}$$

Consider the DTFT of the sinc function, which is a single box in the CTFT. Due to the periodic nature of the DTFT, the result in discrete time is multiple boxes at intervals of 2π as shown in Figure 2.25.

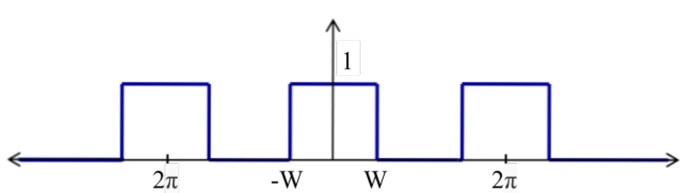


Figure 2.25: The DTFT of $\frac{\sin(Wn)}{\pi n}$ is repeated boxes because the DTFT is periodic. This contrasts the single box in the CTFT.

The DTFT of $\frac{\sin(2\pi Wn)}{\pi n}$ is multiple rectangular pulses at intervals of 2π, and the Fourier Transform pair for $\left(\frac{\sin(2\pi Wn)}{\pi n}\right)^2$ is the self-convolution of these rectangular pulses. The self-convolution of a rectangular pulse is an isosceles triangle with base from $-2W$ to $2W$ and height of $2W$. Because the DTFT is periodic, finding the DTFT of $\left(\frac{\sin(2\pi Wn)}{\pi n}\right)^2$ only requires convolving a single rectangular pulse with the trail of rectangular pulses. Passing a single rectangular pulse by a train of rectangular pulses (or functions) is an example of what is referred to as **circular convolution**. The circular convolution of two periodic signals is the linear convolution of one of the periodic signals with a single period of the other. Further, we have used the property that in DT multiplication in time becomes circular convolution in frequency.

Example 2.14

Find the DTFT of $\left(\frac{\sin(\frac{\pi}{2}n)}{\pi n}\right)^2$

Hint: when solving this example, remember Equation 2.63

$$x[n]y[n] \longleftrightarrow \frac{1}{2\pi}\int_{<2\pi>} X(e^{j\theta})Y(e^{j(\theta-\omega)})d\theta \tag{2.69}$$

and that solving this DTFT requires convolving only one 2π period of one function with the entirety of other function.

Solution:

Figure 2.26 below depicts a rectangular pulse train.

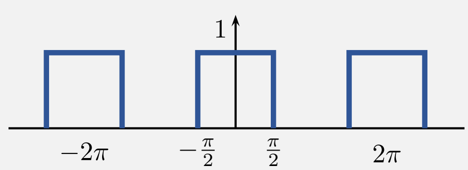


Figure 2.26: A rectangular pulse train.

And Figure 2.27 depicts a single rectangular pulse.

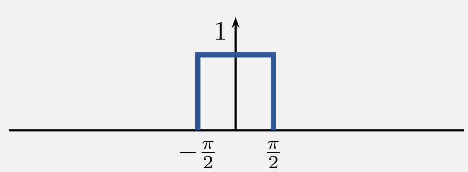


Figure 2.27: A single rectangular pulse.

When we convolve the two, we produce the following triangular pulse train:

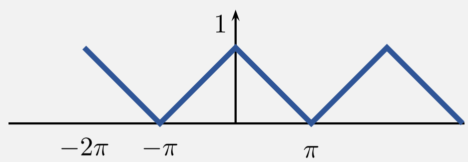


Figure 2.28: The resulting triangular pulse train.

Discrete Time Fourier Transform Pairs

As you saw the CTFT transform pairs in Figure 2.13 with the sinc-box pair and the impulse train pair, there are also transform pairs in DTFT.

$$\delta[n] \longleftrightarrow 1 \tag{2.70}$$

$$e^{jw_0 n} \longleftrightarrow 2\pi \sum_{k=-\infty}^{\infty} \delta(\omega - \omega_0 - 2\pi k) \tag{2.71}$$

$$\cos(w_0 n) \longleftrightarrow \pi \sum_{k=-\infty}^{\infty} [\delta(\omega - \omega_0 - 2\pi k) + \delta(\omega + \omega_0 - 2\pi k)] \tag{2.72}$$

$$\sin(w_0 n) \longleftrightarrow j\pi \sum_{k=-\infty}^{\infty} [\delta(\omega - \omega_0 - 2\pi k) + \delta(\omega + \omega_0 - 2\pi k)] \tag{2.73}$$

2.5 *Problem Set 0: Signals Review*

Problem	Topic	Points
1	Steps for Proving Linearity and Time Invariance (LTI)	1
2	Proving LTI	1
3	Steps for Converting from the Time to Frequency Domain	1
4	FT Pairs	1
5	Dirac Delta	1
6	Euler's Equation and Sinusoids	1
7	Sinc	1
8	Impulse Train	1
Total:		8

Exercise 2.1

(1 point) Describe the steps for determining whether a system is Linear Time Invariant (LTI). Include equations to help aid your descriptions.

Exercise 2.2

(1 point) Use the framework described in Exercise 2.1 to show whether the following systems are LTI:

a) $y(t) = 12x(t) - 3$

b) $y(t) = x^3(t)$

c) $y[n] = \frac{x[n]}{\sin(3\pi n)}$

Exercise 2.3

(1 point) What are the steps for converting a function in the time domain to the frequency domain? That is, what are the steps for performing a Fourier Transform (FT)? Include a description and equations in your response.

Exercise 2.4

(1 point) List the equations and sketch figures of four FT pairs in continuous time frame. How are these different if we wanted to represent them in discrete time?

Exercise 2.5

(1 point) What is the definition of the Dirac Delta function in both the time and frequency domain? Please include a sketch, and label the axes, height, and area.

Exercise 2.6

(1 point) Using Euler's equation, derive the Fourier transform of $\cos(\theta)$ and $\sin(\theta)$.

Exercise 2.7

(1 point) Sketch and write the equation for the sinc function. Label the values of zero crossings on your figure.

Exercise 2.8

(1 point) Sketch and write the equation for an impulse train with (1/T) = 100. By how much are the impulse trains separated?

2.6 *Digital Processing and Sampling*

Many signals in the real world are continuous in nature, but are often measured, processed and recorded in discrete time (DT). This is because DT processing is often cheaper, more flexible, and can be more accurate given that modern computers all perform mostly DT math operations. Fortunately, DT operations can be performed at a fine enough scale such that the difference from CT is negligible, in most applications. In communications systems, CT signals are converted into DT via an Analog to Digital Converter (ADC), so that processing can be done digitally. Then we convert from DT back to CT using a Digital to Analog Converter or a DAC. Filtering audio signals, digitizing music from CD storage, and taking an image on your cellphone are examples of processes that convert between the analog and digital domains.

Mathematically, these systems can be analyzed by multiplying the CT signal, $x(t)$, by a unit impulse train, $p(t)$, with sampling period T, where $p(t)$ is defined as

$$p(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT) \tag{2.74}$$

This produces an impulse train where the area of each impulse corresponds to the value of the function at that point in time as depicted in Figure 2.29.

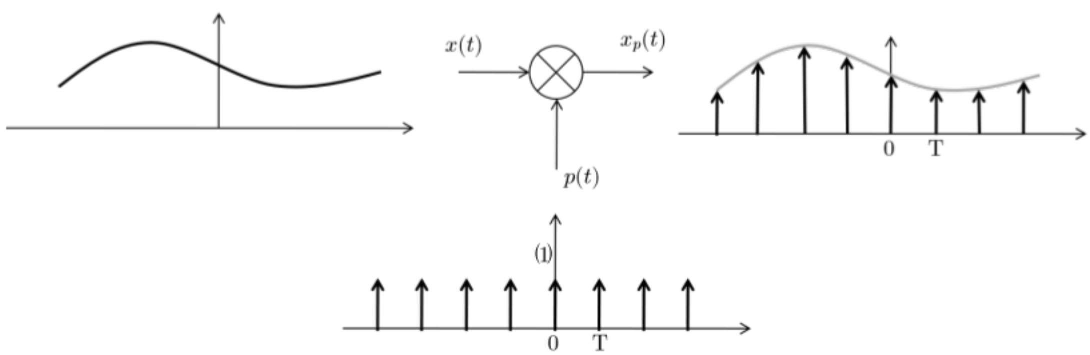


Figure 2.29: The sampled version of $x(t)$ is depicted in CT as an impulse train following the curve, which is acquired by multiplying $x(t)$ and a unit impulse train $p(t)$.

Figure 2.30 provides a block diagram of process beginning with sampling a continuous time signal, $x(t)$, to produced $x_p(t)$, then converting that sampled signal to a fully digital signal, $x[n]$, and processing it to produce an output signal $y[n]$. After the output signal is obtained, it is converted to a continuous signal, $y_p(t)$ and filtered.

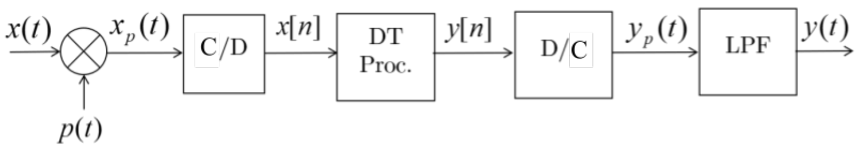


Figure 2.30: A Communications Schematic of a continuous time input signal, $x(t)$ to a filtered output $y(t)$

Nyquist Sampling Theorem

The rate at which we sample our signal dictates how accurately we can recover a signal. When our sampling frequency is sufficiently higher than the frequencies present in the signal, mathematically, there is no loss of information. To show this, we sketch our signal in both time and frequency:

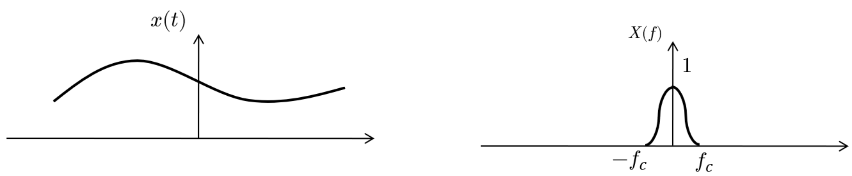


Figure 2.31: CT Signal, $x(t)$, in the time domain (left) and converted to $X(f)$ and represented in the frequency domain (right).

Remember that FT functions are sometimes in terms of frequency, Hz, and will use notations written as $X(f)$ and other times they are in terms of angular frequency, and use $X(e^{j\omega})$.

The impulse train representing our sampled signal is also an impulse train in the frequency domain, but the impulses have area of $\frac{1}{T}$ for all frequencies and are spaced by the sampling rate $f_s = \frac{1}{T}$. As shown in Figure 2.32, increasing the spacing between impulses in the time domain corresponds to a decrease in the spacing between impulses in the frequency domain. Due to scaling, you might see a spacing of $\omega_o = \frac{2\pi}{T}$.

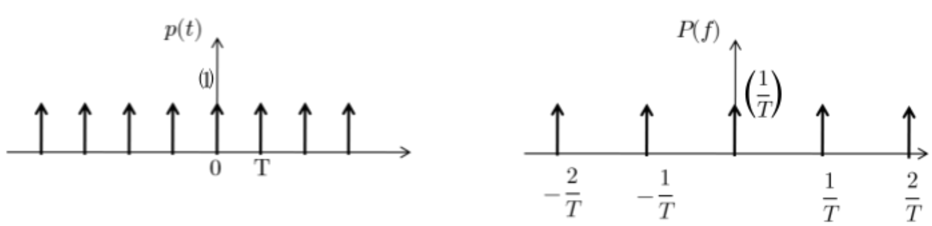


Figure 2.32: Impulse train in the time (left) and frequency (right) domain

The sampled result in the frequency domain is the input signal, $x(t)$, convolved by the new impulse train. As shown in Figure 2.33, this produces the sampled signal $x_p(t) = \sum_{n=-\infty}^{\infty} x[n]\delta(t - nT)$ and when mapped to the frequency domain places a copy of the sketched "blob" at every $\frac{1}{T}$ interval scaled by $\frac{1}{T}$.

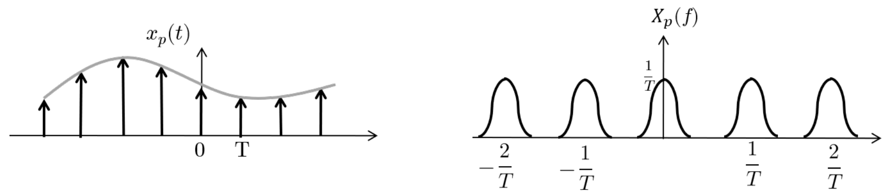


Figure 2.33: Sampled result in the time (left) and frequency (right) domain

From Figure 2.34, the next step in **fully** recovering our original CT signal involves utilizing a CT low pass filter (LPF) that reduces the blob train to a single blob centered at $f = 0$ (the original).

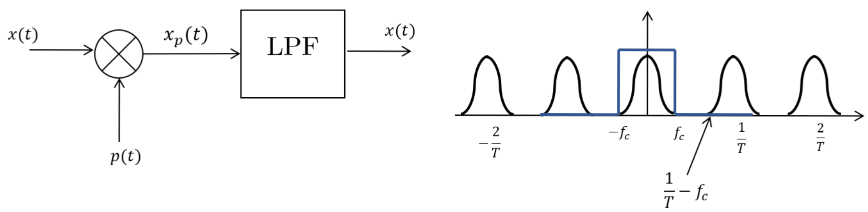


Figure 2.34: The inclusion of a low pass filter (LPF) in a sampled LTI system (left) and the resulting fully recovered original CT signal shown centered around $f = 0$) (right)

If the blobs are not spaced out as shown in Figure 2.34, or mathematically speaking, if the maximum frequency, f_M, is greater than $\frac{1}{T} - f_M$, i.e., $2f_M > \frac{1}{T}$, then aliasing occurs. Aliasing represents a loss of information in the region where the blobs overlap, which is shown in Figure 2.35 below. When the spacing is not large enough due to a low sampling rate, the frequency domain blobs overlap with each other. Where the regions overlap and add together, the original signal cannot be discerned.

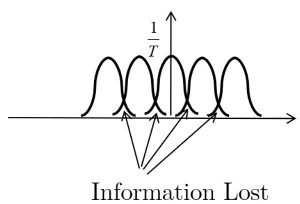


Figure 2.35: Aliasing due to low sampling rate

This defines the fundamental limit of sampling, known as the Nyquist Criteria for Zero Aliasing: **the sampling frequency f_s must be strictly greater than twice the maximum frequency of the signal or information is lost**.

$$2f_M < f_s = \frac{1}{T} \tag{2.75}$$

The **Nyquist frequency**, $2f_M$ is a lower bound on the sampling frequency, f_s, required to completely recover the sampled signal, and prevent aliasing. If the inequality doesn't hold, then we can't reconstruct our signal without the loss of information! Keep in mind that f_s is equivalent to $\frac{1}{T}$.

Example 2.16

If a signal has a maximum frequency between 2000 Hz to 8000 Hz, what is the minimum sampling rate that you can have to ensure zero aliasing? Support your answer with a drawing.

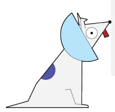
Solution:

In order to have zero aliasing, the sampling frequency must be greater than two times the maximum frequency in the signal, as stated by Nyquist Sampling Theorem in Equation 2.75:

$$2f_c < f_s$$

In this case, the maximum frequency in the signal is 8000 Hz, meaning that the minimum sampling frequency must be **greater than (and NOT equal to) 16000 Hz**.

Fun Facts from Doggo



In the real world, the sample rate of audio CDs is 44.1 kHz, which is a little greater twice the maximum frequency of a human's range of hearing: 20 Hz to 20 kHz! The extra 4 kHz allows for extra inaudible information to be stored and to allow for non-ideal filters.

Digital Filters

Earlier, we covered difference equations (or impulse responses) of DT LTI systems. Many digital processes and filters are examples of DT LTI systems, though they are often not ideal. Consider the ideal low pass & high pass filters in Figure 2.36 and their non-ideal, but still optimistic, counterparts.

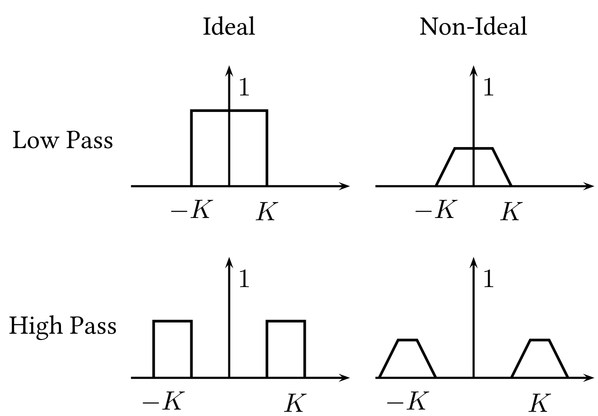


Figure 2.36: Digital low pass and high pass filters shown represented in ideal and non-ideal forms

It is helpful to have intuition for what a DT filter might do using only the difference equation or impulse response. Two examples of difference equations are provided in Equations 2.76 and 2.77.

Equation 2.76 shows a three-point moving average and represents a low pass filter.

$$y[n] = \frac{1}{3}x[n-1] + \frac{1}{3}x[n] + \frac{1}{3}x[n+1] \tag{2.76}$$

Equation 2.77 shows numeric differentiation and represents a high-pass filter.

$$y[n] = x[n] - x[n-1] \tag{2.77}$$

What might each of these do to the frequency content of a signal? If we were to provide the moving average a sequence of constant values, it would be unchanged. On the other hand, a sinusoidal signal would smooth out to a lower amplitude. **The moving average is a form of a low pass filter**.

Similarly, the constant sequence would disappear completely if passed through the differentiating filter. The sinusoidal signal would experience little attenuation since the derivative of a sinusoidal signal is only a phase shift. A phase shift on a sinusoid is a delay of the sinusoid. More generally, phase shifts of filters are frequency dependent, and represent different delays that different frequency components experience when acted upon by the filter.

Figure 2.37 below depicts the frequency response for a three-point moving average and low pass filter represented by Equation 2.76.

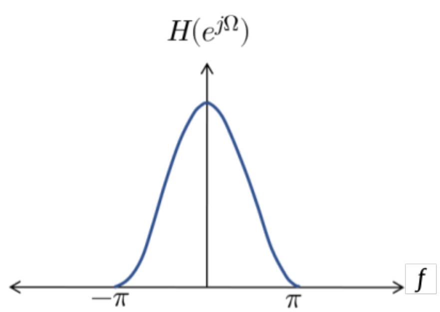


Figure 2.37: Frequency Response for $y[n] = \frac{1}{3}x[n-1] + \frac{1}{3}x[n] + \frac{1}{3}x[n+1]$, a Low Pass Filter

And Figure 2.38 below depicts the frequency response for numeric differentiation and high-pass filter represented by Equation 2.77.

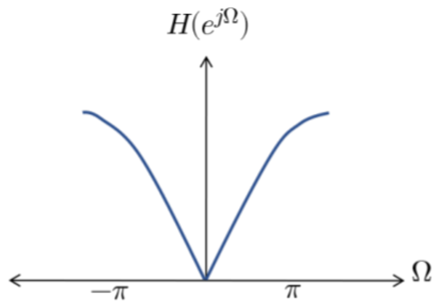


Figure 2.38: Frequency Response for $y[n] = x[n] - x[n-1]$, a High-Pass Filter

Next, we will encounter an example of a difference equations and their applications to filters int eh time and frequency domain.

Example 2.17

Given the difference equation

$$y[n] = \frac{1}{2}x[n+1] + x[n] + \frac{1}{2}x[n-1]$$

What are $h[n]$ and $H(e^{j\Omega})$? Is the system high pass, low pass, or neither?

Solution:

The impulse response of the system is $h[n]$, so replace $x[n]$ with an impulse $\delta[n]$.

$$h[n] = \frac{1}{2}\delta[n+1] + \delta[n] + \frac{1}{2}\delta[n-1]$$

The frequency response is the DTFT of the impulse response. The following two pairs are needed in this case:

$$\delta[n] \longleftrightarrow 1$$
$$a\delta[n-n_0] \longleftrightarrow ae^{-j\omega n_0}$$

The DTFT of the impulse reponse is then

$$H(e^{j\Omega}) = \frac{1}{2}e^{-j\omega} + 1 + \frac{1}{2}e^{j\omega}$$

This equation may not be intuitive, but the substitution of Euler's equation

$$\cos(\theta) = \frac{1}{2}(e^{j\theta} + e^{-j\theta})$$

yields the following frequency response:

$$H(e^{j\Omega}) = 1 + \cos(\omega)$$

The DTFT is periodic over the interval $[-\pi, \pi]$. Considering the frequency response over this interval, the response at zero is $1+\cos(0) = 2$. Cosine decreases to -1 in either direction towards $\pm\pi$, so the response decreases to $1+\cos(\pi) = 0$. This means that higher frequencies are attenuated, or weakened, while lower frequencies are not. Therefore, this is a **low pass filter**.

In the previous example, you solved for the impulse response and frequency response given the difference equation. In this next example, you will solve for the difference equation given the frequency response.

Example 2.18

Given the frequency response

$$\begin{aligned} H(e^{j\omega}) &= \frac{(1 - e^{j\omega_0}e^{j\omega})(1 - e^{j\omega_0}e^{-j\omega})}{(1 - qe^{j\omega_0}e^{j\omega})(1 - qe^{j\omega_0}e^{-j\omega})} \\ &= \frac{1 - 2\cos\omega_0 e^{-j\omega} + e^{-j2\omega}}{1 - 2q\cos\omega_0 e^{-j\omega} + q^2 e^{-j2\omega}} \end{aligned}$$

- *What is the difference equation?*
- *Suppose $|q| < 1$. What type of filter is it when $q \approx 1$?*

Solution:

Recalling Equation 2.51 we write

$$Y(e^{j\omega}) = X(e^{j\omega})H(e^{j\omega}) \longrightarrow H(e^{j\omega}) = \frac{Y(e^{j\omega})}{X(e^{j\omega})} \tag{2.78}$$

With this in mind, we rearrange the equation to

$$(1 - 2q\cos\omega_0 e^{-j\omega} + q^2 e^{-j2\omega})H(e^{j\omega}) = 1 - 2\cos\omega_0 e^{-j\omega} + e^{-j2\omega} \tag{2.79}$$

Next, we take the inverse DTFT of each term using DT transform pairs

$$(\delta[n] - 2q\cos\omega_0\delta[n-1] + q^2\delta[n-2]) * h[n] = \delta[n] - 2\cos\omega_0\delta[n-1] + \delta[n-2] \tag{2.80}$$

Convolution is distributive, so we move $h[n]$ inside to each term

$$\delta[n]*h[n] - 2q\cos\omega_0\delta[n-1]*h[n] + q^2\delta[n-2]*h[n] = \delta[n] - 2\cos\omega_0\delta[n-1] + \delta[n-2] \tag{2.81}$$

By the definition of impulse response, $h[n]$ is the output $y[n]$ when the input $x[n]$ is an impulse $\delta[n]$. We can replace every $\delta[n]$ with $x[n]$

$$x[n]*h[n] - 2q\cos\omega_0 x[n-1]*h[n] + q^2 x[n-2]*h[n] = x[n] - 2\cos\omega_0 x[n-1] + x[n-2] \tag{2.82}$$

Since, $x[n-n_0]*h[n] = y[n-n_0]$, we can write the left side in terms of $y[n]$. This gives the **difference equation**

$$y[n] - 2q\cos\omega_0 y[n-1] + q^2 y[n-2] = x[n] - 2\cos\omega_0 x[n-1] + x[n-2] \tag{2.83}$$

If $|q| < 1$, when $q \approx 1$, then $|H(e^{j\omega})|$ becomes

$$H(e^{j\omega}) = \frac{(1 - e^{j\omega_0}e^{j\omega})(1 - e^{j\omega_0}e^{-j\omega})}{(1 - qe^{j\omega_0}e^{j\omega})(1 - qe^{j\omega_0}e^{-j\omega})} \tag{2.84}$$

Since $|q|$ is not exactly 1, the denominator is not exactly zero, but $|q|$ is close enough that the magnitude of $H(e^{j\omega})$ is about 1 for all frequencies except ω_0. When $\omega = \pm\omega_0$, one of the factors in the numerator is exactly zero while the similar factor in the denominator is only approximately zero, setting the response to exactly zero for those values of ω. If $|q|$ were exactly 1, then the response at $\omega = \pm\omega_0$ would be undefined. **The response magnitude is approximately constant for all frequencies except near $\pm\omega_0$, so this is a notch filter!**

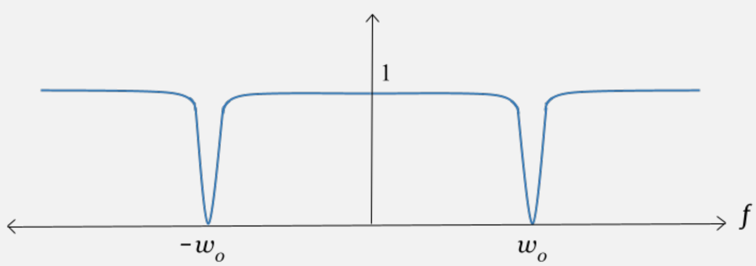


Figure 2.39: The frequency response of a notch filter. Depending on how close $|q|$ is to 1, the notch can be narrowed or widened to remove a specific frequency.

An example of when a notch filter could be used is if a signal is corrupted by a tone picked up from the electrical grid. In most parts of the world, the culprit frequency is either 50 Hz or 60 Hz.

2.7 *Analog to Digital Conversion*

In analog to digital conversion, the sampled signal, $x_p(t)$, is converted to a digital signal, $x[n]$. The signal, $x_p(t)$, is constructed with impulses while $x[n]$ is constructed with regular, digitally processed points. It's important to note that some texts interchange $x_s(t)$ and $x_p(t)$, but there is not a standard notation. An equation form of $x_p(t)$ and its CTFT, $X_p(f)$, are useful in analyzing analog to digital conversion.

$$x_p(t) = x(t)p(t) \tag{2.85}$$

The sampled signal, $x_p(t)$, is acquired by multiplying $x(t)$ by an impulse train with period T. .

$$x_p(t) = x(t) \sum_{n=-\infty}^{\infty} \delta(t - nT) \tag{2.86}$$

Figure 2.40 depicts the discretization of our sampled signal, $x_p(t)$.

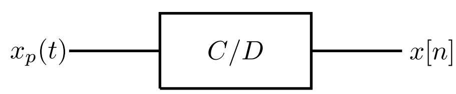


Figure 2.40: Block Diagram of Discretized Sampled Signal

We only need the values of $x(t)$, which occur at the same time as an impulse, so we replace $x(t)$ with $x[n] = x(nT)$ inside the summation.

$$x_p(t) = \sum_{n=-\infty}^{\infty} x[n]\delta(t - nT) \tag{2.87}$$

We perform the CTFT by applying Equation 2.8

$$X_p(f) = \int_{-\infty}^{\infty} \left(\sum_{n=-\infty}^{\infty} x[n]\delta(t - nT) \right) e^{-j2\pi f t} dt \tag{2.88}$$

and since $x[n]$ does not depend on t, it and the summation operator are moved outside the integral. This leaves the CTFT of an impulse train, which is easily found with a table of CT Fourier Transform Pairs.

$$\begin{aligned} X_p(f) &= \sum_{n=-\infty}^{\infty} x[n] \int_{-\infty}^{\infty} \delta(t - nT) e^{-j2\pi f t} dt \\ &= \sum_{n=-\infty}^{\infty} x[n] e^{-j2\pi f n T} \end{aligned} \tag{2.89}$$

Here, we have shown that

$$X_p(f) = X(e^{j\Omega})\big|_{\Omega = 2\pi f T} \tag{2.90}$$

$$X(e^{j\Omega}) = X_p(f)\big|_{f = \frac{\Omega}{2\pi T}} \tag{2.91}$$

The difference between these two equations comes down to units: frequency, f, has units of Hz or cycles per second while angular frequency, Ω, is unitless. This has the simple effect of rescaling the frequency axis, but is also the relationship formed by C/D conversion, as depicted in Figure 2.41. In other words, we have shown how an C/D converter converts $x_p(t)$ to $x[n]$ in the frequency domain.

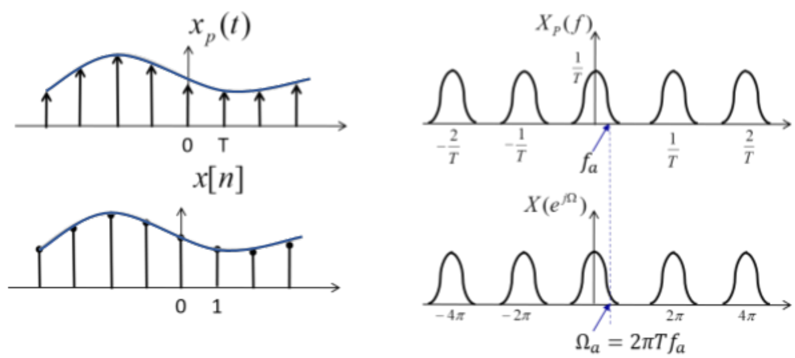


Figure 2.41: Sampled signal, $x_p(t)$, converted to a digital signal, $x[n]$, with the resulting frequency domain representations.

Quantization

Analog to digital conversion involves both conversion from CT to DT and **quantization**. In the previous section, we discretized time to produce a DT signal and related CT to DT in angular frequency. In doing so, it was implicitly assumed the amplitude could still take on a continuum of values. When a number is stored for DT calculations, it is also **quantized** to take the closest value from finite set of values determined by the number of bits (i.e. binary digits) available.

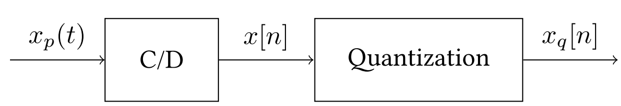


Figure 2.42: Block Diagram Including Quantization after Continuous to Discrete Conversion

At $x[n]$ in Figure 2.42, the amplitude can take a continuum of values, and at $x_q[n]$, the amplitude is discretized, meaning that it is mapped onto known values for modulation.

A simplified model of the effect this has on the signal is to assume the process of quantization is like additive noise, **quantization noise**. This is to say quantization will randomly change values higher or lower as the continuum is rounded to the finite set of values, introducing error in the measurement. A depiction of this addition is provided in Figure

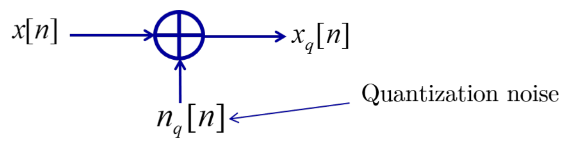


Figure 2.43: Errors modeled as additive noise included in the quantized signal

The accuracy of digital samples of an analog DT signal, such as in Figure 2.44, can be improved with higher order modulation schemes, which utilize more bits per quantization level. As accuracy is improved, the storage and data rate requirements also increase. We have just scraped the surface of quantization here, but will explore this topic more in Part III of this textbook.

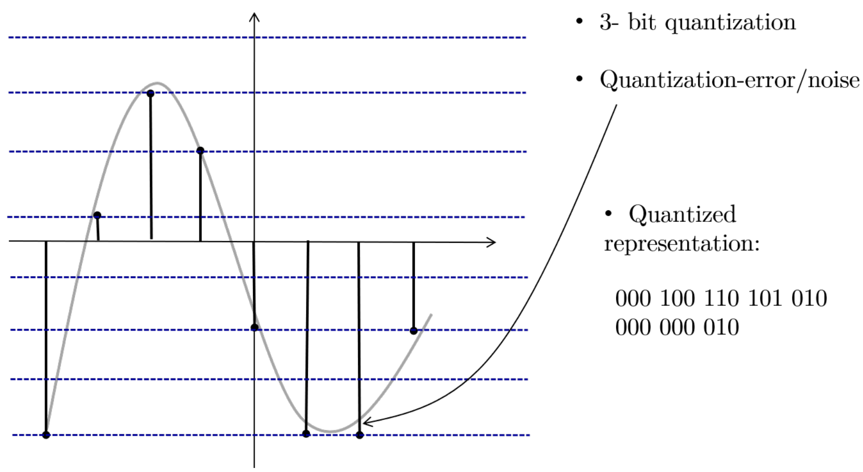


Figure 2.44: A digitally sampled signal comes with an error as it is mapped onto a finite set of values. The error is sometimes modeled as quantization noise.

2.8 *Problem Set 1: LTI Systems and the Fourier Transform*

Problem	Topic	Points
1	Proving Linearity and Time Invariance	3
2	Introduction to Transmitters and Receivers	3
3	Input/Output of LTI Systems	3
4	Continuous Time Fourier Transform Sketches	3
5	Impulse Trains and Square Wave Fourier Transforms	3
6	Manipulating Discrete Time Signals and Systems	3
Total:		18

Exercise 2.9

(3 points) Consider a system with input $x(t)$ and output $y(t)$. For the following examples, please determine if the system is Linear, Time-Invariant, or both (LTI).

(a) $y(t) = 3x(t) + 4$.

(b) $y(t) = 3x(t) + 2x(t-1)$.

(c) $y(t) = x(t)\cos(2\pi f t)$

Exercise 2.10

(3 points) In this problem, you will analyze a simplified model of two different transmitters communicating with two different receivers, e.g. two radio stations that are received by two different people. Figure 2.45 illustrates this system. Note: You may not have seen the exact steps for setting up this problem, but you have seen each of the parts, which you can piece together!

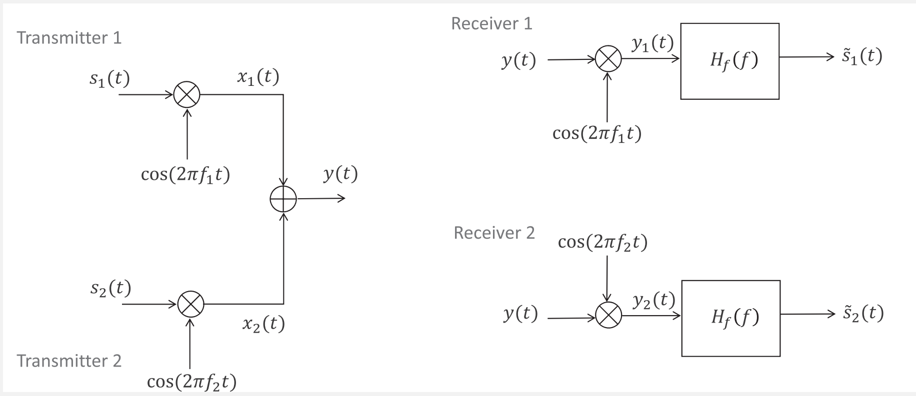


Figure 2.45: A communications system with two transmitters and two receivers

The systems transfer function is provided in Figure 2.46.

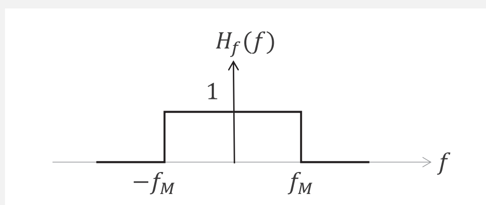


Figure 2.46: The transfer function of the system.

Transmitter 1 wishes to communicate a signal $s_1(t)$ by modulating it with $\cos(2\pi f_1 t)$ and transmitting the resulting signal $x_1(t)$ on its antenna. Transmitter 2 wishes to communicate a signal $s_2(t)$ by modulating it with $\cos(2\pi f_2 t)$ and transmitting the resulting signal $x_2(t)$. Let the frequency-domain versions of $s_1(t)$ and $s_2(t)$ be as shown in Figure 2.47.

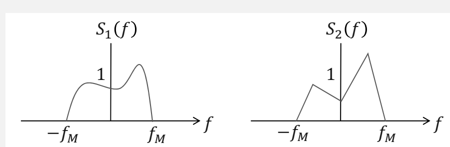


Figure 2.47: $S_1(f)$ and $S_2(f)$

By superposition, the two signals add in free space. For simplicity, call the resulting signal $y(t)$. Receiver 1 and Receiver 2 try to recover the signals $s_1(t)$ and $s_2(t)$ from $y(t)$.

a) Sketch $X_1(f)$ and $X_2(f)$ using the representative frequency domain blobs in Figure 2.47.

b) Find a relationship between f_1, f_2 and f_M such that the signals from the two transmitters don't interfere with each other. Assume that $f_2 > f_1$.

c) Assuming that linearity holds, please sketch $Y_1(f)$ and $Y_2(f)$ and describe why $\tilde{s}_1(t) = \frac{1}{2}s_1(t)$ and $\tilde{s}_2(t) = \frac{1}{2}s_2(t)$.

d) In the North American 2.4 GHz, 802.11 WiFi standard the range of frequencies used is 2.401GHz - 2.473GHz. The bandwidth of the WiFi signal is 22MHz (i.e $f_M = 11MHz$). How many independent WiFi systems can operate in the same physical location without overlapping? For further related information, you can read about 802.11 frequency bands at 2.4 GHz Wi-Fi Bands.

Exercise 2.11

(3 points) Consider a linear, time-invariant system which produces the output $y(t)$ when the input is $x(t)$ as shown in Figure 2.48.

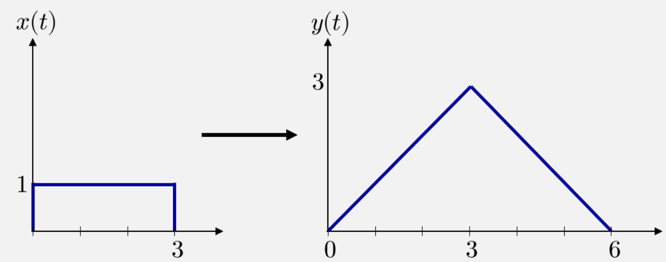


Figure 2.48: Input, $x(t)$, and output, $y(t)$ of a LTI System

a) Sketch the output of the system if the input signal is given by $x(t)$ below. Hint: Given that linearity holds, think about how our input compares to that of the LTI system shown in the figure above. What does this imply about the output?

$$x(t) = \begin{cases} 2, & 0 \leq t < 6 \\ 0, & \text{otherwise} \end{cases}$$

b) Sketch the output of the system with the input, $x(t)$, defined below. Hint: You will need to determine h(t) based off the system shown in 2.48, and take a similar approach based on linearity as you did in the previous part of this problem!

$$x(t) = \begin{cases} 1, & 0 \leq t < 1 \\ 0, & \text{otherwise} \end{cases}$$

Exercise 2.12

(3 points) Consider a function $w(t) = \frac{sin(2\pi Bt)}{\pi t}$ and assume that $B \ll f_C$. Use the properties of Fourier transforms to answer the following. Include amplitudes in your Fourier transform plots.

a) Plot the Fourier transform of $w^2(t)$.

b) Plot the Fourier transform of $w^2(t)cos(2\pi f_C t)$.

c) Plot the Fourier transform of $w^2(t)cos^2(2\pi f_C t)$.

d) Suppose that $x(t) = w^2(t)cos^2(2\pi f_C t)$ and $y(t) = w * x(t)$. Plot the Fourier transform of $y(t)$.

Exercise 2.13

(3 points) With reference to the Fourier transform of an impulse train,

a) plot the Fourier transform of a unit-amplitude, 100 Hz, square wave with 50% duty cycle and a DC offset of 0.5 (i.e., the square wave goes from 0 to 1).

b) plot the Fourier transform of a unit-amplitude, 100 Hz, square wave with 50% duty cycle and no DC offset (i.e., the square wave goes from -0.5 to 0.5).

Exercise 2.14

(3 points) Consider a Discrete-time, Linear-Time-Invariant (LTI) system with input $x[n]$ and output $y[n]$ which are related by the following difference equation.

$$y[n] = x[n] + 2x[n-1]$$

(a) What is the unit sample/impulse response of this system?

(b) What is the output of the system $y[n]$, if the input is $x[n] = \delta[n+1] - \delta[n] + \delta[n-1]$?

(c) Show that the DT Fourier transform for $a^n u[n]$ for $|a| < 1$ equals $\frac{1}{1-ae^{-j\Omega}}$. Remember that Ω represents angular frequency. The DT sequence $u[n]$ is called the DT unit step function, and is defined as

$$u[n] = \begin{cases} 1, & \text{if } n \geq 0 \\ 0, & \text{otherwise.} \end{cases}$$

Hint: Recall the geometric series summation formula

(d) **Optional:** Using the result of the previous part, show that if the input to the system $x[n] = \left(\frac{1}{4}\right)^n u[n]$, then the output of the system is $y[n] = 9\left(\frac{1}{4}\right)^n u[n] - 8\delta[n]$. Hints: Apply the DTFT to the difference equation and make substitutions, then do some algebraic manipulations.

This problem is primarily intended to get you comfortable to manipulating discrete-time signals and systems.

Part II

Analog Communication

Chapter 3: Amplitude Modulation (AM)

Did you know that the Eiffel Tower was saved from dismantlement in part by its use as an antenna? [4]

In this chapter, we will discuss Amplitude Modulation (AM), which is a useful method for communicating analog signals. In the early days of broadcast radio, AM was the primary approach used to communicate audio signals. AM also forms an important building block to other approaches for communication, including digital communication as we will see later in this text.

Additionally, we will discuss Frequency Division Multiple Access (FDMA), a multiplexing technique that enables simultaneous transmissions, Single Sideband, a spectrally efficient method of amplitude modulation, and lastly, conventional AM.

3.1 *Electromagnetic Propagation*

Electromagnetic (EM) waves tend to propagate more effectively than sound over large distances in air, which is why they are often used to transmit power and information over large distances. Consider a transmitter emitting EM waves with power, P_Tx, and a receiver at a distance d from the transmitter in free space. The received power P_Rx is given by

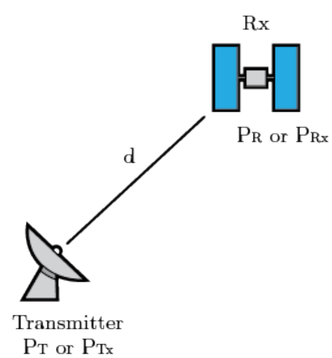


Figure 3.1: Schematic of a transmitter emitting EM waves to a receiver

$$P_{Rx} = \frac{A_{Tx} A_{Rx}}{d^2 \lambda^2} P_{Tx} \tag{3.1}$$

where A_Tx and A_Rx are the apertures of the transmitting and receiving antenna, respectively, and λ is the wavelength of the emitted signal. Aperture is roughly proportional to the physical area of the antenna, which means large wavelength transmitters generally require large antennae.

Audio signals range from 20 Hz to 20 kHz. These signals are fairly low in frequency and therefore, based off the fundamental relationship of frequency, wavelength and speed of light ($f = \frac{c}{\lambda}$), relatively longer in wavelength. Traditionally, low frequency signals like audio required extremely large antenna with high gain to efficiently propagate in free space. Higher frequencies can be received with smaller antennae, but propagation in a real environment is complicated as they are more susceptible to loss or fade. For example, high frequencies in the K_a-band (26.5–40 GHz) are impacted by a propagation loss due to rain called **rain fade**. As the signal frequency increases, the wavelength approaches the size of a rain drop and depolarization occurs, making that particular frequency unsuitable for longer-range propagation without impacting service availability or requiring adaptive coding and modulation (ACM).

The physics of EM propagation introduce a trade-off between antenna size and efficiency. In other words, low frequencies require large antennae, which are not always practical. AM radio represents a solution to this problem so large antennae are not necessarily required to communicate low frequency signals.

3.2 *Amplitude Modulation*

With AM, an information signal, $m(t)$, an unmodulated electrical signal, is multiplied by, or mixed with, a carrier signal, $c(t)$, to produce an AM signal, $u(t)$. Amplitude and frequency modulation are both considered analog modulation schemes because they transmit a continuous waveform as opposed to digital/discrete data. Figure 3.2 shows how this technique works visually. The frequency of the carrier signal, f_c, is called the **carrier frequency**.

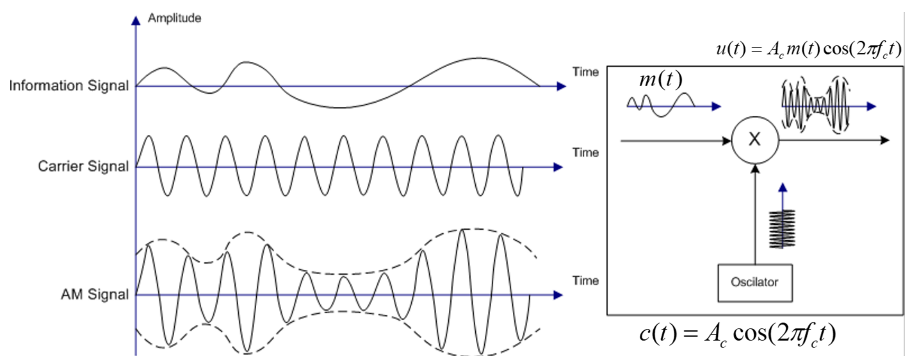


Figure 3.2: Signals used in an AM system (left) along with a block diagram of an AM transmitter (right).

As previously mentioned, the information signal, $m(t)$, occupies low frequencies and is referred to as a baseband signal. When this signal is mixed with our modulated carrier frequency via a high frequency oscillator, the time domain output of the modulator translates our baseband signal to a higher frequency passband, and produces our AM signal, $u(t)$. Tracing the boundary of the modulated signal produces the signal's **envelope**.

AM in the Frequency Domain

To better understand AM systems in the Frequency Domain, sketch a representative "complex" shape $M(f)$ to symbolize the information signal since it can be practically anything. Along with the shape, specify the maximum frequency, f_M, and the height, B, of this "blob". The carrier wave, or our modulating signal, is conveniently defined as a pure cosine wave, which is a pair of impulses $C(f)$ at $\pm f_c$ each with area equal to half the amplitude $\frac{A_c}{2}$. If you forgot why this is true, think back to the mathematically beautiful Euler equation!

The information signal and carrier wave are convolved in the frequency domain (multiplied in time domain) shifting the signal to the impulse locations, placing the representative blob shape at $\pm f_c$. This process yields the AM signal, $U(f)$, which is depicted in Figure 3.3. Typically, the maximum frequency in the signal f_M is much less than the frequency of the carrier f_c. This is written as $f_M \ll f_c$.

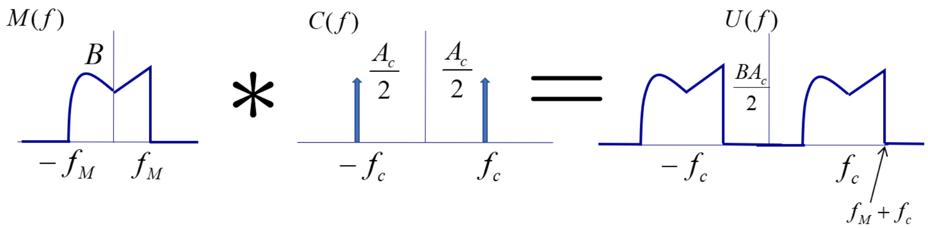


Figure 3.3: A frequency domain view of Amplitude Modulation showing how a low-frequency baseband signal is translated to the higher frequency passband

How would you retrieve an audio information signal that is band-limited to 20 kHz and modulated to 100 MHz? To **demodulate** the received signal back down to the audible range, multiply the received version of $u(t)$, called $r(t)$, by the same carrier wave in the transmitter to demodulate and retrieve the original audio signal.

After the transmitted AM signal, $u(t)$, has propagated over the air, it may experience some loss or attenuation. We will call the resulting signal at the receiver, $r(t)$, but assume for now that no attenuation has occurred. Further, for simplicity, we shall assume that the propagation time is negligible and that there is no noise in the system.

$$r(t) = A_c m(t) \cos(2\pi f_c t) \tag{3.2}$$

Multiplying $r(t)$ by a cosine, $d(t)$, like the carrier cosine demodulates our signal producing $r_d(t)$. In hardware, as shown in Figure 3.4, this multiplication occurs with a mixer, and an oscillator at the same frequency as the transmitter.

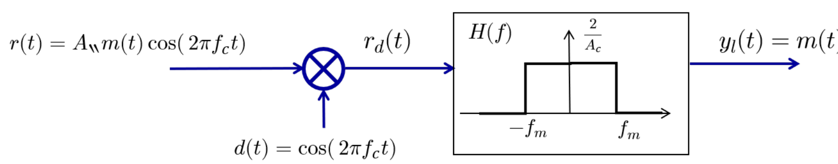


Figure 3.4: A block diagram of AM demodulation.

In the frequency domain, $R_d(f)$ is the convolution of $R(f)$ and $D(f)$, placing a copies of $R(f)$ at the two impulses $\pm f_c$. Since we have convolved a second time, this turns the two blobs of $R(f)$ into three blobs: one at $-2f_c$, one at zero, and one at $2f_c$. We now have our baseband signal at zero with new height $\frac{BA_c}{2}$.

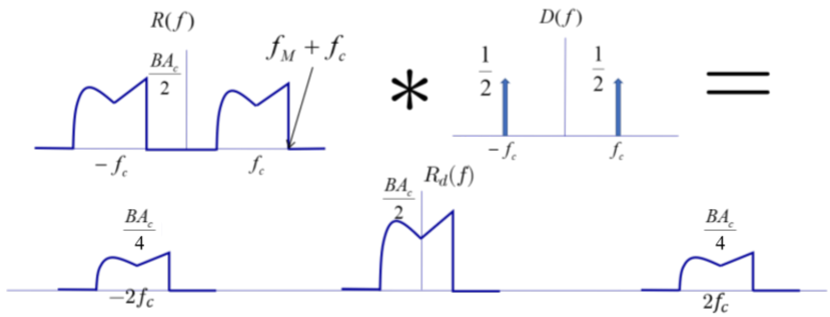
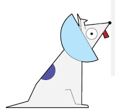


Figure 3.5: Frequency Domain Representation of AM Demodulation

From this figure, we can see how a high-frequency AM signal is used to recover the original baseband / low frequency information signal. Multiplying by a cosine at the same frequency and phase as the carrier and filtering out high frequency components completely recovers the original signal.

Fun Facts from Doggo



You may be asking yourself why the blob at $f = 0$ has twice the amplitude of the blobs at $\pm 2f_c$. Consider the modulated signal with blobs at $\pm f_c$ convolved with $\frac{1}{2}$ area impulses at $\pm f_c$: each impulse places a copy of the two blobs with the midpoint between the impulses at f_c. For the positive impulse, one blob goes to $f_c + f_c = 2f_c$ and the other goes to $f_c - f_c = 0$. For the negative impulse, they go instead to $-2f_c$ and 0. Two blobs are placed at 0, so the result is one blob with twice the height of the others.

Using a low pass filter with gain $\frac{2}{A_c}$, we remove the two extra blobs and scale the baseband part back to its original height. This recovers the original information signal, $m(t)$, which is equal to our demodulated, filtered signal $y_l(t)$.

AM with pilot tone

We have seen how AM demodulation works when the carrier signal frequencies of our transmitter and receiver are identical, but what happens if they contain a phase offset? In other words, what happens if our transmit and receive oscillators have a phase offset, ϕ, which is inevitable in the real world, since oscillators do not typically have a common timing reference. What is the output of the demodulation $y_l(t)$ if

$$d(t) = \cos(2\pi f_c t + \phi) \tag{3.3}$$

and everything else is the same?

To answer this, we start by writing out $r_d(t)$

$$r_d(t) = A_c m(t) \cos(2\pi f_c t) \cos(2\pi f_c t + \phi) \tag{3.4}$$

Using the trigonometric identity

$$\cos\alpha\cos\beta = \frac{1}{2}\cos(\beta-\alpha) + \frac{1}{2}\cos(\beta+\alpha) \tag{3.5}$$

We rewrite $r_d(t)$ using $\alpha = 2\pi f_c t$ and $\beta = 2\pi f_c t + \phi$

$$r_d(t) = A_c m(t)\left(\frac{1}{2}\cos(2\pi f_c t + \phi - 2\pi f_c t) + \frac{1}{2}\cos(2\pi f_c t + \phi + 2\pi f_c t)\right) \tag{3.6}$$

$$= \frac{A_c}{2}m(t)\cos\phi + \frac{A_c}{2}m(t)\cos(4\pi f_c t + \phi) \tag{3.7}$$

When this is run through an ideal low pass filter, the signals above f_m are removed. Given that $f_c >> f_m$, the second term cosine (located at $2f_c$) will be removed leaving the demodulated and filtered output $y_l(t)$.

$$y_l(t) = m(t)\cos\phi \tag{3.8}$$

Since the $\cos\phi$ factor is between 1 and -1, a phase offset attenuates or weakens the signal when $\phi \neq 2\pi k$ for integers k. In fact, when the signals are 90 degrees out of phase, zero power is received! One way to combat this problem is to transmit the carrier wave $c(t)$ for receivers to use for synchronization. Instead of multiplying by their own cosine $d(t)$, which has a nonzero phase offset ϕ, the receiver uses the received wave to synchronize its demodulation and maximize received power. The received carrier wave is called a **pilot tone**. The spectrum of the transmitted signal with a pilot tone is depicted in Figure 3.6.

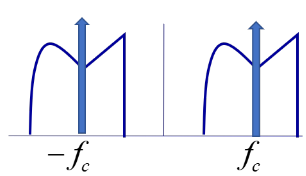


Figure 3.6: A frequency domain view of an AM signal (blobs) and pilot tone (impulses) used for carrier synchronization.

There is a small catch, however, the pilot tone consumes some fraction of the available transmit power, which we control with the pilot amplitude A_p as shown in Figure 3.7.

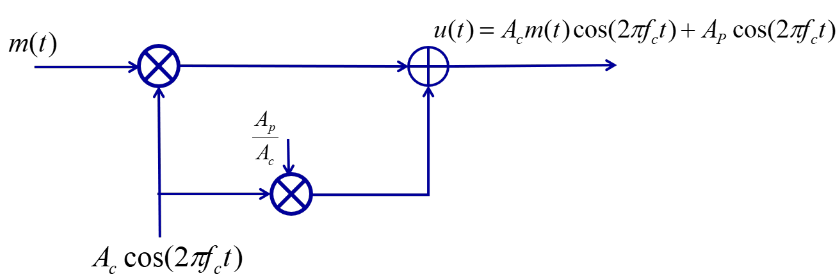


Figure 3.7: A block diagram for an AM transmitter with a pilot tone of amplitude A_p.

At the receiver shown in Figure 3.8, a narrow bandpass filter extracts the pilot tone for demodulation, which is sent through a low pass filter. After filtering, the DC offset of $\frac{A_p}{A_c}$ that has been introduced by the presence of the pilot tone is removed.

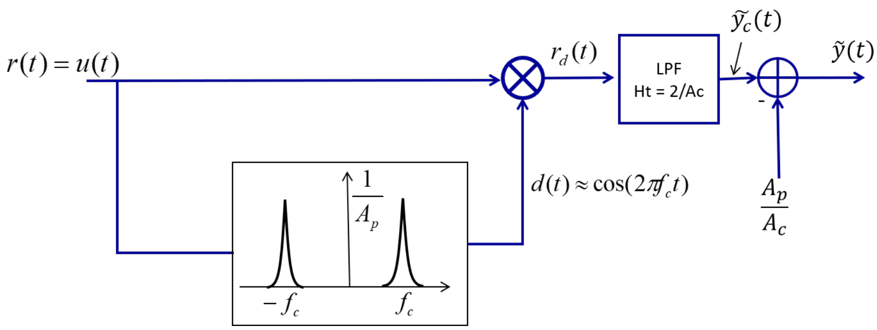


Figure 3.8: A block diagram for an AM receiver using a pilot tone. $\tilde{y}_c(t)$ has a constant offset (called a DC component or bias) because the pilot tone was still in the signal during demodulation.

At this point, the original signal has been recovered and any phase offset between transmitter and receiver does not have a significant impact! In real world systems, the extracted pilot signal $d(t)$ is often fed into another system called a phase-locked-loop (PLL) which generates a clean cosine with the same phase and frequency of the approximate cosine in $d(t)$.

3.3 Amplitude Modulation (AM) and Frequency Division Multiple Access (FDMA)

Have you ever noticed the frequency of an AM radio station and wondered how multiple AM stations share the same air waves? The answer is called Frequency Division Multiple Access (FDMA). In this multiplexing scheme, multiple users divide the band into different frequencies or channels. An example of this this in our every day life is a musical choir. When the altos, sopranos, and tenors come together and sing at the same time, they are effectively transmitting information at different frequencies.

With FDMA for AM radio, multiple stations can transmit as long as the propagated signals do not overlap in the frequency domain and interfere. In practice, stations must center their station frequency in a way such that there is a gap or buffer between adjacent stations, and so that there is no overlap. For example, consider two stations that transmit a message represented by $M_1(f)$ and $M_2(f)$, respectively. Figure 3.9 shows different scenarios that occur depending on how the two stations choose to locate their center frequencies f_1 and f_2 relative to each other.

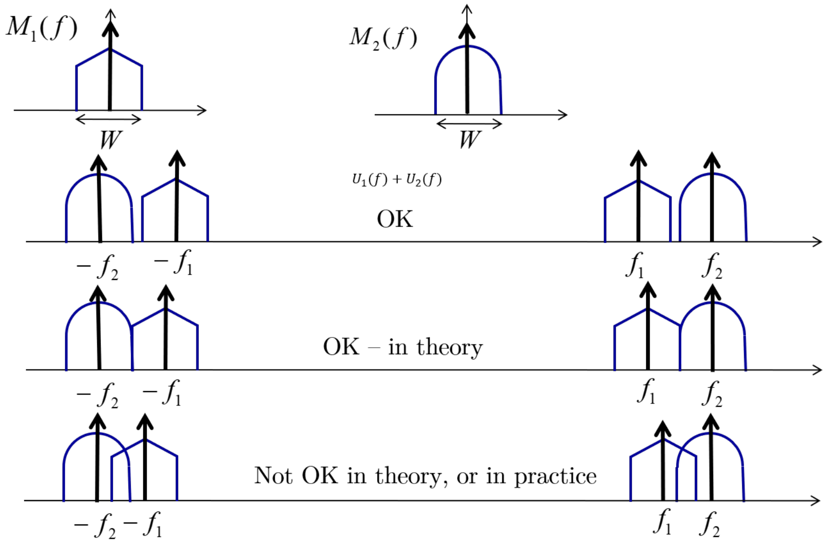


Figure 3.9: When two stations transmit separate message signals $M_1(f)$ and $M_2(f)$, they can both do so simultaneously if there is no overlap in the transmitted signals in the frequency domain. In practice, a buffer between stations is also needed.

In theory, the tightest packing possible for these two stations is such that there is exactly W Hz of gap between f_1 and f_2, or

$$f_2 - f_1 = W \tag{3.9}$$

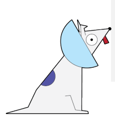
However, in practice, there should be some gap in addition to the theoretically required W. A more realistic requirement for f_1 and f_2 (assuming that $f_2 > f_1$) would then be

$$f_2 - f_1 > W + W_{guard} \tag{3.10}$$

where W_{guard} is the gap between channels to guard against interference from adjacent channels due to non-ideal hardware. As such, W_{guard} is called the *guard band*.

When two stations transmit simultaneously, their transmitted signals add together and are observed at the receiver. As a result, AM receivers also include a tuneable band pass filter to select the desired frequency band and avoid amplifying neighboring unwanted bands.

Fun Facts from Doggo



The FM radio band spans from 88 MHz to 108 MHz. The AM radio band ranges from 535 kHz to 1605 kHz. [5]

3.4 Single Sideband (SSB)

A sideband is the portion of a transmitted signal that resides above or below the carrier frequency f_c. The portion above f_c is called the upper sideband and the portion below f_c is called the lower sideband. The upper and lower sidebands can both be used, as we have seen so far with our "blob". This is called *dual sideband*, which is spectrally wasteful. If only the upper or lower sideband is used, then it is called *single sideband*.

Considering basic AM (no pilot, no FDMA), it can be shown that dual sideband AM is spectrally wasteful. Observe that each sideband independently contains both the pointed and rounded parts of a transmitted signal, as shown in Figure 3.10. This is redundant and wasteful in the frequency domain because all information about the signal is contained in each upper and lower sideband. Using either a low pass filter or a high pass filter, only the lower or upper sidebands, respectively, can be used. This makes the required channel narrower, allowing additional stations to transmit.

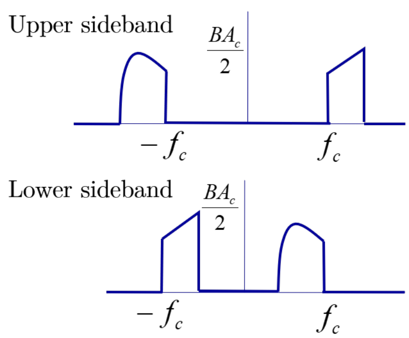


Figure 3.10: The upper and lower sidebands. Added together, these form dual sideband, or regular AM transmission.

If you are not convinced, draw out the frequency domain of the lower side band receiver, $R_l(f)$, as in Figure 3.11 Not only does SSB work, but the configuration of the AM receiver already supports it. The only change required is a bandpass filter to remove neighboring stations.

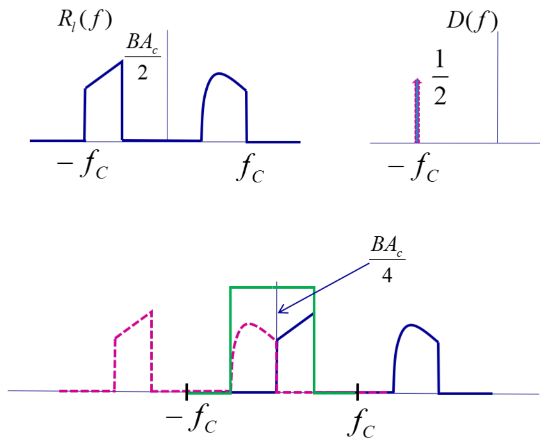


Figure 3.11: Frequency Domain Representation for a Lower Sideband Receiver (R_l)

Also consider sketching the upper side band receiver, $R_U(f)$, as given in Figure 3.12.

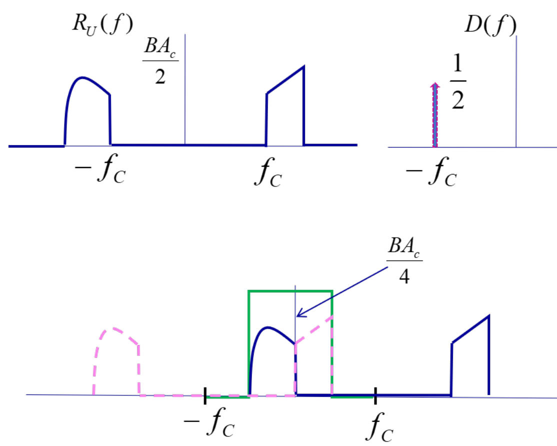
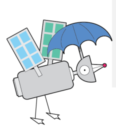


Figure 3.12: Frequency Domain Representation for an Upper Sideband Receiver (R_U)

Exercise with Sat



Lets have another go at it! I want to exchange messages with Doggo. He has informed me that he will be transmitting at F1 with a bandwidth of M. I am going to transmit my message with a bandwidth of 2M. Give me a frequency and system that will allow me to retrieve Doggo's message while still allowing me to transmit my own at the same time. I am a visual learner, so draw it out!

3.5 *Conventional AM*

Most AM radio stations actually use a different system from those covered so far. The transmitted (Tx) signal, $u(t)$, uses a normalized, $m_n(t)$, and offset version of the information signal,

$$u(t) = A_c\left[1 + m_n(t)\right]\cos\left(2\pi f_c t\right) \tag{3.11}$$

where $m_n(t) = \frac{m(t)}{\max|m(t)|}$. This way, $|m_n(t)|$ is ensured to never be greater than 1. Due to normalization of $m_n(t)$, $1 + m_n(t) > 0$ and the envelope of the modulated signal lies above zero, as shown in Figure 3.13.

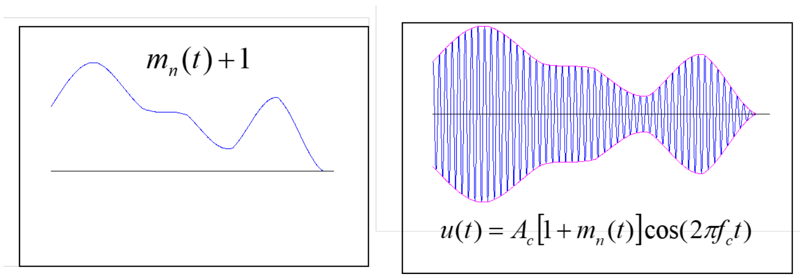


Figure 3.13: The normalized and offset message signal for conventional AM (left). The transmitted signal for conventional AM. This forms an envelope curve for the normalized message signal (right).

When the received signal is rectified (negative components removed, set to zero), the result is the positive parts of $u(t)$. By applying a low pass filter to "fill in the gaps", the original message $m(t)$ can be recovered from the resulting envelope.

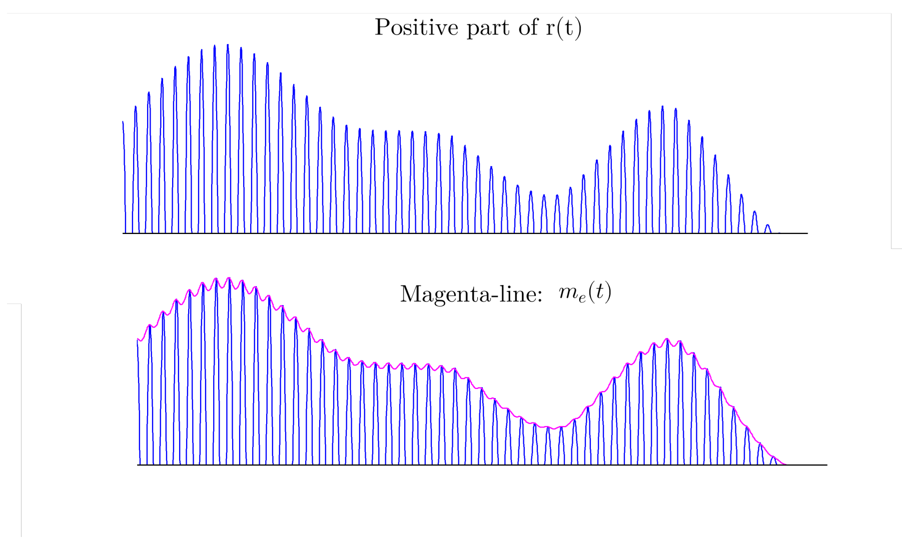


Figure 3.14: The Received Signal for Conventional AM After Rectification (top) and the Received Signal for Conventional AM After Rectification and Low pass Filtering has occurred (bottom). The rectifier-LPF pair forms an envelope detector, which estimates the original message $m_n(t) + 1$.

Rectification and low pass filtering can both be accomplished with a diode and an RC circuit, shown in Figure 3.15. This greatly simplifies the receiving circuit and reduces the overall cost! As long as RC is chosen such that $\frac{1}{f_c} \ll RC \ll \frac{1}{f_M}$, the estimated message $m_e(t)$ will smooth out the carrier wave to track changes in the signal.

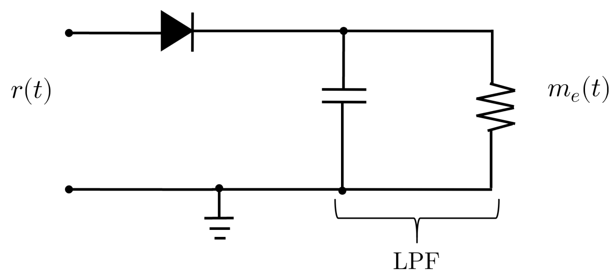


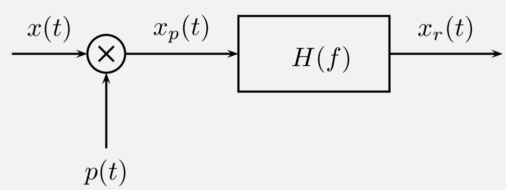
Figure 3.15: A simple diode and RC circuit conventional AM receiver, also called an envelope detector.

3.6 *Problem Set 2: Amplitude Modulation*

Problem	Topic	Points
1	Sampling and Low-pass Filters	1
2	Implementing the Notch Filter in MATLAB	4
3	AM Demodulation in MATLAB	4
4	Real and Imaginary Parts of Fourier Transforms	3
Total:		12

Exercise 3.1

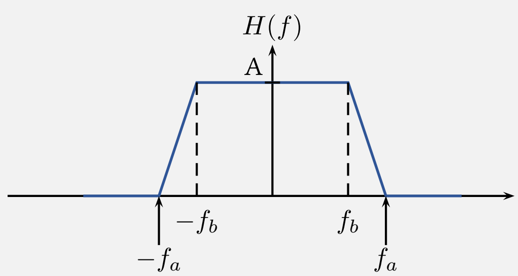
(1 points) Consider the following system which samples and reconstructs a continuous-time signal $x(t)$.

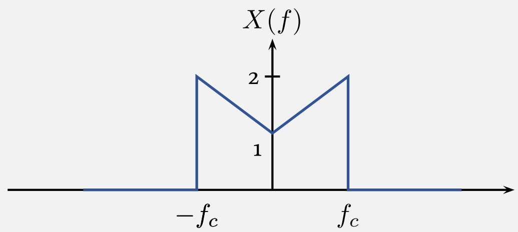


Let the signal $p(t)$ be an ideal impulse train with the unit impulses separated by a period T, i.e.,

$$p(t) = \sum_{k=-\infty}^{\infty} \delta(t - kT)$$

Suppose that $H(f)$ and $X(f)$ are





What is the largest value of T such that with an appropriate A, $x_r(t) = x(t)$? Assume that $f_b \geq f_c$. Note that the $H(f)$ given above is still an idealized model for a low-pass filter, but it is much closer to a real low-pass filter compared to a box in the frequency domain. This problem is intended to help you become familiar with the fundamentals of sampling.

Exercise 3.2

This Exercise provides review on difference equations of a DT signal, which was discussed in Chapter 2.

(4 points) The difference equation of a DT notch filter which removes a tone at the angular frequency ω_0 from a signal $x[n]$ is

$$y[n] - 2q\cos\omega_0 y[n-1] + q^2 y[n-2] = x[n] - 2\cos\omega_0 x[n-1] + x[n-2]$$

where $|q| < 1$ and $q \approx 1$.

In this problem, you will clean up an audio signal corrupted by tones using the notch filter. Start MATLAB and load the file `ps2p2.mat`. The file contains 4 vectors `x`, `x_a`, `x_b` and `x_c`. `x` contains the original audio signal, `x_a` contains the signal corrupted by a single tone at 600 Hz, `x_b` contains the signal corrupted by a tone at an unknown frequency, and `x_c` contains the signal corrupted by **two** tones at unknown frequencies. Please show all your code and calculations.

a) Remove the corrupting tone from `x_a`. This tone is at 600 Hz.

b) Remove the corrupting tone from `x_b`. This tone is at an unknown frequency that you should determine.

c) Remove the two corrupting tones from `x_c`. These tones are at unknown frequencies that you should determine.

d) The original audio signal in `x` itself contains a tonal component, which is part of the background score. Your job is to filter that out.

You may (or may not) find the following MATLAB functions helpful: `sound`, `fft`, `fftshift`, `abs`, `max`, `plot`, `length`, `filter`, `freqz`, `doc`, `help`. These functions have equivalents in Python. Remember that MATLAB indexes arrays starting with 1, and Python starts with 0.

For your convenience, wave files with the audio signals are also provided in case you wish to use Python, or some other software tool. The file names corresponding to `x`, `x_a`, `x_b` and `x_c` are, respectively, x.wav, xa.wav, xb.wav and xc.wav.

Exercise 3.3

(4 points) The relationships between the frequency content of a CT signal and a sampled DT version of that signal (assuming the Nyquist criteria is satisfied) can be used to decode an amplitude modulated (AM) signal. The signal in this problem was synthesized in MATLAB, i.e. it is not a real AM signal. Please load the signal contained in the file TwoAM.wav by running the following in MATLAB.

```
[x, Fs] = audioread('TwoAM.wav');
```

The vector `x` contains samples of a signal $x(t)$ sampled at a frequency of Fs measured in Hz. $x(t)$ was generated as follows:

$$x(t) = m_1(t)\cos(2\pi \cdot 3000t) + m_2(t)\cos(2\pi \cdot 8000t)$$

Both $m_1(t)$ and $m_2(t)$ are audio signals with maximum frequencies of approximately 2000 Hz.

a) By using two different "blobs" to represent $M_1(f)$ and $M_2(f)$, sketch $X(f)$.

b) Use the function `plot_FT.m` in the provided MATLAB file of the same name to plot the Fourier transform of `x`. Note that `plot_FT` takes two arguments. The first is the signal of which to plot the Fourier Transform, and the second is the sample rate. Does the signal appear as expected?

c) What is the impulse response of a DT lowpass filter, which has a cutoff frequency corresponding to 2100 Hz assuming a sample rate of 44100 Hz? Plot the impulse response and frequency response in MATLAB. Note that 100 samples is a sufficient length for the filter.

d) Demodulate the two audio signals ($m_1(t)$ and $m_2(t)$), which are contained (and modulated) in `x`. You will have to think about how to generate the appropriate cosines and lowpass filter to do this in MATLAB. Hint: Consider using a lowpass filter with an impulse response of 100 samples and the functions `conv`, `sinc`, `cos`. Additionally, MATLAB treats all arrays as vectors, so multiplications will be treated as vector multiplications. In order to do element-wise multiplication, you would use `A.*B` instead of `A*B`.

e) Play the two resulting signals using the `sound` function in MATLAB. One is a guitar riff and the other is a violin. Both sound clips are from FreeSound.Org

Exercise 3.4

(3 points) Consider two signals $m_1(t)$ and $m_2(t)$ whose Fourier transforms are given in Figure 3.16.

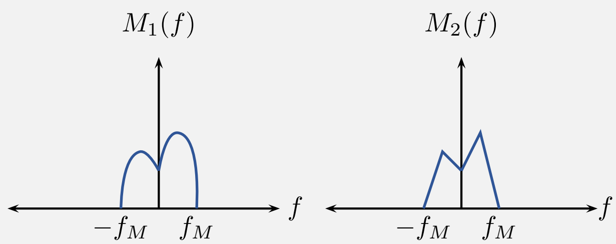


Figure 3.16: $M_1(f)$ and $M_2(f)$

Suppose that $f_m \ll f_c$ and

$$x(t) = m_1(t)\cos(2\pi f_c t) + m_2(t)\sin(2\pi f_c t) \tag{3.12}$$

a) Please sketch the Real and Imaginary parts of $X(f)$ on separate axes

b) Please sketch the Real and Imaginary parts of the Fourier transform of $x(t)\cos(2\pi f_c t)$ on separate axes

c) Please sketch the Real and Imaginary parts of the Fourier transform of $x(t)\sin(2\pi f_c t)$ on separate axes

d) How can you recover $m_1(t)$ from $x(t)$?

e) How can you recover $m_2(t)$ from $x(t)$?

Hint: it is possible to recover both signals from $x(t)$.

Refer back to the figure in Section 2.3 to see what the frequency domain spikes look like for cosine and sine waves, and remember the real and imaginary components.

Chapter 4: Frequency Modulation (FM)

In this chapter, we will discuss Frequency Modulation (FM), which is a form of analog modulation since the baseband signal is typically an analog waveform, although digital information could be encoded in the frequency of a signal as well.

Frequency Modulation is a variation of the more general phase modulation technique that works by changing the phase of a carrier wave. When the relationship between the phase and the modulated signal is linear, the frequency changes, resulting in FM. FM is used in broadcast radio, for baby monitors and other applications. Aside from communications applications, FM signals can also be used for radar, and certain applications involving electroencephalograms (EEGs). A form of FM is also used for digital communications systems that use frequency shift keying (FSK) modulations schemes. These systems encode data by shifting the carrier frequency f_c among a predefined set of frequencies, each frequency representing a particular combination of bits.

4.1 FM Transmitter

In phase modulation systems, The information signal is then embedded in the varying phase of the transmitted signal, $u(t)$, where

$$u(t) = A_c \cos\left(2\pi f_c t + \phi(t)\right) \tag{4.1}$$

and phase, $\phi(t)$, is a function of of the message signal $m(t)$. For FM systems, the relationship between $\phi(t)$ and $m(t)$ is given by

$$\phi(t) = 2\pi k_f \int_{-\infty}^{t} m(\tau) d\tau \tag{4.2}$$

In the above equation, k_f is called the deviation constant. The deviation constant controls how far the carrier signal moves in frequency from the nominal value, and a larger k_f implies a larger fluctuation in frequency. A single phase shift only changes the horizontal offset of the signal, whereas a time-varying phase shifts the frequency. Furthermore, the time derivative of phase, $\phi(t)$, is known as the instantaneous frequency, $f(\tau)$.

The instantaneous carrier frequency is given by

$$f_t(t) = f_c + \frac{1}{2\pi}\frac{d}{dt}\phi(t) \tag{4.3}$$

$$= f_c + k_f m(t) \tag{4.4}$$

which is the sum of the nominal carrier frequency and an offset proportional to $m(t)$.

In order to generate the FM signal, we need an integrator for the phase function and a voltage controlled oscillator (VCO) using, e.g., a crystal oscillator or a voltage controlled capacitor (also called a varactor diode or varicap) as in Figure 4.1.

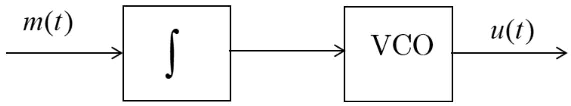


Figure 4.1: FM Transmitter Circuit. The Integrator can be built with an Operational Amplifier (OpAmp) and the Voltage-Controlled Oscillator (VCO) can be built with Transistors, Inductors, and Variable Capacitors.

Example 4.1

To visualize FM, consider the message signal and carrier in Figure 4.2.

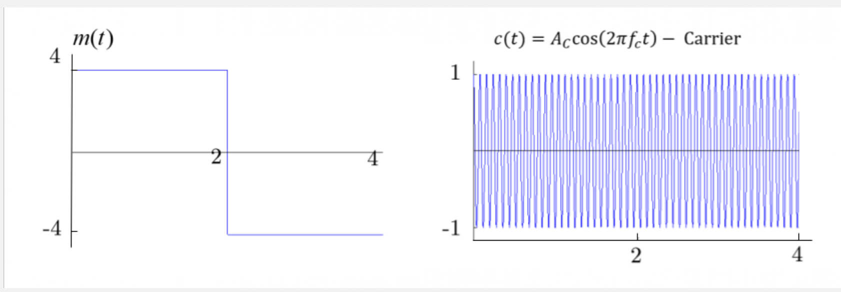


Figure 4.2: An example Message (left) and Carrier Signal (right).

If we let $k_f = 1$ *for simplicity (in practice this* k_f *can equal 1000s), then* $\phi(t)$ *is obtained from the integral of* $m(t)$ *in Figure 4.2 as*

$$\phi(t) = 2\pi k_f \int_{-\infty}^{t} m(\tau) d\tau \tag{4.5}$$

Evaluate $\phi(t)$ *and* $u(t)$.

Solution:

Note that the upper limit in the integral in Equation 4.5 is t, and therefore the equation simplifies to

$$\begin{aligned}
\phi(t) &= 2\pi(1) \int_{-\infty}^{t} m(\tau) d\tau && (4.6)\\
&= 2\pi \int_{0}^{t} 4 d\tau + 2\pi \int_{2}^{t} -4 d\tau && (4.7)\\
&= 2\pi 4\tau \Big|_{0}^{t} + 2\pi(-4)\tau \Big|_{2}^{t} && (4.8)\\
&= \begin{cases} 8\pi t & 0 \le t \le 2 \\ 32\pi - 8\pi t & 2 \le t \le 4 \end{cases} && (4.9)
\end{aligned}$$

From $t = 0$ to $t = 2$, the value of $m(t)$ is 4, so the phase increases with slope 8π. After that, the phase decreases with negative slope 8π back to zero. The resulting phase function and transmitted FM signal, shown in Figure 4.3, contain the message signal. The constantly changing phase changes the frequency between a relatively high frequency before $t = 2$ and a lower one after $t = 2$.

$u(t)$ can be calculated by using what we know from Equation 4.1 and what we found for $\phi(t)$ in Equation 4.9. Therefore, $u(t)$ simplifies to

$$u(t) = \begin{cases} \cos(2\pi f_c t + 8\pi t) \\ \cos(2\pi f_c t + 32\pi - 8\pi t) \end{cases} \tag{4.10}$$

$$= \begin{cases} \cos[2\pi(f_c + 4)t] \\ \cos[2\pi(f_c - 4)t + 32\pi] \end{cases} \tag{4.11}$$

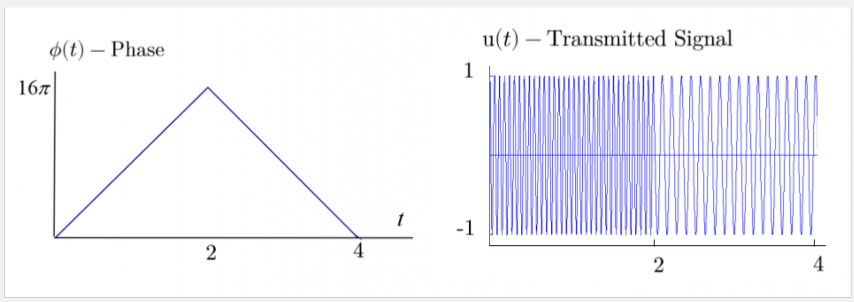


Figure 4.3: An example Phase (left) and u(t): Transmitted FM Signal (right).

4.2 *FM Receiver: Detection*

At the receiver, we will assume an ideal channel for simplicity, meaning that the received signal is exactly what was transmitted. To retrieve the original signal, we first take the time derivative of the FM modulated signal then use an envelope detector to recover the transmitted message $m(t)$ with a DC offset, K, as shown in the block diagram in 4.4.

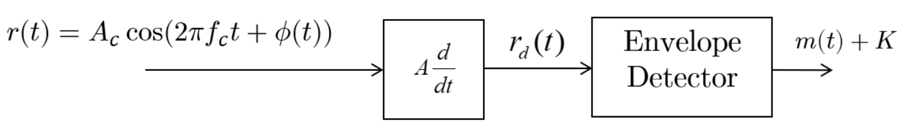


Figure 4.4: FM Receiver Block Diagram

The derivative of the received signal $r(t)$ is $r_d(t)$. With the following steps, we see that $r_d(t)$ is a lot like the conventional AM signal after taking the derivative.

$$
\begin{aligned}
r_d(t) &= \frac{d}{dt} r(t) && (4.12)\\
&= \frac{d}{dt} A_c \cos\left(2\pi f_c t + \phi(t)\right) && (4.13)\\
&= A_c \left(2\pi f_c + 2\pi k_f \frac{d}{dt} \int_{-\infty}^{t} m(\tau) d\tau\right) \cos\left(2\pi f_c t + \phi(t)\right) && (4.14)\\
&= 2\pi A_c \left(f_c + k_f m(t)\right) \cos\left(2\pi f_c t + \phi(t)\right) && (4.15)
\end{aligned}
$$

Observe that $r_d(t)$ is the product of a cosine with a time-varying phase and a scalled, offset version of the message signal. if the relationship between f_c, k_f and the amplitude of the message signal $m(t)$ is such that the term in the first parenthesis in (4.15) is non-negative, $r_d(t)$ is very similar to a conventional AM signal.

By applying an envelope detector, which we discussed in Section 3.5, we can connect the peaks and recover a signal that is proportional to the original message plus a constant offset, commonly referred to as a DC offset. Figures 4.5, 4.6, 4.7, and 4.8 provide an overview of the process from transmission of the message signal to recovery. The original message, $m(t)$, shown in Figure 4.5, is the raw signal in the time domain.

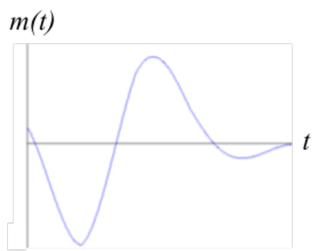


Figure 4.5: Message Signal, $m(t)$

The original signal is modulated to create an FM signal, r(t), shown in Figure 4.6

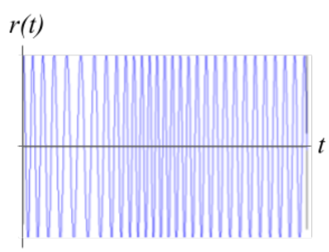


Figure 4.6: Modulated Message Signal, $r(t)$

The signal $r(t)$ is represented by the following equation:

$$r(t) = u(t) = \cos\left(2\pi f_c t + 2\pi k_f \int_{-\infty}^{t} m(\tau) d\tau\right) \tag{4.16}$$

Next, the derivative of the FM signal, $r(t)$, is taken with respect to time, as seen below in Figure 4.7.

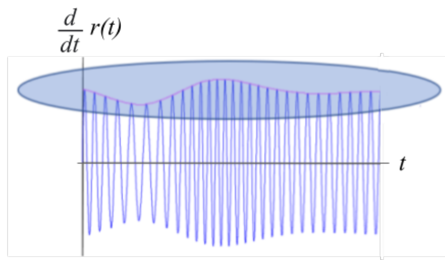


Figure 4.7: Derivative of Modulated Message Signal, $r(t)$

The derivative of $r(t)$ is represented by the following equation:

$$\frac{d}{dt}r(t) = -(2\pi f_c + 2\pi k_f m(t))\sin\left(2\pi f_c t + 2\pi k_f \int_{-\infty}^{t} m(\tau)d\tau\right) \tag{4.17}$$

Lastly, when the FM signal is run through an envelope detector circuit, the outcome becomes proportional to the message signal with a constant offset, as shown in Figure 4.8.

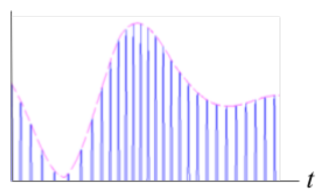


Figure 4.8: Recovered Message

4.3 *FM Effective Bandwidth*

For wireless communications systems which use some form of Frequency Division Multiple Access (FDMA), e.g., broadcast radio, the bandwidth occupied by a transmitter is an important system parameter which helps ensure that stations don't interfere with each other. In AM radio, the bandwidth is well defined. Due to the highly nonlinear nature of FM, the bandwidth occupied by an FM transmitter is not easy to compute, except for a few very simple message signals, $m(t)$. Carson's Rule approximates the bandwidth B_c of a transmitted FM signal.

$$B_c = 2(\beta + 1)W \tag{4.18}$$

where W is the bandwidth of the message signal and β is the **modulation index**:

$$\beta = \frac{k_f \max|m(t)|}{W} \tag{4.19}$$

Greater bandwidth brings higher noise immunity and fidelity for audio signals, so it should be maximized without causing harmful interference to other stations.

Example 4.2

Let $m(t) = \frac{10^{-3}\sin(10^4 \pi t)}{\pi t}$ *and* $k_f = 4000$. *What is the approximate bandwidth of the FM signal? Note that* $F = \frac{2\pi k t}{\pi t}$ *corresponds to a box with height* 1 *from* $-k$ *to* k, *where* $k = W$.

Solution:

We know from Carson's Rule in Equation 4.18 that

$$B_c = 2(\beta + 1)W \tag{4.20}$$

We can use Equation 4.19 to find the value of β

$$\begin{aligned} \beta &= \frac{k_f \max|m(t)|}{W} && (4.21)\\ &= \frac{4000 \max|m(t)|}{10^4/2} && (4.22)\\ &= \frac{8000 \max|m(t)|}{10^4} && (4.23) \end{aligned}$$

Since the numerator and denominator of $m(t)$ are zero at $t = 0$, we can use L'Hopital's Rule

$$\begin{aligned} \lim_{t\to 0} m(t) &= \lim_{t\to 0} \frac{10^{-3}\sin(\pi(10^4)t}{\frac{d}{dt}(\pi t)} && (4.24)\\ &= \frac{10\pi\cos(0)}{\pi} && (4.25)\\ &= 10 && (4.26) \end{aligned}$$

By plugging in the value of 10 for $m(t)$, we can finish solving for β

$$\begin{aligned} \beta &= \frac{8000(10)}{10^4} && (4.27)\\ &= 8 && (4.28) \end{aligned}$$

Since we now know that $\beta = 8$ and $W = \frac{10^4}{2} = 5000$, we can plug these values into Carson's Rule, giving us a B_c value of

$$\begin{aligned} B_c &= 2(\beta + 1)W && (4.29)\\ &= 2(8 + 1)(5000) && (4.30)\\ &= 90000 && (4.31) \end{aligned}$$

Therefore, the approximate bandwidth of the FM signal is 90 kHz.

Part III

DIGITAL COMMUNICATION

Chapter 5: Probability and Random Variables

We now switch gears and consider communication of digital information, i.e., transmission of sequences of 1s and 0s. As the saying goes, "Digital circuits are made from analog parts". In this chapter, we will learn how to transmit digital information in an analog world, but will need one to dive into the mathematical tools of probability.

As we reviewed in Chapter 1 of this text, the goal of digital communication systems is to communicate bits from a source to a destination, typically over analog channels. The performance characteristics of these digital communications systems include:

- Data rate (measured in bits per second (bps)
- Transmit power P_{Tx}
- Bandwidth: the amount of spectrum a channel occupies
- Probability of Error (POE): the number of bit flips over a given period of time
- Noise Power: the concentration of noise in a system

In the remainder of this chapter, we will analyze how these quantities are related to each other.

5.1 Probability

Why do we need to learn about probability?

In all non-trivial communications systems, the receiver can only **guess** the bits that were transmitted by a source as information gets corrupted by noise and other types of distortion. The probability that the destination guesses a transmitted bit incorrectly is known as the probability of bit error. The **probability of error** over time is quantified in the form of a **bit error rate (BER)**. Thus, to understand BER, we must know at least some probability. We also need an understanding of probability to meaningfully work with noise, e.g., how to quantify the noisiness of a system.

The Big Picture

Probability is a mathematical tool to formally analyze non-deterministic events. Loosely speaking, if a particular outcome of an experiment has probability, P, then in a large number of independent trials, that outcome will occur a fraction P of the time. For example, the probability that heads will be the outcome of the flip of a fair coin is $\frac{1}{2}$. In 10,000 independent coin flips, approximately 5,000 will be heads.

To speak (slightly) more formally, the **sample space** is a set whose members include all possible outcomes of an experiment of interest. Loosely speaking, each subset[1] of the sample space has a **probability**, a relative likelihood of an event in that subset being the outcome of the experiment. For example, the outcomes of a coin toss may have two members: heads (H) and tails (T). It may have three outcomes if landing

[1] *Note that more formally, not all subsets of a sample space can be assigned probabilities, but a more formal definition of probability is beyond the scope of this text.*

on the edge (E) is another possibility. If we only consider the scencario in which the coin can either be heads or tails, we can describe the scenario with the following equations:

$$P(H) = P(t) = \frac{1}{2} \tag{5.1}$$

The universal set (Ω) is another term for the sample space and should always occur with 100% probability. Mathematically, this can be defined as:

$$P(\Omega) = 1 \tag{5.2}$$

The universal set of four different sample spaces are provided pictorially in Figure 5.1.

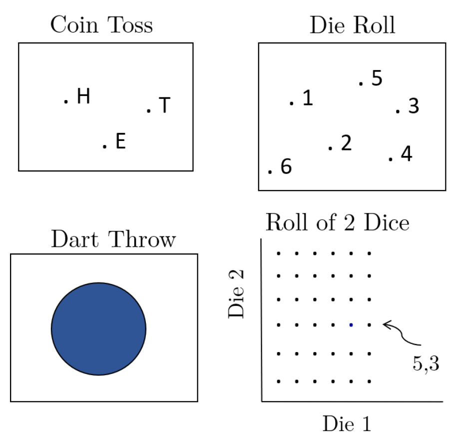


Figure 5.1: The universal set or sample space of a coin toss is heads (H), tails (T), and edge (E). For a single die, it is a number for each of 6 sides. A dart throw may be a coordinate on the board or missing the board entirely. Two dice produce an array of combinations.

If we consider the dart board as a coordinate system of points, then we can compute the probability of given outcomes per throw.

Assuming that a dartboard has radius R, what is the probability that dart hits a circle of radius $d < R$? What is the probability that a dart hits any given point on the board, assuming the dart has an infinitesimally small tip? These are questions we can answer with probability tools.

5.2 Random Variables (RVs)

Random variables (RVs) map events in a sample space to numbers (real or complex). An example of this is mapping the outcomes of die rolls to the random variable x, which equals the number on the face of die. We could also map the coordinates on where the dart hit the board to the variable r, which equals the distance of the dart from the center of the board. In short, when we work with random variables we are connecting an underlying sample space to numbers.

Probability Mass Functions (PMFs) and Probability Density Functions (PDFs)

In communications theory, we primarily work with two types of RVs: discrete and continuous random variables.

Discrete RVs take on discrete values, such as the roll of a die, or a fraction 0.5. Their probabilities are described by a Probability Mass Function (PMF).

$$P\{x = x_i\} = p_i, i = 1, 2, 3, ... \tag{5.3}$$
$$p_i \geq 0 \tag{5.4}$$
$$\sum_i p_i = 1 \tag{5.5}$$

In probability mass functions, the height defines probability of a particular outcome. These are not DT signals/sequences, however, and they can be defined for non-integer values.

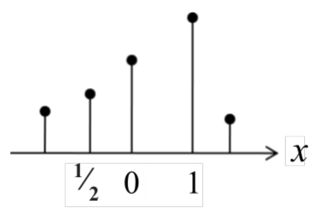


Figure 5.2: Probability Mass Function of $f_X(x)$

Continuous RVs take on a continuum of values, such as the distance of a dart from the center of a dartboard. Their probabilities are described by a Probability Density Function (PDF).

$$P\{X \in (a, b]\} = \int_a^b f_X(x)dx. \tag{5.6}$$
$$f_X(x) \geq 0 \tag{5.7}$$
$$\int_{-\infty}^{\infty} f_X(x)dx = 1 \tag{5.8}$$

A probability density function can determine the likelihood that a random variable is between two numbers, like if the temperature of Boston is between 32-33 degrees. The probability of any particular event is infinitesimally small, however. The probability density function (PDF) is shown below in Figure 5.3.

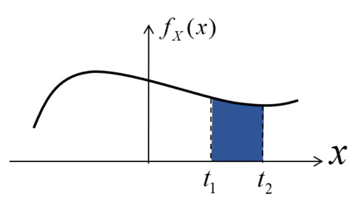


Figure 5.3: Probability Density Function of $f_X(x)$

A PDF is given by the following equation:

$$P\{t_1 < X \leq t_2\} = \int_{t_1}^{t_2} f_X(x)dx \tag{5.9}$$

We will primarily work with two different random distributions. The first is the Uniform Random variable for which each point in $[a, b]$ is equally likely, as shown in Figure 5.4. An example might be the final angle in $[0, 2\pi]$ of a randomly rolled wheel. Its PDF is

$$f_X(x) = \begin{cases} \frac{1}{b-a} & x \in [a, b] \\ 0 & x \notin [a, b] \end{cases} \tag{5.10}$$

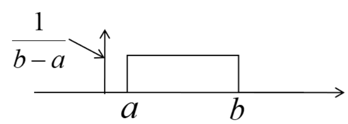


Figure 5.4: The PDF of a uniform random variable. All values between a and b are equally likely.

The second is the Gaussian Random Variable, also called the Normal Distribution. It describes many natural phenomena quite well and has the PDF defined below and depicted in Figure 5.5.

$$f_X(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{(x-\mu)^2}{2\sigma^2}} \tag{5.11}$$

In the above equation, σ is the standard deviation and μ is the mean. It also has a familiar bell-shaped curve. There are many other random variables defined by their own respective distributions, but these two are the most common for our work in communication systems.

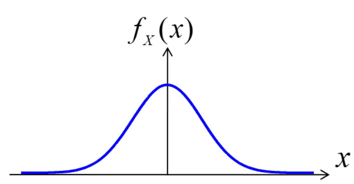


Figure 5.5: The PDF for a Gaussian Random Variable

Cumulative Distribution Function

Another useful function is the Cumulative Distribution Function (CDF). It describes the probability that a random variable is less than or equal to a given value $P\{X \leq x\}$, and is denoted as $F_X(x)$. It is defined for continuous random variables in terms of the PDF, $f_X(x)$.

$$P\{X \leq x\} = F_X(x) \tag{5.12}$$

$$= \int_{-\infty}^{x} f_X(x)dx \tag{5.13}$$

$$f_X(x) = \frac{d}{dx} F_X(x) \tag{5.14}$$

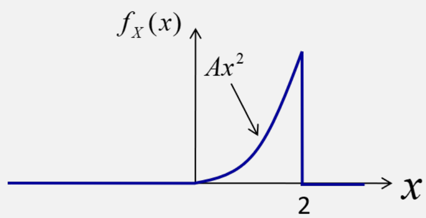
For discrete random variables, the formulation is similar

$$P\{X \leq x\} = F_x(x) = \sum_{x_i \leq x} P\{x = x_i\} \tag{5.15}$$

The discrete CDF is continuous unlike the PMF because it can be evaluated for any value of x.

Example 5.1

Suppose we have a random variable X with the PDF $f_X(x)$ depicted below for some value of A



- *What is the CDF?*
- *What is the probability that $x \leq 1$?*
- *What is the probability that $x > y$?*

Solution:

The CDF can be evaluated by plugging the PDF into the integral

$$F_X(x) = \int_0^x Ax^2 dx \tag{5.16}$$

$$= \frac{A}{3}x^3\Big|_0^x \tag{5.17}$$

$$= \frac{A}{3}x^3 \tag{5.18}$$

The probability that $x \leq 1$ is easily evaluated

$$F_X(1) = \frac{A}{3}(1^3)$$

$$= \frac{A}{3} \tag{5.19}$$

The probability that $x > y$ is the probability that x is not less than or equal to y, or $P\{x \leq y\}$.

$$P\{x > y\} = 1 - P\{x \leq y\} \tag{5.20}$$

$$= 1 - F_X(y) \tag{5.21}$$

$$= 1 - \frac{A}{3}y^3 \tag{5.22}$$

Additionally, you might note that the probability $x > y$ when $y < 0 = 1$. The probability that $x > y$ when $y > 2 = 0$.

Joint Densities of RVs

In communications theory, it is common to work with multiple random variables simultaneously. Given two random variables X and Y, the probability that $X = x$ AND $Y = y$ is called the joint PMF and is denoted $P\{X = x, Y = y\}$. As an example, consider the roll of two die, where X and Y are the numbers on the face of each dice. We would write the probability that both die roll a 6 as $P(X = 6, Y = 6)$.

For continuous random variables, $f_{X,Y}(x, y)$, as defined in the equation below, is used to denote the joint PDF along with the question if the point (X, Y) is in some region A.

$$P\{(x, y) \in A\} = \iint_A f_{X,Y}(x, y)dxdy \tag{5.23}$$

The joint PDF also satisfies the following equation, which states that the probability that (X, Y) lies in the sample space is 100%.

$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f_{X,Y}(x, y)dxdy = 1 \tag{5.24}$$

Independence and Conditional Probability for RVs

Statistical independence describes events whose outcomes do not depend on other events. Strictly speaking, two random variables, X and Y are said to be independent if and only if the joint density is the product of the individual densities.

$$f_{X,Y}(x, y) = f_X(x)f_Y(y) \qquad \text{(continuous)} \tag{5.25}$$

$$P\{X = x, Y = y\} = P\{X = x\}P\{Y = y\} \qquad \text{(discrete)} \tag{5.26}$$

If two random variables are dependent, we can use Bayes' Rule for Random Variables to calculate the PDF for X given a condition on Y for $f_Y(y) > 0$.

$$f_{X|Y}(x|y) = \frac{f_{X,Y}(x, y)}{f_Y(y)} \tag{5.27}$$

Applying Bayes' rule to two independent variables results in the PDF of one of them, e.g., by applying Equation 5.25

$$f_{X|Y}(x|y) = \frac{f_{X,Y}(x, y)}{f_Y(y)} \tag{5.28}$$

$$= \frac{f_X(x)f_Y(y)}{f_Y(y)} \tag{5.29}$$

$$= f_X(x) \tag{5.30}$$

Marginal Probability Functions from Joint Probability Functions

When two random variables X and Y are dependent, we may still wish to know the likelihood of some event for X regardless of the outcome for Y. In that case, we are looking for the **marginal PDF** (continuous RVs) or the **marginal PMF** (discrete RVs).

To find the marginal PDF for continuous RVs, integrate over the sample space for other random variables, e.g.,

$$f_X(x) = \int_{-\infty}^{\infty} f_{X,Y}(x, y)dy \tag{5.31}$$

With discrete RVs, the equivalent is the marginal PMF. To find the marginal PMF, sum over the other random variables instead.

$$P\{X = x\} = \sum_j P\{X = x, Y = y_j\} \tag{5.32}$$

We can also find the marginal PDF for a mixture of discrete and continuous RVs by summing continuous PDFs together over all of the other variables' possible discrete outcomes.

$$f_X(x) = \sum_j f_{X,Y}(x, y = y_j) \tag{5.33}$$

Loosely speaking, these marginal probability equations result from counting/summing all possible scenarios for Y where X has a certain desired outcome.

Mean and Variance of RVs

The expected value of the random variable, X is also referred to as the weighted average and is defined as

$$\mu_X = E[X] = \int x f_X(x)dx \tag{5.34}$$

$$= \sum_j^k x_i p_i \tag{5.35}$$

$$= x_1 p_1 + x_2 p_2 + ... \tag{5.36}$$

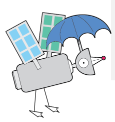
Exercise with Sat

Let's see this in action! Calculate the expected value of rolling a six sided die. There are six sides to the die with equal probability of occurring. What is the weighted average

Solution:

The weighted average of the die can be calculated using Equation 5.36.

$$\begin{aligned}\mu_X &= x_1 p_1 + x_2 p_2 + \ldots\\ &= 1(\frac{1}{6}) + 2(\frac{1}{6}) + \ldots\\ &= 3.5\end{aligned}$$



The weighted average of a six-sided die is 3.5.

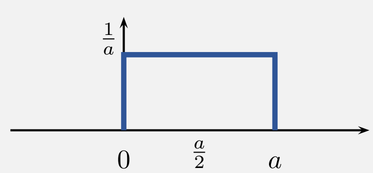
The expected value is a linear operator, so for constants a, b, c the expected value is a linear combination.

$$E[aX + bY + c] = aE[X] + bE[Y] + c \tag{5.37}$$

When the weights, or constants, are equal, the weighted average, or expected value, is equal to the arithmetic mean.

Example 5.2

Given the following PDF, calculate the expected value.



Solution:

The expected value, $E[x]$, can be calculated as follows:

$$
\begin{aligned}
E[x] &= \int_a^0 x\frac{1}{a}dx \\
&= \frac{1}{a}\frac{x^2}{2} \\
&= \frac{a}{2}
\end{aligned}
$$

The variance, σ^2, of a RV is defined as

$$
\begin{aligned}
\sigma_X^2 &= E[(X - E[X])^2] && (5.38)\\
&= E[X^2 - 2X \cdot E[X] + [E[X]]^2] && (5.39)\\
&= E[X^2] - E[2X \cdot E[X]] + [E[X]]^2 && (5.40)\\
&= E[X^2] - 2[E[X] \cdot E[X]] + [E[X]]^2 && (5.41)\\
&= E[X^2] - 2[E[X]]^2 + [E[X]]^2 && (5.42)\\
&= E[X^2] - [E[X]]^2 && (5.43)
\end{aligned}
$$

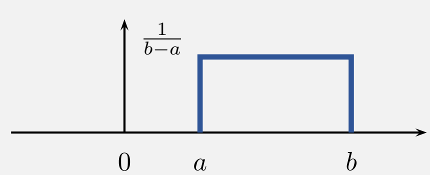
Two random variables are uncorrelated if

$$E[XY] = E[X]E[Y] \tag{5.44}$$

In words, this states that the expected value of a product of two uncorrelated variables is the same as the product of their expected values. When the variables are correlated their average product may be different than the average of their products. It's important to note that Independence implies uncorrelatedness, but not necessarily the other way around.

Example 5.3

Calculate the expected value and variance of the generalized form of the uniform distribution.



Solution:

The expected value, $E[x]$, can be calculated as follows:

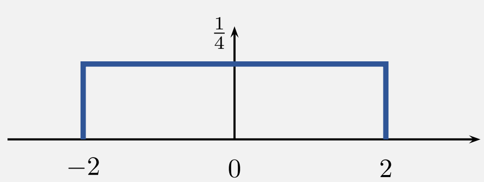
$$\begin{aligned} E[x] &= \int_a^b \frac{x}{b-a} dx \\ &= \left. \frac{x^2}{2(b-a)} \right|_a^b \\ &= \frac{a+b}{2} \end{aligned}$$

The variance, σ^2, is as follows:

$$\sigma_X^2 = \frac{1}{12}(b-a)^2$$

Example 5.4

Calculate the expected value and variance of the following uniform distribution.



Solution:

The expected value, $E[x]$, can be calculated as follows:

$$\begin{aligned}
E[x] &= \int_a^b \frac{x}{b-a} dx \\
&= \left.\frac{x^2}{2(b-a)}\right|_a^b \\
&= \frac{a+b}{2} \\
&= \frac{-2+2}{2} \\
&= 0
\end{aligned}$$

The variance, σ^2, can be calculated as follows:

$$\begin{aligned}
\sigma_X^2 &= E[X^2] - (E[X])^2 \\
&= E[X^2] \\
&= \int_{-\infty}^{\infty} x^2 f_x(x) dx \\
&= \int_{-2}^{2} x^2 \left(\frac{1}{4}\right) dx \\
&= \frac{x^3}{3} \left(\frac{1}{4}\right) dx \\
&= \left.\frac{x^3}{12}\right|_{-2}^{2} \\
&= \frac{2^3}{12} - \frac{(-2)^3}{12} \\
&= \frac{4}{3}
\end{aligned}$$

5.3 Functions of RVs

Suppose that a random variable, Y, is a function of another random variable, X.

$$Y = g(X) \tag{5.45}$$

The mean of Y would be

$$E[Y] = \int_\Omega g(x) f_X(x) dx \tag{5.46}$$

Example 5.5

Suppose a constant but random voltage, v, is applied across a resistor with resistance, R. To find the average power dissipated over the possible voltages, write the random variable for power, P, as a function of the random voltage.

$$P = g(V) = \frac{V^2}{R}$$

The average power dissipated would be

$$E[P] = \int \frac{v^2}{R} f_V(v) dv$$

If the voltage is equally likely to be anywhere in $[-5, 5]$, then the PDF is a uniform distribution with the following function

$$f_V(v) = \begin{cases} \frac{1}{10} & -5 \leq v \leq 5 \\ 0 & \text{otherwise} \end{cases}$$

The integral can then be evaluated to a closed form solution for the average power.

$$\begin{aligned} E[P] &= \int_{-5}^{5} \frac{v^2}{R} \frac{1}{10} dv \\ &= \left. \frac{v^3}{30R} \right|_{-5}^{5} \\ &= \frac{1}{30R}(125 - (-125)) \\ &= \frac{25}{3R} \end{aligned}$$

Now, let us look at sums of random variables.
Let W be the sum of two random variables

$$W = X + Y \tag{5.47}$$

If X and Y are independent, then the PDF of W is the convolution

$$f_W(w) = (f_X * f_Y)(w) \tag{5.48}$$

If X and Y are uncorrelated, then their variances add together

$$\sigma_W^2 = \sigma_X^2 + \sigma_Y^2 \tag{5.49}$$

5.4 Gaussian RVs

Loosely speaking, the sum of many independent random variables will usually be Gaussian distributed. In communications systems, this is important because the PDF of noise is typically Gaussian. If X is Gaussian distributed with mean, μ, and variance, σ^2, then

$$X \sim \mathcal{N}(\mu, \sigma^2) \tag{5.50}$$

If W is a new RV defined by the affine function

$$W = AX + b \tag{5.51}$$

then W is the Gaussian RV

$$W \sim \mathcal{N}(A\mu + b, A^2\sigma^2) \tag{5.52}$$

This is an interesting result; if we multiply a Gaussian RV and/or add a constant offset, the result is also Gaussian! In fact, **all linear combinations of independent Gaussian RVs are also Gaussian RVs**, e.g., for independent $X \sim \mathcal{N}(\mu_X, \sigma_X^2)$ and $Y \sim \mathcal{N}(\mu_Y, \sigma_Y^2)$

$$V \sim \mathcal{N}(\mu_X + \mu_Y, \sigma_X^2 + \sigma_Y^2) \tag{5.53}$$

Thus, if we understand a particular Gaussian RV well, then we can evaluate others with arithmetic operations. Suppose we have a Gaussian RV with $\sigma^2 = 1$ and $\mu = 0$. This is called the **Standard Gaussian Random Variable** and has the PDF

$$f_X(x) = \frac{1}{\sqrt{2\pi}} e^{\frac{-x^2}{2}} \tag{5.54}$$

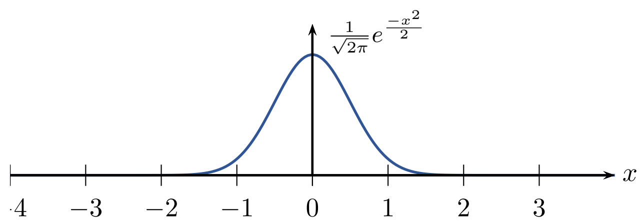


Figure 5.6: The PDF of a standard Gaussian RV.

5.5 *Q-function*

We now introduce one of the most important functions for digital communications, the **Q-function**. The Q-function is the probability that a Standard Gaussian RV exceeds a given value t

$$Q(t) = \int_t^{\infty} \frac{1}{\sqrt{2\pi}} e^{\frac{-\tau^2}{2}} d\tau = P\{x > t\} \tag{5.55}$$

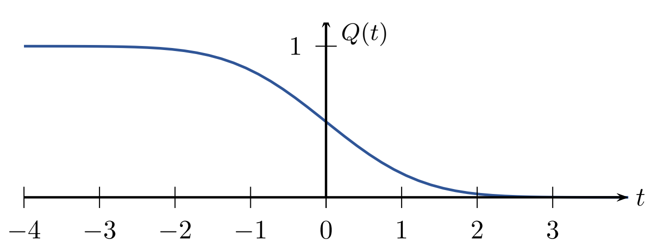


Figure 5.7: The Q-function.

A few values of the Q-function are provided in Table 5.6.

x	$Q(x)$	x	$Q(x)$
$\frac{1}{4}$	0.401	1	0.159
$\frac{1}{3}$	0.369	$\frac{5}{4}$	0.106
$\frac{1}{2}$	0.309	$\frac{4}{3}$	0.0912
$\frac{2}{3}$	0.252	$\frac{3}{2}$	0.0668
$\frac{3}{4}$	0.227	$\frac{5}{3}$	0.0478

It is important to remember that the Q-function is not symmetric, which can be shown by the following:

$$Q(t) \neq Q(-t) \tag{5.56}$$

As you can see in the Q-function values provided in Table 5.6,

$$\begin{aligned} Q\left(\frac{1}{4}\right) &= 0.401, \text{ and} \\ Q\left(-\frac{1}{4}\right) &= 0.599 \\ \therefore Q\left(\frac{1}{4}\right) &\neq Q\left(-\frac{1}{4}\right) \end{aligned}$$

proving that the Q-function is asymmetric.

In fact, it can be shown that the following is true when reversing the Q-function:

$$Q(-t) = 1 - Q(t) \tag{5.57}$$

To test this, we can use the same example as before:

$$\begin{aligned} Q\left(-\frac{1}{4}\right) &\stackrel{?}{=} 1 - Q\left(\frac{1}{4}\right) \\ 0.599 &= 1 - 0.401 \\ 0.599 &= 0.599 \end{aligned}$$

Therefore, we can see that Equation 5.57 holds true.

Using what we just learned about linear combinations of Gaussians, we can use the Q-function on **any** Gaussian RV. The probability that a Gaussian RV exceeds the value t is defined in the following equation, with σ **representing the standard deviation of noise.**

$$P\{x > t\} = Q\left(\frac{t-\mu}{\sigma}\right), \quad x \sim \mathcal{N}(\mu, \sigma^2) \tag{5.58}$$

In other words, if we center a Gaussian RV (i.e. noise in a communications system) and normalize its standard deviation, we can apply the Q-function. The Q-function is used extensively in calculating error rates from noise modeled as a Gaussian RV. $Q(t)$ cannot be written in closed form, which means the integral cannot be evaluated symbolically. Software packages instead provide tables of values for the Q-function found numerically.

Let's consider a digital communication system, shown below in Figure 5.8, which transmits +1 Volt to represent a 1 and transmits -1 Volt to represent 0 on a noisy link.

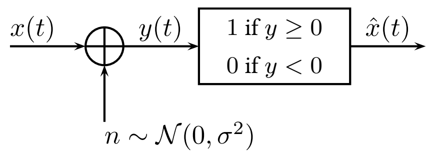


Figure 5.8: A communication system.

The noisy received signal would be a sum of RVs

$$y = x + n$$

where x is the transmitted signal, a random sequence of ± 1, and n is an independent additive noise term defined as a Gaussian RV.

$$n \sim \mathcal{N}(0, \sigma^2)$$

The receiver guesses a 1 was transmitted if $y \geq 0$ and 0 otherwise.

$$\hat{x} = \begin{cases} 1 & y \geq 0 \\ 0 & y < 0 \end{cases}$$

The overall bit error rate can be estimated by considering one case. Given a 1 was transmitted, the probability that a 0 is estimated is the joint probability

$$P\{\text{“0” estimated }|\text{“1” transmitted}\} = P\{y < 0 | x = 1\}$$

Given a particular value of x, it acts like a constant in the equation $y = x + n$.

$$y = 1 + n$$

The PDF of y is then the noise Gaussian with a new average from the given value of x

$$f_{Y|X}(y|x) = \mathcal{N}(1, \sigma^2)$$

Since Y is a Gaussian RV, we can use the Q-function to find the probability that Y is less than 0. This happens when the noise is large and negative enough to move the received value to the negatives, resulting in an error. The noise (before it has mean x) is symmetrical, so the probability that it is less than -1 is equal to the probability that it exceeds 1. Thus, the probability of error in this $x = 1$ case is

$$P\{\text{“0” estimated }|\text{“1” transmitted}\} = Q\left(\frac{1}{\sigma}\right)$$

We might observe that this problem is symmetrical, as depicted below in Figure 5.9.

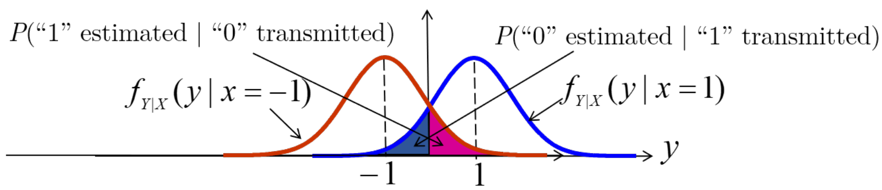


Figure 5.9: Symmetry of a communication system plagued by additive Gaussian noise.

In this case, this means that the probability of error is the same whether a 0 or a 1 was transmitted

$$\begin{aligned} P\{\text{“0” estimated }|\text{“1” transmitted}\} &= P\{\text{“1” estimated }|\text{“0” transmitted}\} \\ &= Q\left(\frac{1}{\sigma}\right) \end{aligned}$$

If 0 and 1 are equally likely to be transmitted, the overall probability of error is found using Bayes' Rule, shown in Equation 5.27. Equation 5.60 below represents all possible outcomes.

$$\begin{aligned} P\{\text{error}\} &= P\{\text{error, “1” transmitted}\} + P\{\text{error, “0” transmitted}\} && (5.59) \\ &= \frac{1}{2}P\{\text{“0” estimated}|\text{“1” transmitted}\} + \frac{1}{2}P\{\text{“1” estimated}|\text{“0” transmitted}\} && (5.60) \\ &= \frac{1}{2}Q\left(\frac{1}{\sigma}\right) + \frac{1}{2}Q\left(\frac{1}{\sigma}\right) && (5.61) \\ &= Q\left(\frac{1}{\sigma}\right) && (5.62) \end{aligned}$$

Example 5.6

If the noise in the aforementioned communications system has variance $\sigma^2 = 4$, what is the probability of error?

Solution:
Consulting Q-function tables, the value of $Q(\frac{1}{2})$ gives a 30.9% probability of error.

x	$Q(x)$	x	$Q(x)$
$\frac{1}{4}$	0.401	1	0.159
$\frac{1}{3}$	0.369	$\frac{5}{4}$	0.106
$\frac{1}{2}$	0.309	$\frac{4}{3}$	0.0912
$\frac{2}{3}$	0.252	$\frac{3}{2}$	0.0668
$\frac{3}{4}$	0.227	$\frac{5}{3}$	0.0478

Example 5.7

In MATLAB, verify that the Q-function estimates the probability of error in this additive noise channel.

Solution:

The following code demonstrates this scenario in MATLAB, confirming the Q-function with greater accuracy the more entries are simulated.

```
% Generate a vector of 10000 +-1 entries, where randn produces Gaussian distributed
     random variables and sign generates a +1 if the value is greater than or
    equal to zero, and a -1 if the value is less than zero.
x = sign(randn(10000,1));
sigma = 2;

% Generate a vector 10000 sigma noise values
n = sigma * randn(10000,1);

% Send x through the noisy channel (summing your input and noise)
y = x + n;

% Estimate x after noise. For each entry in y, decide if it corresponds to a +1 or
    -1
x_est = sign(y);

% Find bit error rate (BER)
BER = sum(x_est ~= x) / 10000 % roughly 0.3

% Verify that BER matches the prediction
qfunc(1/sigma) % = 0.3085
```

Example 5.8

Consider the following equation

$$Y = X + N$$

where N is a random variable which represents noise and is independent of X. The random variable Y is used to obtain an estimate of X which is denoted by $\hat{X}$. Suppose that X can take one of two values, V_1 and V_2 and is estimated by comparing Y to a threshold γ as follows:

$$\hat{X} = \begin{cases} V_1 & \text{if } Y < \gamma \\ V_2 & \text{if } Y \geq \gamma \end{cases}$$

An error is the event that $\hat{X} \neq X$.
The Probability Density Function (PDF) of N is

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

(a) Suppose that $\gamma = 0$, $V_1 = -1$ and $V_2 = 1$.

i. What is the probability of error assuming that $X = 1$ and $X = -1$ are equally likely? You can leave your answer in terms of the Q-function.

ii. What is the probability of error assuming that $X = 1$ with probability $\frac{1}{3}$? You can leave your answer in terms of the Q-function.

(b) Suppose that $V_1 = -1$ and $V_2 = 3$, $\Pr(X = -1) = \frac{1}{2}$, and $\sigma^2 = 4$.

i. Find the probability of error assuming $\gamma = 0$.

ii. If possible, find a different γ for which the probability of error is lower than that you computed in the previous part.

(c) Suppose that $V_2 = V$ and $V_1 = -V$, $\Pr(X = V_1) = \frac{1}{2}$ and $\sigma^2 = 9$. Find the smallest **integer** value of V such that the total probability of error is less than 0.1.

Solution:

a) i. The probability of error is

$$P_{err} = Q\left(\frac{1}{\sigma}\right)$$

ii. The probability of error is

$$\begin{aligned}
P\left\{error|X=1\right\} &= Q\left(\frac{1}{\sigma}\right) \\
P\left\{error|X=-1\right\} &= Q\left(\frac{1}{\sigma}\right) \\
\therefore P\left\{error\right\} &= P\left\{X=1\right\} Q\left(\frac{1}{\sigma}\right) + (1 - P\left\{X=1\right\})Q\left(\frac{1}{\sigma}\right) \\
&= \left(\frac{1}{3}\right) Q\left(\frac{1}{\sigma}\right) + \left(1 - \frac{1}{3}\right) Q\left(\frac{1}{\sigma}\right) \\
&= Q\left(\frac{1}{\sigma}\right)
\end{aligned}$$

b) i. The probability of error in each case is

$$
\begin{aligned}
P\left\{error|X=-1\right\} &= Q\left(\frac{t-\mu}{\sigma}\right)\\
&= Q\left(\frac{0-(-1)}{\sigma}\right)\\
&= Q\left(\frac{1}{2}\right)\\
P\left\{error|X=3\right\} &= Q\left(\frac{6-3}{\sigma}\right)\\
&= Q\left(\frac{3}{2}\right)
\end{aligned}
$$

The overall probability of error is then

$$
\begin{aligned}
P\left\{error\right\} &= P\left\{X=-1\right\}Q\left(\frac{1}{2}\right)+P\left\{X=3\right\}Q\left(\frac{3}{2}\right)\\
&= \frac{1}{2}Q\left(\frac{1}{2}\right)+\frac{1}{2}Q\left(\frac{3}{2}\right)\\
&= \frac{1}{2}\times 0.309+\frac{1}{2}\times 0.0668\\
&= 0.1545+0.0332\\
\therefore P\left\{error\right\} &= 0.1877
\end{aligned}
$$

ii. Consider $\gamma = 1$

$$
\begin{aligned}
P\left\{error|X=-1\right\} &= Q\left(\frac{1-(-1)}{\sigma}\right)\\
&= Q\left(\frac{2}{2}\right)\\
&= Q(1)\\
P\left\{error|X=3\right\} &= Q\left(\frac{5-3}{\sigma}\right)\\
&= Q\left(\frac{2}{2}\right)\\
&= Q(1)
\end{aligned}
$$

The overall probability of error is then

$$
\begin{aligned}
P\left\{error\right\} &= Q(1)\\
&= 1.59\times 10^{-1}
\end{aligned}
$$

which is less than the previous part.

c) The probability of error is

$$P\{error\} = Q\left(\frac{V}{3}\right)$$

We want this to be less than 0.1

$$Q\left(\frac{V}{3}\right) < 0.1$$

$V = 4$ is the smallest solution

$$Q\left(\frac{4}{3}\right) = 0.0912$$

since $V = 3$ does not satisfy the condition

$$Q\left(\frac{3}{3}\right) = 0.159$$

Therefore,

$$V = 4$$

5.6 *Problem Set 3a: Applying Probability and Random Variables*

Problem	Topic	Points
1	Probability Density Functions	1
2	Random Variable Arithmetic	3
3	Conditional Probability	3
Total:		7

Exercise 5.1

(1 points) An *exponential* random variable X has a probability density function (PDF):

$$f_X(x) = \lambda e^{-\lambda x} \quad \text{for } x \geq 0$$

and zero for $x < 0$. Find the following

a) $E[X]$

b) The probability that X exceeds its mean

Exercise 5.2

(3 points) Consider a random variable $Y = X + W$ where X and W are independent. X is equally likely to be either $+1$ or -2 and W is uniformly distributed between -3 and 3.

a) What is the variance, σ^2, of W?

b) What is the variance, σ^2, of the random variable V, where $V = 3W$?

c) What is the probability that $Y < 0$ given that $X = -2$?

d) What is the probability that $Y < 0$ given that $X = +1$?

e) What is the probability that $Y > 0$?

Exercise 5.3

(3 points) You are trying to decide between two routes to take on your drive home for the holidays.

If you take the scenic route, the time it takes for your drive home is uniformly distributed between 4 and 8 hours. If you take the freeway, the time it takes is uniformly distributed between 3 and 6 hours. You flip a fair coin to decide which route to take.

a) What is the probability that it takes more than 5 hours to get home?

b) If you use this route-planning strategy a large number of times, what is the average amount of time you spend on your drive home for the holidays?

c) Suppose that you decide that you want to take the scenic route less frequently. You decide to take the scenic route if the outcome of a roll a fair, six-sided die is 2 or less. You take the freeway otherwise. How do your answers in the previous two parts change?

5.7 *Problem Set 3b: Applying Q-Function and Random Variables*

Problem	Topic	Points
1	Gaussian Random Variables and the Q-function	2
2	Error Probability from Additive Noise	5
3	Probability of Bit-Error in MATLAB	4
Total:		11

Exercise 5.4

(2 points) Recall that a Gaussian distributed random variable has the following PDF:

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

which is frequently denoted as $\mathcal{N}(\mu, \sigma^2)$ where μ is the mean and σ^2 is the variance of the random variable X.
The following is the definition of the Q-function which cannot be evaluated in closed form (i.e., it must be numerically computed):

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

Consider the random variable Y which is distributed according to $\mathcal{N}(\mu, \sigma^2)$.

a) Using the property that any linear function of a Gaussian random variable is also a Gaussian random variable find $P\{Y > y\}$ in terms of the Q-function.

b) Find the probability that Y is between a and b. Your answer should be expressed in terms of the Q function, μ, σ, a and b.

Exercise 5.5

(5 points) Consider the following equation

$$y[k] = x[k] + n[k]$$

$n[k]$ is a Gaussian random variable with zero mean and variance, σ^2. Suppose that $x[k]$ and $n[k]$ are independent random variables,

a) Suppose that $x[k]$ can take values of $\pm V$, with equal probability, and $V > 0$. In other words $\Pr(x[k] = V) = \Pr(x[k] = -V) = \frac{1}{2}$. You estimate $x[k]$ by comparing $y[k]$ with zero. Let the estimate of $x[k]$ be denoted by $\hat{x}[k]$. Then $\hat{x}[k] = V$ if $y[k] \geq 0$, and $\hat{x}[k] = -V$ if $y[k] < 0$.

  i. Please compute the probability of error in estimating the transmitted signal, $x[k]$, from the received signal, $y[k]$, or in other words, what is $\Pr(\hat{x}[k] \neq x[k])$?

  ii. In this part assume that $\sigma^2 = 1$. Suppose that you require the probability of error to be less than 10^{-7}. To an accuracy of one decimal place, what is the minimum value of V to achieve this target probability of error? You may find the table of values for the Q-function available on the web or on page 220 of Proakis and Salehi useful for this part.

  iii. Please write a MATLAB script that verifies that your choice of V achieves the target probability of error in the previous part. Please show your code and the output. You may find the following functions useful: `randn, qfunc, sign`.

b) Suppose that $x[k]$ can take 4 different values $-V, -\frac{1}{3}V, \frac{1}{3}V$, and V with equal probability. In other words, $\Pr(x[k] = -V) = \Pr(x[k] = -\frac{1}{3}V) = \Pr(x[k] = \frac{1}{3}V) = \Pr(x[k] = V) = \frac{1}{4}$. An estimate of $x[k]$, denoted by $\hat{x}[k]$, is made from $y[k]$ using the following function:

$$\hat{x}[k] = \begin{cases} V & \text{if} \quad \frac{2}{3}V \leq y[k] \\ \frac{1}{3}V & \text{if} \quad 0 \leq y[k] < \frac{2}{3}V \\ -\frac{1}{3}V & \text{if} \quad -\frac{2}{3}V \leq y[k] < 0 \\ -V & \text{if} \quad y[k] < -\frac{2}{3}V \end{cases}$$

  i. How many bits of information can $x[k]$ represent?

  ii. On the same axes, sketch the conditional PDFs of $y[k]$ for each possible choice of $x[k]$, i.e. please sketch the following
  - $f_{y[k]|x[k]=V}(y[k]|x[k] = V)$
  - $f_{y[k]|x[k]=\frac{1}{3}V}\left(y[k]\,\middle|x[k] = \frac{1}{3}V\right)$
  - $f_{y[k]|x[k]=-\frac{1}{3}V}\left(y[k]\,\middle|x[k] = -\frac{1}{3}V\right)$
  - $f_{y[k]|x[k]=-V}(y[k]|x[k] = -V)$.

  iii. In your sketch from the previous part, identify the regions whose areas correspond to probabilities of error in estimating $x[k]$ from $y[k]$.

  iv. Find the following in terms of the Q-function:
  - $\Pr\left(\hat{x}[k] \neq V\,|x[k] = V\right)$
  - $\Pr\left(\hat{x}[k] \neq \frac{1}{3}V\,\middle|x[k] = \frac{1}{3}V\right)$
  - $\Pr\left(\hat{x}[k] \neq -\frac{1}{3}V\,\middle|x[k] = -\frac{1}{3}V\right)$
  - $\Pr\left(\hat{x}[k] \neq -V\,|x[k] = -V\right)$

  v. What is the probability of error in estimating $x[k]$ from $y[k]$?

Exercise 5.6

(4 points) Consider a simplified model of a communications channel described by the following equation

$$y[k] = Ax[k] + n[k]$$

Suppose that for each k, $n[k] \sim \mathcal{N}(0, \sigma^2)$, and $A > 0$ is a constant that models the attenuation caused by the channel. Suppose that $x[k] = \pm V$ and is equally likely to be V or $-V$, with V representing a bit value of 1, and $-V$ representing a bit value of 0.

a) As a function of A, V and σ, what is the probability of error in estimating $x[k]$ from $y[k]$?

b) Download the file `noisy_channel.m` , please do not scroll down if you open this file. This file contains a function which simulates a channel of the form given in the equation above. It takes a vector `x` as its input and returns a vector `y` as its output. By generating appropriate `x` vectors, estimate A and σ. You may find MATLAB's `var` function helpful here.

c) Find the smallest value of V for which the probability of error is less than or equal to 10^{-4}. You may find MATLAB's `qfuncinv` and/or `qfunc` helpful here.

d) Generate a long vector (e.g. 10^7 or more) with random $\pm V$ values for the value of V you found in the previous part. Simulate transmitting this vector through the channel and check if the probability of error is as you predicted. Note that you will have to figure out how to calculate the number of errors here. Hint: calculating the error rate from simulated data does not involve the Q-function.

This surprisingly simple model for a communication channel is actually applicable in a number of scenarios in wireless communications. This is called a *flat-fading* model, although typically, we model A, $x[k]$ and $n[k]$ as complex numbers. Real-world systems also typically estimate the attenuation caused by the channel and the noise in the channel in a manner similar to what you did above. With knowledge of A and σ, you can minimize V to meet a probability of error requirement. This would result in a minimization of the power used. Alternatively, if you allow $x[k]$ to take on multiple levels, you could try to maximize the number of levels that $x[k]$ is allowed to take, for a given probability of error and power limit. This would result in a maximization of the data rate in with constraints on the probability of error and power.

Chapter 6: Random Processes

Random processes map the sample space to signals. A sample of a random process is a random variable. E.g. the sample of a noise wave form at a particular instant in time is a random variable.

Statistical properties of random processes $X(t)$ or $X[n]$ can change over time, (t continuous or n discrete). Possible realizations of processes, such as $x_1(t)$ and $x_2(t)$, as shown in Figure 6.1, can be considered, but which realization is the outcome of an experiment is usually unknown.

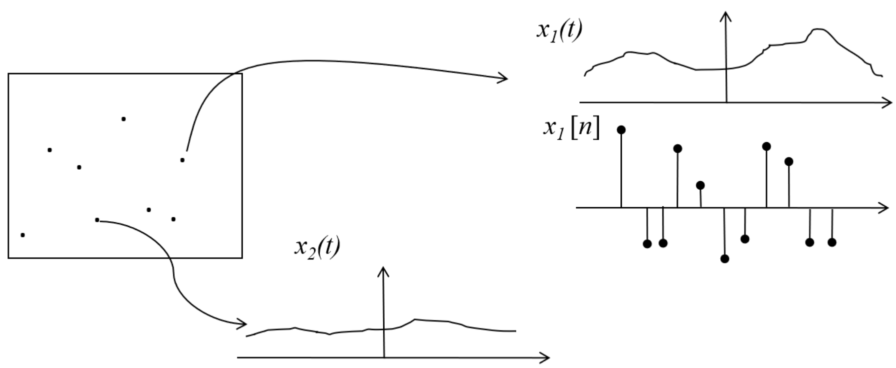


Figure 6.1: A sample space (left), possible realizations of the continuous random process $X(t)$, and a realization of the discrete process $X[n]$ (right).

Some examples of random processes are a microphone recording a speech waveform on a cell phone or an antenna/scope probe picking up noise. Figure 6.2 shows some possible waveforms picked up by a microphone. Note that the possible realizations of a random process could be drawn from a continuum of possibilities.

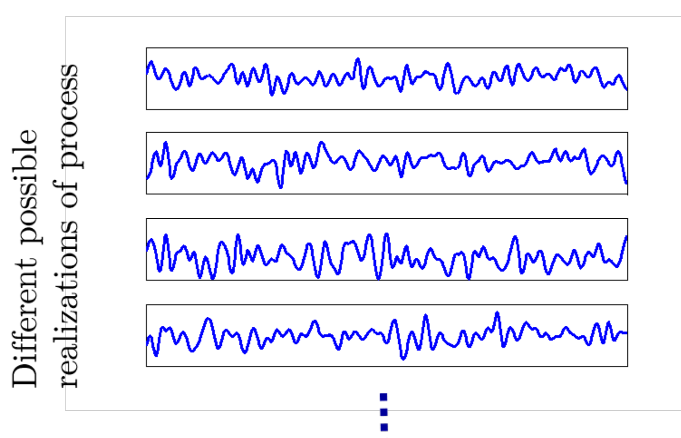


Figure 6.2: Some possible cell phone microphone waveforms, all realizations of a random process.

In general, statistical properties of random processes can change significantly over time. In this text however, we will restrict ourselves to random processes for which certain statistical properties remain constant over time. These processes are called **wide-sense-stationary** random processes. Statistical properties (mean, variance and covariance) are preserved under time shifts for stationary processes.

A mathematical result of the WSS assumption is that the mean (expected value from the process) is constant

$$E[X(t)] = \mu = \text{constant} \tag{6.1}$$

The auto-correlation is a function of a time-lag τ, which has units of time, but the function itself remains constant with time.

$$E[X(t-\tau)X(t)] = R_{XX}(\tau) \tag{6.2}$$

If a process is **not** stationary, then the previous two equations would instead be time-dependent, i.e.,

$$E[X(t)] = \mu(t) \tag{6.3}$$

$$E[X(t-\tau)X(t)] = R_{XX}(\tau, t) \tag{6.4}$$

Auto-correlation is covered in detail later in this text, but roughly speaking it describes similarity of a process to itself at a later (or earlier) time. To test for this similarity, correlation asks whether the product of samples from the process tends to be positive, negative, or near zero by taking an average. For example, suppose you are stationary and yell at a canyon wall to hear an echo a second later. What you hear will be positively correlated to what you yelled at lag $\tau = 0$ (when you hear your voice) and at lag $\tau = 1$ (when the echo reaches your ear). You expect the correlation between what you yelled and what you heard to be high at $\tau = 1$ because the waveforms have a similar shape at this delay.

6.1 *Action of Linear Time-Invariant Systems on Stationary Processes*

Now we will begin to connect the mathematical tools we learned in Chapter 5 to communication systems. It is helpful to know the relationships between the statistical properties of the input and output of system, such as in Figure 6.3. Cross-correlation gives us powerful tools to explore the relationships in systems like this.

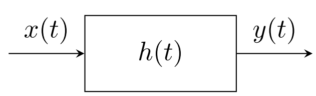


Figure 6.3: An example system with input $x(t)$ and output $y(t)$

What happens to a white noise process when it passes through an LTI system, such as a low pass filter? An ideal white noise process has equal content in all frequencies, but Figure 6.5 shows a realization where this can only be approximately true.

Passing the above white noise through a 50 Hz low pass filter produces a new signal with the FFT in Figure 6.4 below.

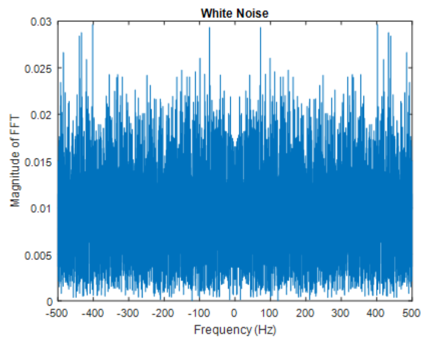


Figure 6.4: Real white noise with only approximately equal amplitude across frequencies.

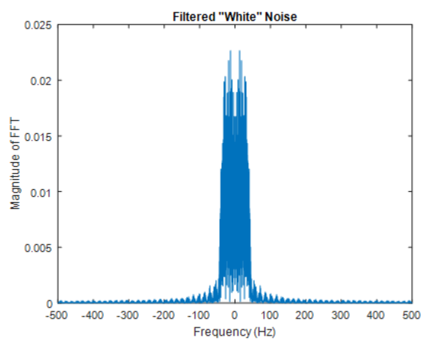


Figure 6.5: Filtered white noise with frequency only in the pass band of a 50 Hz low pass filter.

What are the new statistical properties of this filtered noise? You will learn how to track changes to mean, variance, and other properties when random processes are subjected to LTI systems. More importantly, you will learn how to characterize an LTI system using indeterminate (random) inputs and outputs.

6.2 *Auto-correlation and Cross-correlation*

A random process at any given time is a random variable. Correlation describes how the value of a random process at one time predicts the value of another (possibly the same) process at another time. A correlation function is the measure of this predictability.

Correlation between two different functions is called the cross-correlation. The cross-correlation between random processes X and Y would be evaluated as the expected value

$$R_{XY}(\tau) = E[X(t+\tau)Y(t)] \tag{6.5}$$

In this work, we have chosen to define the cross-correlation as Equation 6.5. Some authors define the cross-correlation as $E[X(t)Y(t-\tau)]$.

When the correlation is evaluated for the same function compared to itself, it is called the **auto-correlation**.

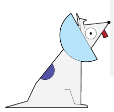
$$R_{XX}(\tau) = E[X(t+\tau)X(t)] \tag{6.6}$$

The notation $R_{XX}(\tau)$ is often abbreviated to $R_X(\tau)$. Why is $E[X(t+\tau)Y(t)]$ a measure of correlation? If $X(t+\tau)$ tends to be large whenever $Y(t)$ tends to be very large, then the product of $X(t+\tau)$ and $Y(t)$ numbers is typically large as well. In other words, $E[X(t+\tau)Y(t)]$ would be large and positive and suggest strong correlation. If the product instead tended to be large and negative, X and Y would be inversely correlated. If X and Y are not related to each other, the average value of their products would tend towards a minimum (zero if both are zero-mean processes).

Fun Facts from Doggo

In MATLAB, one can compute the auto-correlation or cross-correlation by using the `xcorr` function, where `xcorr(x)` computes the auto-correlation of random process `x`, and `xcorr(x,y)` computes the cross-correlation between random processes `x` and `y`.

In addition, we can find the lags at which the correlations are computed by entering in MATLAB `[Rxy, lags] = xcorr(x)` for auto-correlation or `[Rxy, lags] = xcorr(x,y)` for cross-correlation. `lags` gives us the value of τ, which represents the shift forward in time.



To plot the correlation, we can use the `stem` function, where `stem(lags, Rxy)` creates a stem plot of either the auto- or cross-correlation as defined earlier.

Cross-correlation is used in our cell phone systems and has additional applications such as in radar instrumentation. We can use the cross-correlation function to determine the time it takes a radar satellite to receive a reflected signal that it transmitted. To explain this further, the signal is originally transmitted from space and contacts and reflects off a medium (such as a building, the ocean, or a given target), then after some time lag, τ, the signal is received from our satellite! To determine τ, we can take the signal we transmit, x_a, and the reflected signal we receive, y_a, and compute the cross-correlation using `xcorr(`y_a`,` x_a`)` in MATLAB. When we perform this computation we find Ryx.

In a simple radar or echo channel, the input and output will be most correlated at $t = 0$ and $t = n_0$. Other points in the cross-correlation R_{YX} are expected to be negligible.

Since $R_{XX}[m] = A\delta[m]$ (i.e., X is a zero-mean white-noise process) and $R_{YX} = R_{XX} * h[k]$, an equation for the impulse response can be written as follows:

$$R_{YX} = A\delta[k] * h[k] \tag{6.7}$$

$$= Ah[k] \tag{6.8}$$

solving for our channel, h[k]

$$h[k] = \frac{1}{A} R_{YX}[k], \text{where} \tag{6.9}$$

$$A = R_{YX}[0] \tag{6.10}$$

The impulse response can also be written using the given difference equation

$$y[n] = \beta x[n] + \alpha x[n - n_0] \tag{6.11}$$

$$h[k] = \beta\delta[k] + \alpha\delta[k - n_0] \tag{6.12}$$

A is chosen such that the impulse at zero is consistent with the given information that $\beta = 1$. The impulse response equations found from the difference equation and cross-correlation can be used to find the values of α and β

$$\beta\delta[k] + \alpha\delta[k - n_0] = \frac{1}{R_{YX}[0]} R_{YX}[k] \tag{6.13}$$

In MATLAB, $R_{YX}[0]$ is found using the syntax: `Ryx(lags==0)`.

Given $R_{YY} = R_{XX} * h(\tau) * \overleftarrow{h}(\tau)$ and $R_{XX}[m] = \delta[m]$ (X is a zero-mean white-noise process), R_{YY} becomes

$$R_{YY} = h(\tau) * \overleftarrow{h}(\tau)$$

Graphically, R_{YY} is found by convolving the known form of the impulse response assuming unknown values α and n_0, as shown in Figure 6.6.

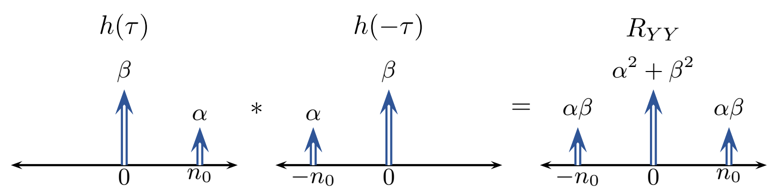


Figure 6.6: R_{YY} when R_{XX} is a zero mean white noise process assuming no particular values for α, β, and n_0

When we solve for the channel, or impulse response, $h(\tau$

For stationary processes, recall that the correlation function is the same for any value of t (varies only with the lag between test points τ). A visualization for a particular value of t, t_0 is shown in Figure 6.7.

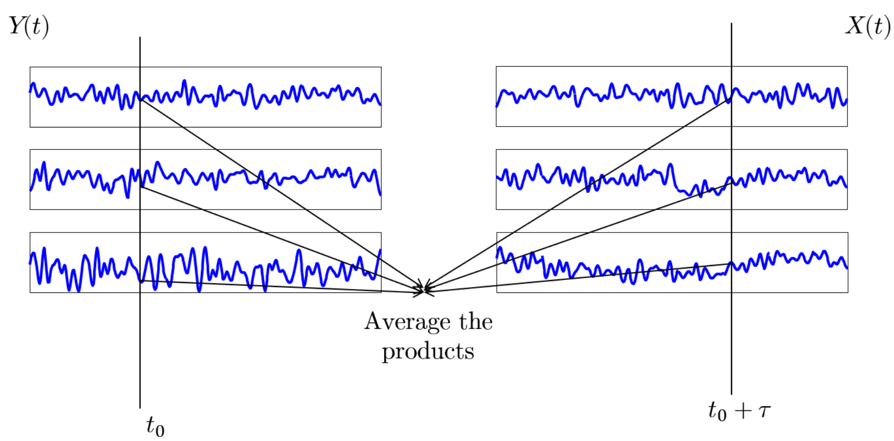


Figure 6.7: The cross-correlation of stationary random process $Y(t)$, left, and $X(t)$, right, found by averaging the products of $X(t)$ and $Y(t)$ for some time t_0.

Considering the figure, what happens to the correlation if the sign of τ is changed or if the order of X and Y is switched? Each of these would change the result, but it turns out that applying both time-reversal and switching the order returns the original cross-correlation. We can see this in the following equations, where we define s as t - τ.

$$R_{XY}(\tau) = E[X(t-\tau)Y(t)] \tag{6.14}$$
$$= E[X(s)Y(s-\tau)] \tag{6.15}$$
$$= R_{YX}(-\tau) \tag{6.16}$$

In the case of auto-correlation, this property is

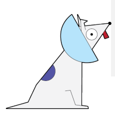
$$R_{XX}(\tau) = R_{XX}(-\tau) \tag{6.17}$$

Fun Facts from Doggo

The cross-correlation function of stationary random processes $X(t)$ and $Y(t)$ is

$$R_{XY}(\tau) = E[X(t+\tau)Y(t)] \tag{6.18}$$

which states that the cross-correlation function is the typical product of two random processes at times t and $t+\tau$. It has the following property for time-reversal



$$R_{XY}(\tau) = R_{YX}(-\tau) \tag{6.19}$$

The cross-correlation function can be used in many different applications, including in detection of weak signals embedded in noise. One scenario where this is used is in detecting the absence or presence of an object in a radar application. In its simplest form, a radar transmits a known electromagnetic signal and listens for the reflection of that signal off an object. If the object is present, the signal is reflected back, and if not, the radar observes noise. The amount of time it takes for the signal to return tells the distance to the object.

Echo cancellation, another application of correlation, is implemented far and wide in modern computer audio hardware to prevent sound from speakers feeding back into microphones. Using cross-correlation between the known speaker output and the observed microphone input, the feedback can be estimated and removed from the microphone signal before transmission to a call. When this control loop is absent, the feedback amplifies continuously to no-end in a positive feedback loop.

Correlation can be used to describe changes to random processes acted upon by LTI systems. Suppose that X is a real-valued stationary random process and that passes through a real transformation $h(t)$. Let Y the output process when $h(t)$ is applied to X. Just like before, the application of $h(t)$ to X is described by convolution.

$$Y = (X * h)(t) \tag{6.20}$$

How correlated is the input to the output after applying $h(t)$? Perhaps $h(t)$ delays the signal to a later time or changes its amplitude. To find out how exactly X is related to Y, write the cross-correlation R_{YX}

$$R_{YX}(\tau) = E[Y(t+\tau)X(t)] \tag{6.21}$$

Using Eq. 6.20, replace $Y(t)$ with $(X * h)(t)$

$$R_{YX}(\tau) = E[(X * h)(t+\tau)X(t)] \tag{6.22}$$

Recalling the definition of convolution from Eq. 2.50, write the correlation in terms of an integral

$$R_{YX}(\tau) = E[X(t)\int_{-\infty}^{\infty} h(s)X(t+\tau-s)ds] \tag{6.23}$$

The expected value operator can be moved inside the integral[1], and $h(s)$ remains outside of it because it is deterministic (has all known values).

$$R_{YX}(\tau) = \int_{-\infty}^{\infty} h(s)E[X(t)X(t+\tau-s)]ds \tag{6.24}$$

The expected value term should now look familiar, it is an auto-correlation for X! Next, write as such

$$R_{YX}(\tau) = \int_{-\infty}^{\infty} h(s)R_{XX}(\tau-s)ds \tag{6.25}$$

Using the convolution integral definition again, write the final result

$$R_{YX}(\tau) = R_{XX} * h(\tau) \tag{6.26}$$

This derivation may have been hard to follow, but it tells us a simple relationship between the correlations before and after an LTI system $h(t)$. Similar derivations give two additional relationships that are helpful in communications analysis.

$$R_{YX}(\tau) = R_{XY} * \overleftarrow{h}(\tau) \tag{6.27}$$

$$R_{YY}(\tau) = R_{XX} * h(\tau) * \overleftarrow{h}(\tau) \tag{6.28}$$

The time-reversed impulse response $\overleftarrow{h}(t)$ was used above. It is shorthand for $h(-t)$

$$\overleftarrow{h}(t) = h(-t) \tag{6.29}$$

Time reversal has a notable Fourier Transform

$$FT\{\overleftarrow{h}(t)\} = H^*(f) \tag{6.30}$$

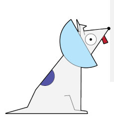
Fun Facts from Doggo

Correlations between the input $x(t)$ and output $y(t)$ of an LTI system impulse response $h(t)$ give three important relationships

$$R_{YX}(\tau) = R_{XX} * h(\tau) \tag{6.31}$$

$$R_{YX}(\tau) = R_{XY} * \overleftarrow{h}(\tau) \tag{6.32}$$

$$R_{YY}(\tau) = R_{XX} * h(\tau) * \overleftarrow{h}(\tau) \tag{6.33}$$



Depending on how much is known about X, Y, and h, these functions can be used to solve for unknown signals or impulse responses.

[1] *we implicitly assume here that the conditions on $X(t)$ and $h(t)$ that allow the expected value operator to be moved into the integral hold*

6.3 *Power Spectral Density*

The power distribution of a random signal at different frequencies is useful like the amplitude distribution from the Fourier Transform. We define the power spectral density (PSD) of a random signal $x(t)$ as

$$S_{XX}(f) = FT\{R_X(\tau)\} = \int_{-\infty}^{\infty} R_X(\tau)e^{-2\pi f \tau}d\tau \tag{6.34}$$

The PSD tells us how much signal power there is **on average** for different frequencies of a random signal. Important note: the PSD here is in terms of *signal* power, not physical power. Because it is a measure of density, it only makes sense to evaluate for a range of frequencies, e.g.,

$$\int_{f_1}^{f_2} S_{XX}(f)df = \text{The average energy of the random signal } X(t) \text{ that lies in the range } f_1 < f \leq f_2 \tag{6.35}$$

This is called the Einstein-Weiner-Khinchin Theorem. The average signal power of a signal $X(t)$ across all frequencies is $E[X^2(t)]$. This happens to be equal to the variance of the signal.

$$E[X^2(t)] = \sigma^2 \tag{6.36}$$

The expected value side of the equation also suggests that the auto-correlation at lag zero gives signal power

$$E[X^2(t)] = R_X(0) \tag{6.37}$$

Contrasting signal power to physical power, the above equation is only proportional to the real world power; time and impedance/resistance are not considered. To turn a signal power into a physical power, divide by the impedance and symbol rate to get

$$P_X = \frac{1}{TR}E[X^2(t)] = \sigma^2 \tag{6.38}$$

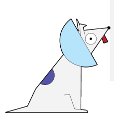
where X has units of Volts, T has units of seconds, and R has units of Ohms. This way, the power has units of physical power in Watts.

Fun Facts from Doggo

Applying the Fourier Transform to the auto-correlation function produces another function called the Power Spectral Density (PSD) with units of $\frac{V^2}{Hz}$. It is denoted S_{XX}.

$$S_{XX}(f) = FT\{R_{XX}(t)\} \tag{6.39}$$

Conceptually, the PSD describes how much power is concentrated at a particular frequency, but because it is a **density** it only has a non-zero value when integrated over a frequency interval.



You may predict that the PSD of a random process changes when a system acts on that process. It can be shown that the relationship between the input PSD and output PSD of a process depends on the frequency response of that system.

$$S_{YY}(f) = S_{XX}(f)|H(f)|^2 \tag{6.40}$$

Example 6.1

Consider the following system

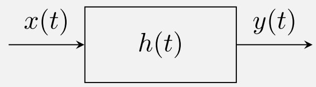


Figure 6.8: An LTI system with frequency response $h(t)$.

with a band pass filter as its frequency response

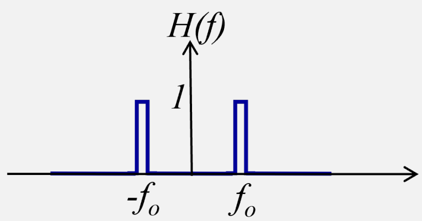


Figure 6.9: A band pass filter frequency response.

What is the power of the output of the system if the input has known power spectral density $S_{XX}(f)$?

Solution:

The power of the output is the power of X that lives in the pass band, which we can find using methods we have learned so far.

$$E[Y^2(t)] = \int_{-\infty}^{\infty} S_{YY}(f)df \tag{6.41}$$

$$= \int_{-\infty}^{\infty} S_{XX}(f)|H(f)|^2 df \tag{6.42}$$

$$= \int_{\text{PASS-BAND}} S_{XX}(f)df \tag{6.43}$$

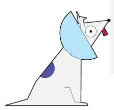
In the final step, notice that integration limits are known because $H(f)$ is zero outside of the pass-band.

Many different expressions were presented in this chapter that are equivalent. For reference, they are gathered here.

Fun Facts from Doggo

In a system with input random process X, impulse response $h(t)$, and output random process Y, the following expressions describe the signal power of Y.

$$\begin{aligned} var\{Y\} &= \sigma_Y^2 && (6.44)\\ &= E[Y^2(t)] && (6.45)\\ &= R_{YY}(0) && (6.46)\\ &= \int_{-\infty}^{\infty} S_{YY}(f)df && (6.47)\\ &= \int_{-\infty}^{\infty} S_{XX}(f)|H(f)|^2 df && (6.48) \end{aligned}$$



These equations relate the variance, signal power, auto-correlation, and power spectral density of a process subject to an LTI system.

6.4 *Noise Processes*

Real-world communication systems transmit bits/groups of bits by mapping them into waveforms and sending the waveforms through channels (e.g., wires, glass, etc.). The receiver looks at the received waveform and guesses what bits were sent. Received waveforms are inevitably corrupted by noise, and the result can be modeled as the addition of a *Gaussian random variable*.

$$Y = X + N \qquad N \sim \mathcal{N}(\mu, \sigma^2) \tag{6.49}$$

The original transmitted waveform might look like Figure 6.10(a). The received waveform after filtering might look like Figure 6.10(b), (c), or even (d) as more and more noise is added to the signal. To mitigate these effects and others, an understanding of how a channel's impulse response effects signal and noise random variables is needed.

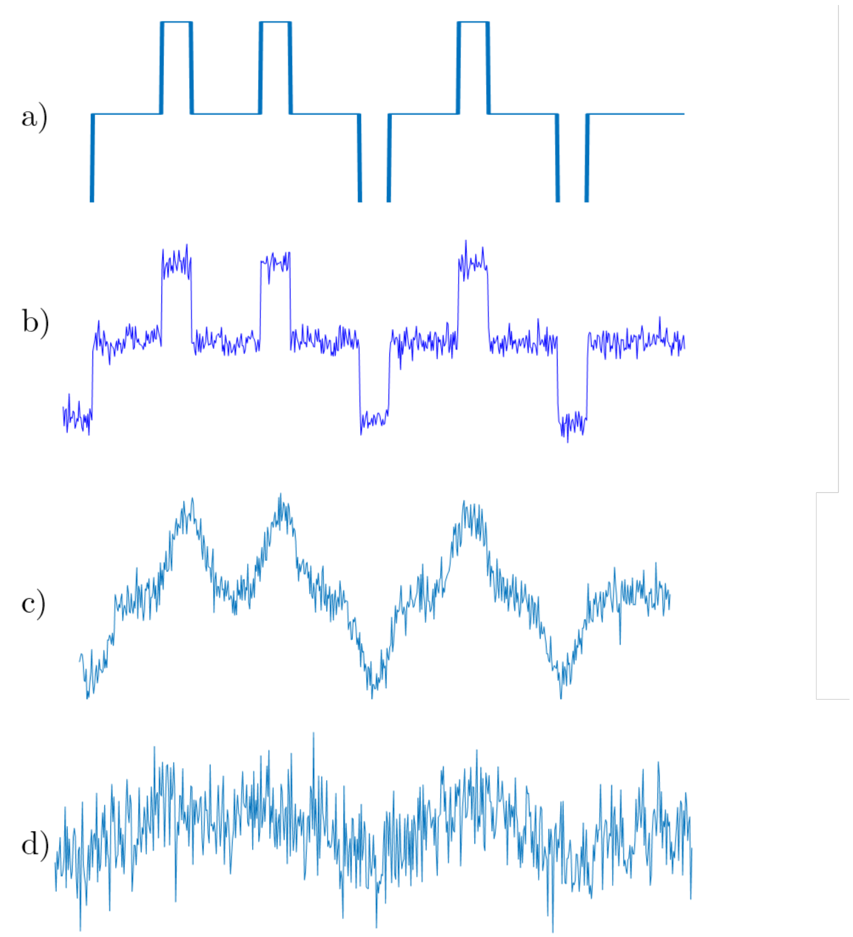


Figure 6.10: A transmitted signal (a) and progressively worse received signals (b), (c), and (d).

White Noise Processes

A white noise process $N(t)$ has zero-mean and is perfectly uncorrelated to itself at an earlier or later time. The autocorrelation $R_{NN}(t)$ of such a process is an impulse at zero

$$R_{NN}(t) = \frac{N_0}{2}\delta(t) \tag{6.50}$$

where $\frac{N_0}{2}$ is a constant called the noise power spectral density (PSD). It follows that, for an ideal white noise process, white noise is uncorrelated to itself at any lag except zero

$$R_{NN}(\tau) = 0 \quad \tau \neq 0 \tag{6.51}$$

Consider the PSD of true white noise, i.e. the Fourier Transform of the auto-correlation in Equation 6.50.

$$S_{NN}(f) = \frac{N_0}{2} \tag{6.52}$$

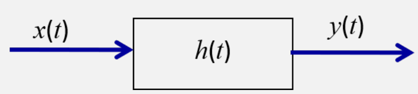
The above equation shows that white noise has equal power across all frequencies. This is not a surprising result as in Chapter 2 white noise was shown to have equal amplitude across all frequencies. The total signal power of true white noise is then the integral of $S_{NN}(f)$ for all frequencies.

$$P_N = \int_{-\infty}^{\infty} \frac{N_0}{2} df \tag{6.53}$$

Thus, it has been shown that true white noise processes have infinite power when considering all frequencies.

Example 6.2

Suppose that $x(t)$ is a white noise process with PSD $\frac{N_0}{2}$. What is the variance of the output $var\{y(t)\}$ in terms of $H(f)$ and $\frac{N_0}{2}$?



Solution:

To find the variance, write the integral of the PSD

$$var\{y(t)\} = \int_{-\infty}^{\infty} S_{YY}(f) df \tag{6.54}$$

Using Equation 6.40, substitute $S_{YY}(f) = S_{XX}(f)|H(f)|^2$ then replace $S_{XX}(f)$ with its known value $\frac{N_0}{2}$.

$$var\{y(t)\} = \int_{-\infty}^{\infty} S_{XX}(f)|H(f)|^2 df \tag{6.55}$$

$$= \frac{N_0}{2} \int_{-\infty}^{\infty} |H(f)|^2 df \tag{6.56}$$

Aside: Thermal Noise

Thermal noise is a more realistic model of noise in communication systems. For demonstrative purposes, this text does not use it in favor of ideal white noise. Thermal noise is similar to white noise in that its spectrum is nearly flat, but it does not contain theoretically infinite power.

From quantum mechanics, it is known that the PSD of thermal noise is

$$S_N(f) = \frac{hf}{2\left(e^{\frac{hf}{kT}} - 1\right)} \tag{6.57}$$

where $h = 6.6 \times 10^{-34}$ [J/s], Boltzmann's constant $k = 1.3806503 \times 10^{-23}$[m^{2}kg s^{-2}K^{-1}], and T is the temperature in Kelvin. As seen in Figure 6.11, there is significant rolloff at 2×10^{12} [Hz].

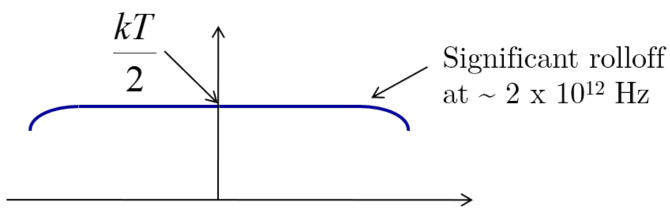


Figure 6.11: The PSD of thermal noise, which we typically model as white noise.

Because the rolloff does not occur until extremely high frequencies(orders of magnitude beyond those seen in this text) approximating thermal noise as white noise is sufficient. However, temperature is sometimes an important consideration still because it dictates the PSD of observed noise in the real world.

Noise in antennae and other hardware is often specified in terms of temperature because Equation 6.57 depends on it. Suppose an antenna/receiver with a temperature of 1000 K has equivalent noise PSD to a perfectly designed receiver at 1000 K. The resulting thermal noise PSD $\frac{N_0}{2}$ would then be approximately

$$\frac{N_0}{2} = kT = 1000k \quad \text{[W/Hz]} \tag{6.58}$$

Another metric often used is the Noise Figure and is specified in dB. Suppose that a receiver front end (antenna + amplifiers) has a noise figure of X dB. This means that the noise PSD is X dB higher than a perfect receiver at 290 K. The noise PSD $\frac{N_0}{2}$ would then be approximately

$$\frac{N_0}{2} = 290k \cdot 10^{X/10} \quad \text{[W/Hz]} \tag{6.59}$$

6.5 *Problem Set 4: Random Processes*

Problem	Topic	Points
1	Discrete Time Echo Channel	3
2	Impulse Response from Training Signal	2
3	Radar and Cross-Correlation	5
4	Filtered White Noise	3
Total:		13

Exercise 6.1

(3 points) Consider the following simple DT model of an echo channel:

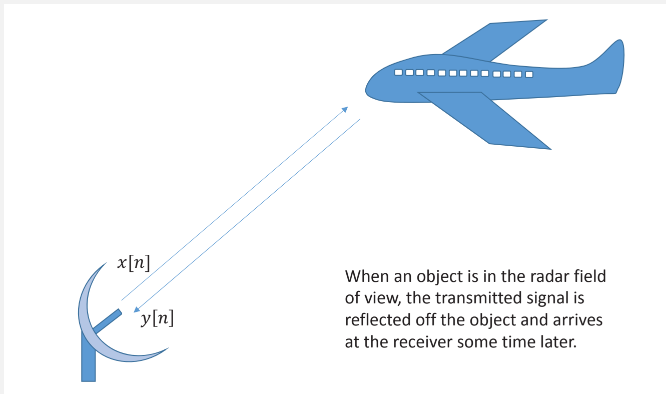
$$y[n] = \beta x[n] + \alpha x[n - n_0]$$

where the output signal $y[n]$ is the sum of the scaled input signal $x[n]$ and a scaled, delayed version of the input signal. It is known that α and β are positive. Please load the file `ps4_echo.mat` in MATLAB. Please include your code and stem plots for each part of the problem. You will find the `xcorr` function with the `lags` output helpful here. You can look up its usage on the MATLAB help, by typing `help xcorr`. Remember that for radar systems, we find the cross-correlation by comparing the output signal to the input signal, rather than vice-versa.

a) The vectors `ya` and `xa` contain $y[n]$ and $x[n]$ respectively. In MATLAB, please estimate n_0 and α assuming that $\beta = 1$, and use the `stem` function to show the output of the cross-correlation graphically.

b) The vector `yb` contains $y[n]$ which was generated with $x[n]$ equal to a zero-mean, white-noise process with $R_{xx}[m] = \delta[m]$. Please estimate n_0 and α in this case assuming that $\beta = 1$, and use the `stem` function to show the output of the cross-correlation graphically. Note that the answer to this part may be different from the previous part.

c) The vector `yc` contains $y[n]$ which was generated with $x[n]$ equal to a zero-mean, white-noise process with $R_{xx}[m] = \delta[m]$. Please estimate n_0 and α in this case, and use the `stem` function to show the output of the cross-correlation graphically. The answer to this part may be different from the previous part and β may not necessarily equal 1. Use the `unbiased` option for cross-correlation for this problem.



Exercise 6.2

(2 points) Load the file `ps4_vlc.mat` in MATLAB. This file contains two vector `x` and `y_sig`. `x` is a vector of Independent and Identically Distributed (IID) ± 1 values. The values in `x` were transmitted through a real-life Visible-Light Communication (VLC, or so-called LiFi) channel and the received signal is stored in `y_sig`. Please estimate and plot the impulse response of the VLC channel.

Hint: Zoom in a lot to get a good visualization of the impulse response.

Exercise 6.3

(5 points) Imagine a simple radar system that detects objects when a transmitted signal is reflected back and detected among some noise. Consider the following two difference equations where $y_0[k]$ or $y_1[k]$ are the received signals corresponding to two different scenarios depending on whether or not an object is present, $x[k]$ is the transmitted signal and $n[k] \sim \mathcal{N}(0, \sigma^2)$ is a white-noise signal which is independent of $x[k]$. α and k_0 are unknown.

$$y_0[k] = n[k] \tag{6.60}$$
$$y_1[k] = \alpha x[k - k_0] + n[k] \tag{6.61}$$

a) Which of the difference equations above describes a situation where there is an object present, and why?

b) What is the cross correlation of $y_0[k]$ and $x[k]$?

c) What is the cross correlation of $y_1[k]$ and $x[k]$?

d) Load the file `ps4_radar.mat` in MATLAB. It contains vectors `x`, which represents a transmitted radar pulse, and `ya` and `yb`, which are two different received signal waveforms. One of these represents a situation when an object is present, and the other represents a situation where there is no object present.

   i. Plot `ya` and `yb`. By looking at the plots, can you tell which of the two corresponds to a system where an object is present?

   ii. Using the `xcorr` function in MATLAB and appropriate plots, determine which of `ya` and `yb` corresponds to a system where there is an object present. You may find it helpful to have the `xcorr` function return the "lags" at which the cross-correlation is evaluated. Please see the documentation for `xcorr` for details.

   iii. For the signal corresponding to the situation where there is an object present, determine k_0.

   iv. What important radar-related information does k_0 give you?

Exercise 6.4

(3 points) Consider a zero-mean, white Gaussian noise process $N(t)$ with power-spectral-density (PSD) $\frac{N_0}{2}$. Suppose that this process is the input to a system with frequency response $H(f)$ and the output of this system is $Y(t)$.

a) Let $H(f)$ be given by the following:

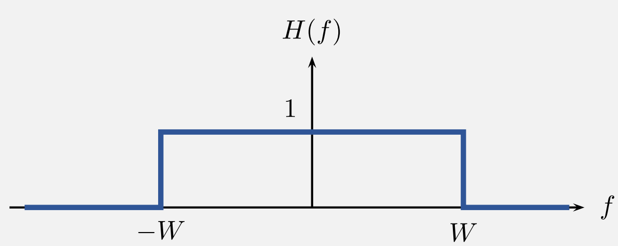


Figure 6.12: Frequency response $H(f)$

   i. What is $E[Y(t)]$?

   ii. What is $E[Y(t)^2]$?

   iii. What is $R_{YY}(\tau)$?

   iv. For what value(s) of τ (if any) are the random variables $Y(t)$ and $Y(t+\tau)$ uncorrelated?

b) Let $H(f)$ be given by the following:

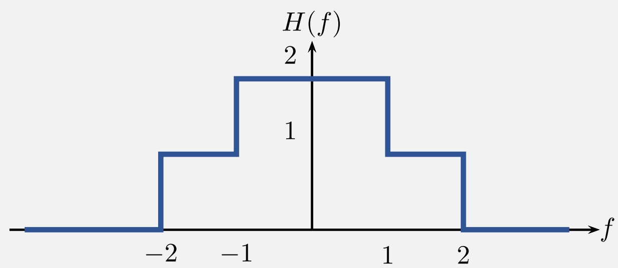


Figure 6.13: Frequency response $H(f)$

i. What is $E[Y(t)]$?
ii. What is $E[Y(t)^2]$?

Chapter 7: Pulse Amplitude Modulation (PAM) and Inter-Symbol Interference (ISI)

7.1 *Pulse Amplitude Modulation*

Thus far, a broad set of statistical tools has been introduced, and serve as background required to analyze complex transmission systems. Pulse amplitude modulation (PAM) is a communications technique that sends symbols in a particular manner, at some regular interval, to increase the likelihood that the receiver will decipher the initial message, $m[k]$, with little error.

Impulse Train Transmitter/Receiver

Figure 7.1 provides a schematic of a communications system. The message signal, $m[k]$, is encoded and then transmitted as $x(t)$ through our channel.

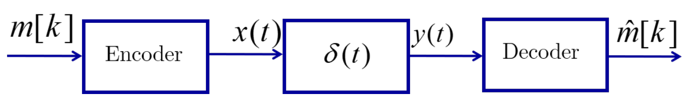


Figure 7.1: A relatively simple transmitter/receiver system.

The transmitted input signal, $x(t)$, is defined as

$$x(t) = \sum_{k=-\infty}^{\infty} m[k]\delta(t - kT) \tag{7.1}$$

and the received output signal, $y(t)$, is defined as

$$y(t) = \sum_{k=-\infty}^{\infty} m[k]\delta(t - kT) \tag{7.2}$$

The estimated received message is represented by $\hat{m}[k]$, and is defined as

$$\hat{m}[k] = \int_{kT-\Delta}^{kT+\Delta} y(\tau)d\tau \tag{7.3}$$

A pictorial representation of the transmitted and received signals is depicted in Figure 7.2.

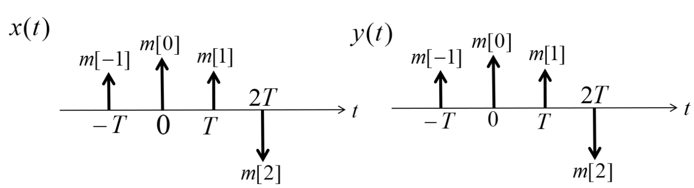


Figure 7.2: The transmitted (left) and received (right) signals of a simple system.

Each $m[k]$ is referred to as a **symbol**, and the rate at which they occur, known as the symbol rate is defined as $\frac{1}{T}$.

It is important to note that impulses are not realizable in the real world, so instead small width pulses, defined with the function $p(t)$, are sent at the location of each impulse, scaled by the transmit data, as shown in Figure 7.3. The receiver then samples the received waveform at every T time units.

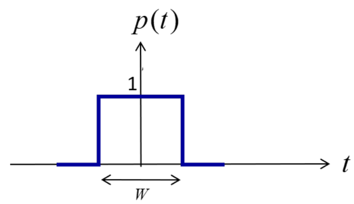


Figure 7.3: A simple transmitted pulse, a box with height 1 and width W.

With these changes, the system diagram becomes that in Figure 7.4.

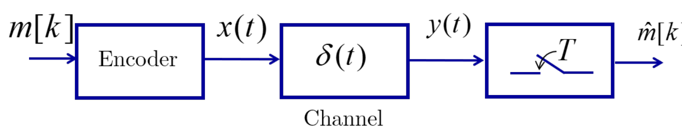


Figure 7.4: A Pulse Amplitude Modulation (PAM) system with an ideal channel ($x(t) = y(t)$)

The encoder converts the message, $m[k]$, into the transmitted signal, $x(t)$, which is a sequence of pulses with heights determined by the message in this case. Recall the result when an impulse is convolved by any signal: the signal is copied to the position of the impulse. So, convolving the impulse train by the pulse, $p(t)$, will produce a pulse train for $x(t)$.

$$x(t) = \left(\sum_{k=-\infty}^{\infty} m[k]\delta(t - kT) \right) * p(t)$$

$$x(t) = \sum_{k=-\infty}^{\infty} m[k]p(t - kT) \tag{7.4}$$

The objective is to transmit these pulses as quickly as possible, minimizing the symbol period, T, without a sampling point at $t = nT$ overlapping with a previous pulse, as seen in Figure 7.5.

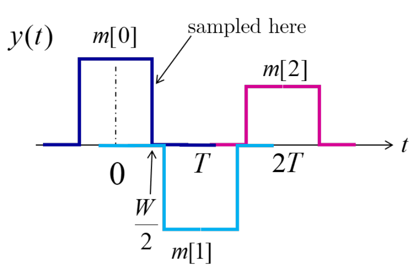


Figure 7.5: A received sequence of pulses for the message $[1, -1, 0.6]$.

From Figure 7.5, we can see that the symbol period, T, is barely large enough that the boxes do not overlap. This means that a given pulse function, $p(t)$, should be zero whenever a different pulse is being measured and one when it is being measured. Therefore, an ideal channel needs the following to be true:

$$p(nT) = \begin{cases} 1 & \text{when } n = 0 \\ 0 & \text{otherwise} \end{cases} \tag{7.5}$$

Impulse Train Transmitter with Non-Ideal Channel

Consider a channel impulse response, $h(t)$, that is not ideal, or in other words that is not $\delta(t)$. To model transmission of an impulse train with the area of the impulses given by $m[k]$ (no pulse function $p(t)$ for now), use the system diagram shown in Figure 7.6

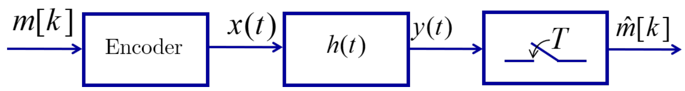


Figure 7.6: A system that transmits an impulse train through a channel $h(t)$.

and the equations

$$x(t) = \sum_{k=-\infty}^{\infty} m[k]\delta(t-kT) \tag{7.6}$$

$$y(t) = \left(\sum_{k=-\infty}^{\infty} m[k]\delta(t-kT)\right) * h(t) = \sum_{k=-\infty}^{\infty} m[k]h(t-kT) \tag{7.7}$$

$$\hat{m}[k] = y(kT+\tau) \tag{7.8}$$

In the equation for $\hat{m}[k]$, note $kT+\tau$ instead of kT because the channel response could introduce a delay. In other parts of this text, it is assumed that the receiver samples at the correct times, so only kT will be written for simplicity. Notice that a channel is modeled in the same way as a pulse function: by convolving impulse responses. With this strategy, any LTI system between the source and destination can be effectively modeled.

Effective Channel PAM

It is time to put the previous sections together into a system that transmits pulses $p(t)$ through a channel $h(t)$ to a receiver with a filter $f(t)$.

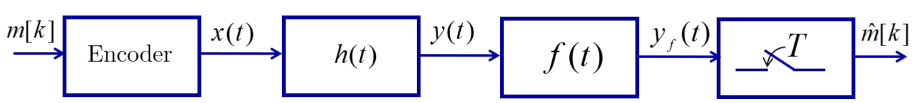


Figure 7.7: A generalized PAM system with transmitted pulse $p(t)$, channel response $h(t)$, and receive filter $f(t)$.

The framework developed allows analysis of a generalized system mathematically by combining the pulse, channel, and filter into an "effective" channel. Generally, $h(t)$ is uncontrolled and there are some restrictions on $p(t)$, so the receiver filter $f(t)$ is used to account for any problems that arise. The transmitted signal $x(t)$ after the encoder is still a pulse train

$$x(t) = \sum_{k=-\infty}^{\infty} m[k]p(t - kT) \tag{7.9}$$

but the signal the decoder receives $y_f(t)$ has passed through two impulse responses, the channel, $h(t)$, and filter, $f(t)$.

$$y_f(t) = \left(\sum_{k=-\infty}^{\infty} m[k]p(t - kT) \right) * h(t) * f(t) \tag{7.10}$$

The pulse function can be factored out of the summation

$$y_f(t) = \left(\sum_{k=-\infty}^{\infty} m[k]\delta(t - kT) \right) * p(t) * h(t) * f(t) \tag{7.11}$$

Consider the effective channel $r(t) = p(t) * h(t) * f(t)$ formed by $p(t)$, $h(t)$, and $f(t)$. An equivalent model of a PAM system can then be created with $r(t)$ as in Figure 7.8.

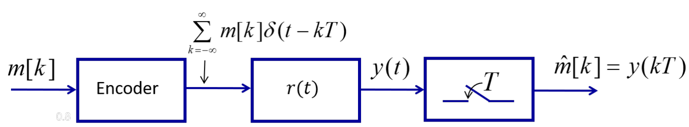


Figure 7.8: An generalized PAM system with the pulse, channel, and receive filter combined into a single effective channel $r(t)$.

With this model, a wide variety of systems described by various functions for the transmitter pulse, receiver filter, and the channel between can be considered.

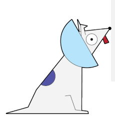
Fun Facts from Doggo

When the transmitted pulse $p(t)$, channel $h(t)$, and receiver filter $f(t)$ are combined into one impulse response, the result is called the *effective channel* $r(t)$

$$r(t) = p(t) * h(t) * f(t) \tag{7.12}$$

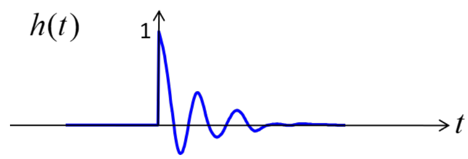
In the frequency domain, $R(f)$ is the product of the frequency responses instead

$$R(f) = P(f)H(f)F(f) \tag{7.13}$$

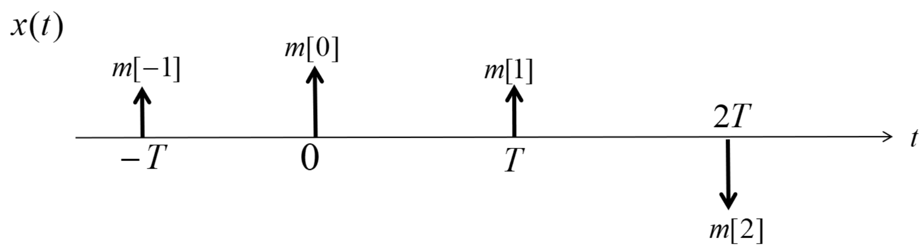


7.2 *Inter-Symbol Interference*

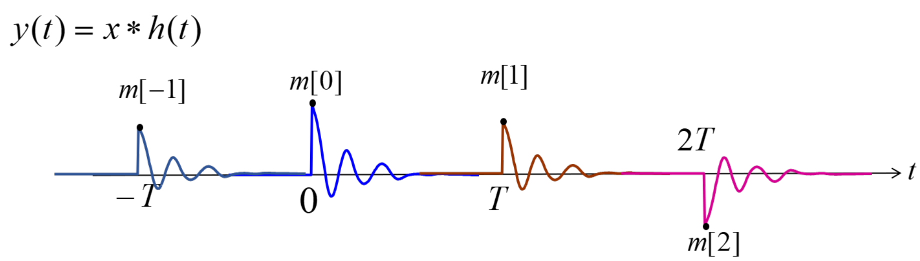
To see the effect of a channel, suppose that the channel impulse response in a PAM system is



with transmitted impulses $x(t)$



At the receiver, the samples are marked by dots. There is minimal interference between symbols if the symbol period T is large enough.



If the symbol period is too small compared to the length of the channel impulse response, symbols interfere with each other. This Inter-Symbol Interference (ISI) causes information to be lost as in Figure 7.9 where the value of one symbol is changed by other symbols.

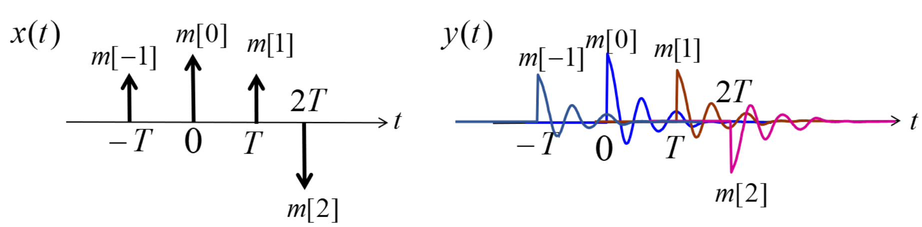


Figure 7.9: Inter-Symbol Interference caused by a too small symbol period T.

With some clever choices of the symbol period and transmit pulses, this inter-symbol interference can be eliminated.

Eliminating ISI

In Figure 7.10, suppose that the channel ($h(t)$) is an ideal low-pass filter. Many channels are modeled as low-pass filters because transmission media such as cables attenuate higher frequencies.

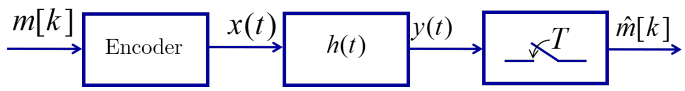


Figure 7.10: A system that encodes discrete messages $m[k]$ into a signal $x(t)$, then sends $x(t)$ through a channel $h(t)$ and decodes the result by sampling.

Specifically, let the frequency response and impulse response of the channel be as in Figure 7.11.

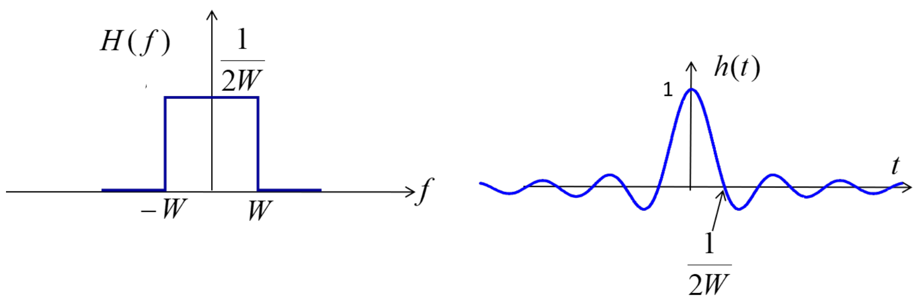


Figure 7.11: The frequency response (left) and impulse response (right) of an ideal low-pass filter.

If some information in the form of impulses with the symbol period $T = \frac{1}{2W}$ is transmitted, then the zeros of the sinc function align perfectly so that the value of some received pulse is not changed by any of the other pulses. This choice of T is special because it has allowed a fast symbol rate to be used in a low-pass channel without corrupting the sent information. The result is graphically shown in Figure 7.12.

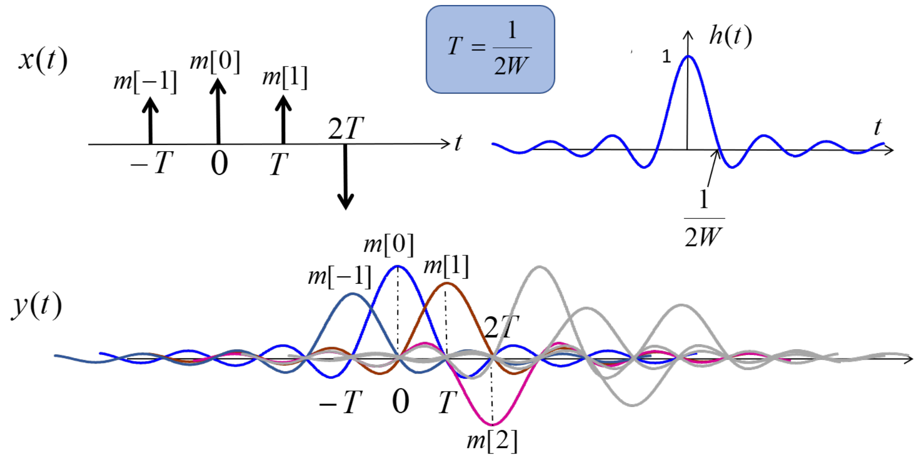


Figure 7.12: A visual depiction of *equalization*. When data are transmitted at a specific frequency, the peaks of the filtered impulses line up with the zeros of other impulses. This eliminates interference between symbols and minimizes the symbol period, maximizing the data rate without introducing error!

Data are transmitted with period $T = \frac{1}{2W}$ to make the value of any given symbol align with the zeros of **all** other symbols. This allows quick data transmission even though there could be dramatic ISI at all times other than kT. In other words, if the receiver does not sample at these precise times, the ISI is theoretically infinite as in Figure 7.13).

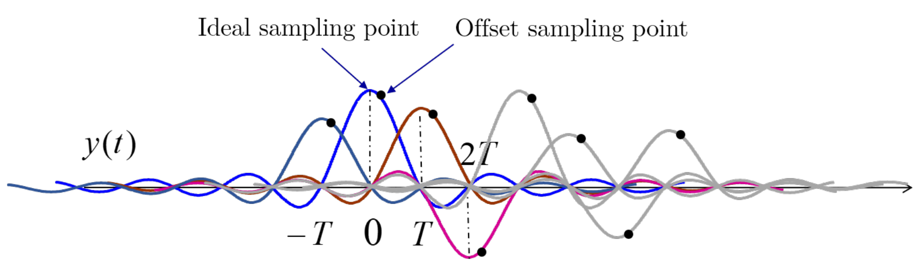


Figure 7.13: When data are transmitted with a valid period for zero ISI but the receiver samples with a timing delay, the resulting ISI is infinite.

A practical system will never be perfectly synchronize the transmitter and receiver, so the offset sampling point error will always be present. A different function from the sinc function reduces this error. A favorable alternative effective channel impulse response should decay rapidly to zero. The *raised cosine* does just that and is explained in the next section.

Raised Cosine Impulse Response

A practical solution for this problem is a new impulse response. The raised cosine pulse has the same zero-crossings as the sinc function, but it approaches zero far faster. This reduces the cost of desynchronization to a more reasonable level **at a bandwidth cost**. It is defined by the piecewise functions for $0 \leq \beta \leq 1$

$$h(t) = \begin{cases} \frac{\pi}{4T}\text{sinc}\left(\frac{1}{2\beta}\right), & t = \pm\frac{T}{2\beta} \\ \frac{1}{T}\text{sinc}\left(\frac{t}{T}\right)\frac{\cos\frac{\pi\beta t}{T}}{1-(\frac{2\beta t}{T})^2}, & \text{otherwise} \end{cases} \tag{7.14}$$

$$H(f) = \begin{cases} 1, & |f| \leq \frac{1-\beta}{2T} \\ \frac{1}{2}\left[1+\cos\left(\frac{\pi T}{\beta}\left[|f| - \frac{1-\beta}{2T}\right]\right)\right] & \frac{1-\beta}{2T} < |f| \leq \frac{1+\beta}{2T} \\ 0, & \text{otherwise} \end{cases} \tag{7.15}$$

Figure 7.14 compares the sinc with the raised cosine to depict the quicker decay towards zero in time domain.

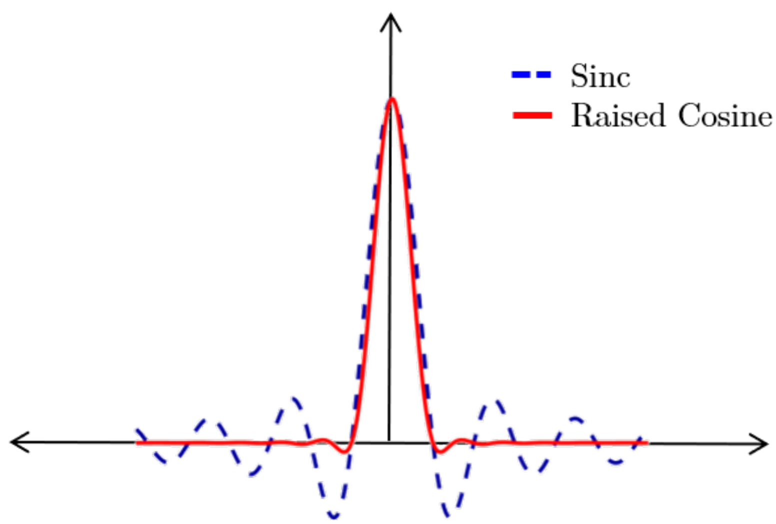


Figure 7.14: The raised cosine is an attractive alternative to the sinc function. It reduces ISI due to sampling offset at a cost of higher bandwidth. A sketch of this function is in the company Broadcom's logo.

Equalization and the Nyquist Criteria for Zero-ISI

Recall the effective channel, $r(t)$ from Section 7.1.

$$r(t) = (p * h * f)(t) \tag{7.16}$$

The mathematical condition for zero-ISI in the effective channel is

$$r(kT) = \begin{cases} A & k = 0 \\ 0 & k \neq 0 \end{cases} \tag{7.17}$$

This is known as *equalization*, and it occurs when the zeros of several transmitted sinc pulses line up perfectly as to not interfere with later pulses, as shown in Figure 7.12. A frequency domain view of the zero-ISI condition offers some insight. Write the result of sending data through the effective channel $r(t)$.

$$r(t) \sum_{k=-\infty}^{\infty} \delta(t - kT) = A\delta(t) \tag{7.18}$$

Then take the Fourier Transform of the result

$$R(f) * \frac{1}{T} \sum_{k=-\infty}^{\infty} \delta(f - \frac{k}{T}) = A \tag{7.19}$$

$$\sum_{k=-\infty}^{\infty} R(f - \frac{k}{T}) = TA \tag{7.20}$$

This is called the Nyquist Criteria for Zero-ISI.

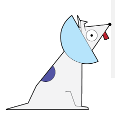
Fun Facts from Doggo

The Nyquist criteria for zero-ISI in the time domain is

$$r(kT) = \begin{cases} A & k = 0 \\ 0 & k \neq 0 \end{cases} \tag{7.21}$$

for integers k where T is the symbol period, A is a constant, and $r(t)$ is the impulse response of the effective channel. The Nyquist criteria for zero-ISI in the frequency domain is equivalently

$$\sum_{k=-\infty}^{\infty} R(f - \frac{k}{T}) = TA \tag{7.22}$$



where $R(f)$ is the frequency response of the effective channel.

To find the right symbol rate for zero-ISI using the frequency domain if the effective channel is a sinc function or a box in the frequency domain as shown below in Figure 7.15, recall the zero ISI condition.

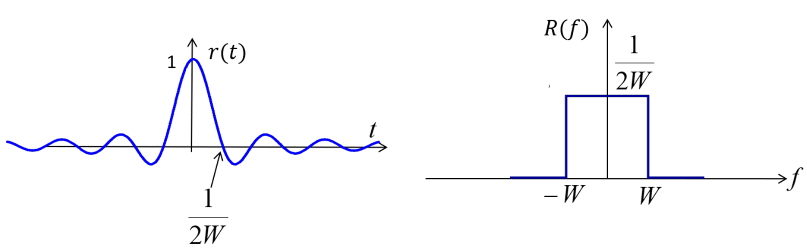


Figure 7.15: A sinc function (left) in frequency domain (right) is our hypothetical effective channel response. The sinc is easy to work with because of its box shape in the frequency domain.

The condition for zero ISI requires the sum of shifted $R(f)$s to equal a constant. The boxes are all the same height, so they must be placed end-to-end , as seen in Figure 7.16.

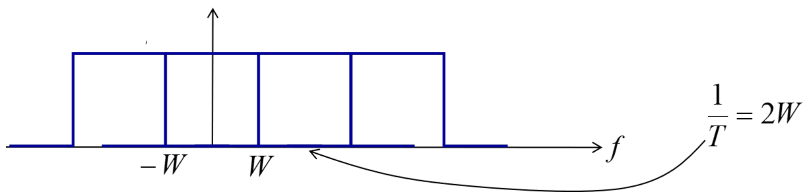


Figure 7.16: The condition for zero ISI when a box with width $2W$ is transmitted is met when the boxes are placed every $2W$. This corresponds to a symbol period of $\frac{1}{2W}$.

In the frequency domain, the boxes must be placed every $2W$ Hz. The symbol period is the reciprocal of this, $T = \frac{1}{2W}$.

Example 7.1

Other geometries also have trivial solutions or multiple solutions, as can be seen below in Figure 7.17.

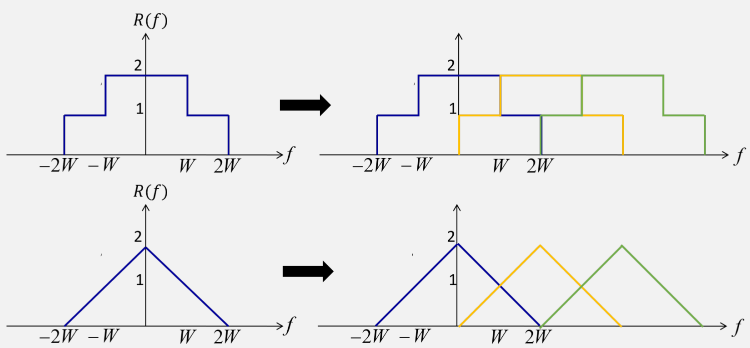


Figure 7.17: Some examples of how the Nyquist criteria for zero-ISI can be met with the right choice of symbol period. Some effective channels have multiple symbol periods that satisfy the zero-ISI condition.

7.3 *Problem Set 5: Inter-Symbol Interference and Noise Power*

Problem	Topic	Points
1	Channel and Noise Estimation	7
2	Eliminating ISI with Raised Cosine	7
3	Finding zero ISI Symbol Periods (OPTIONAL)	2
Total:		14

Exercise 7.1

(7 points) Download the MATLAB script `chan_ps5p1.m` and the data file `ps5p1.mat`. The script simulates a channel with some unknown impulse response (do not look at it) and adds Gaussian noise with a small variance to the input signal. You can simulate this channel by running `chan_ps5p1(x)` where `x` is a vector containing transmitted samples.
The data file contains a vector `x_dat` which contains samples of values ± 1 with equal probability. Please include all MATLAB code and plots used in this problem.

a) Estimate the impulse response of the channel by generating an appropriate training signal, running it through the channel, and using what you know about cross-correlation functions. Note that you can generate a DT white noise signal using MATLAB's `randn` command, for instance. You should use a very large signal vector to get an accurate impulse response estimate, e.g. 10^6.

b) Estimate the symbol period T that will enable zero ISI communication. You may do this visually and present a figure.

c) Generate a transmit signal vector `x` that represents the values in `x_dat` and transmit it through the channel to produce a vector `y = chan_ps5p1(x);`. You should do so in a manner that reduces/minimizes ISI. Sample the received vector `y` at the appropriate times to produce an estimate of the data. MATLAB's `upsample` and `downsample` functions may be useful here.

d) Verify that the estimated data at the receiver matches the transmitted data.

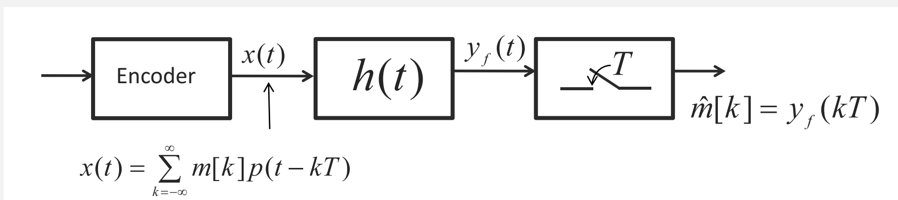
e) Estimate the variance of the noise in the channel. Hints: You may find MATLAB's `var` function useful here. Think about how you can obtain noise samples from this simulated system.

f) Generate a transmit signal vector `x` that represents the values in `x_dat` as in Part c), except this time, the restrict the non-zero values of `x` to ± 0.002. Transmit it through the channel to produce a vector `y = chan_ps5p1(x);`. Sample the received vector `y` at the appropriate times to produce an estimate of the data. Find the number of errors in your transmissions.

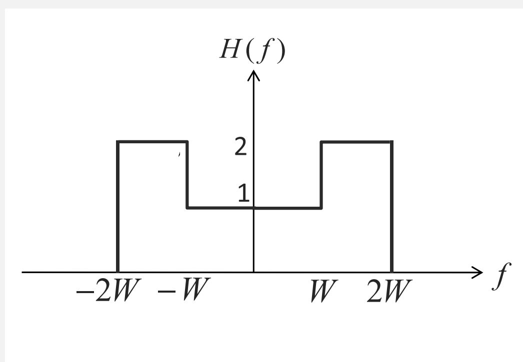
g) Check if your measured error rate agrees with a theoretical prediction based on the Q-function. Explain any observed discrepancy.

Exercise 7.2

(7 points) Consider the following system which is used to communicate bits through a noiseless channel represented by the impulse response $h(t)$. $m[k]$ is a function of the bits encoded in the k-th symbol. Note that it is not important how this encoding occurs, or how many bits are represented by each symbol for this problem.



a) The Fourier transform of $h(t)$ above is



i. Suppose that $p(t) = \delta(t)$. If possible, find a value for T such that there is no Inter-Symbol-Interference (ISI), in other words, find a T such that $\hat{m}[k]$ is not a function of $m[k + k_0]$ for any integer $k_0 \neq 0$.

ii. Assuming that you are free to design $p(t)$, what is the smallest possible value of T such that there is no ISI?

iii. Find $p(t)$ that corresponds to your answer above.

b) Suppose now that $h(t) = \delta(t)$, and $p(t)$ has the raised-cosine frequency characteristic. The following expressions describe $P(f)$ and $p(t)$.

$$P(f) = \begin{cases} T, & |f| \leq \frac{1-\alpha}{2T} \\ \frac{T}{2}\left[1 + \cos\left(\frac{\pi T}{\alpha}\left[|f| - \frac{1-\alpha}{2T}\right]\right)\right], & \frac{1-\alpha}{2T} < |f| \leq \frac{1+\alpha}{2T} \\ 0, & \text{otherwise} \end{cases}$$

and

$$p(t) = \left(\frac{T \sin\left(\pi \frac{t}{T}\right)}{\pi t}\right)\left(\frac{\cos\left(\frac{\pi \alpha t}{T}\right)}{1 - \frac{4\alpha^2 t^2}{T^2}}\right)$$

where α is known as the roll-off factor and it is between 0 and 1, i.e. $0 \leq \alpha \leq 1$.

i. Verify that the Nyquist criteria for zero ISI is met if $\alpha = 0$ and the symbol period is T.

ii. Verify that the Nyquist criteria for zero ISI is met if $\alpha = 1$ and the symbol period is T.

iii. (Optional) Verify that the Nyquist criteria for zero ISI is met for any $0 \leq \alpha \leq 1$.

iv. Using MATLAB or some other software, plot $p(t)$ for both $\alpha = 1$ and $\alpha = 0$. You may find the included script `RaisedCosineIR.m` useful for this (just pick some T, e.g. $T = 10$, and `t` ranging from $-5T$ to $+5T$). Discuss what benefit there is to using $\alpha = 1$ instead of $\alpha = 0$.

The raised-cosine pulse is commonly used in many real systems as it allows the designer to trade-off between exploiting the presence of zero-crossings to reduce symbol periods and reducing the height of the side-lobes in the time domain pulse. By using a raised-cosine pulse, even in channels that are ideal low-pass filters, the sensitivity of the system to errors in symbol periods (i.e. T) is reduced.

Exercise 7.3

(OPTIONAL: 2 points) Consider a system where a pulse $p(t)$ is used to communicate data through an ideal channel with impulse response $\delta(t)$. Suppose that there is no receive filtering and the receiver samples the received signal every T time units.

a) Suppose that $p(t) = \frac{\sin^2(2W\pi t)}{\pi^2 t^2}$. What is the smallest symbol period that will ensure no ISI?

b) Suppose that $p(t) = \frac{\sin(2W\pi t)}{\pi t} + \frac{\sin(4W\pi t)}{\pi t} + \frac{\sin(6W\pi t)}{\pi t}$. What is the smallest symbol period that will ensure no ISI?

Chapter 8: Noise Power and the Cauchy Schwarz Inequality

8.1 *Additive Gaussian Noise*

Suppose a PAM system includes additive noise, $n(t)$, between the transmitter and the receiver, as shown in the system model in Figure 8.1.

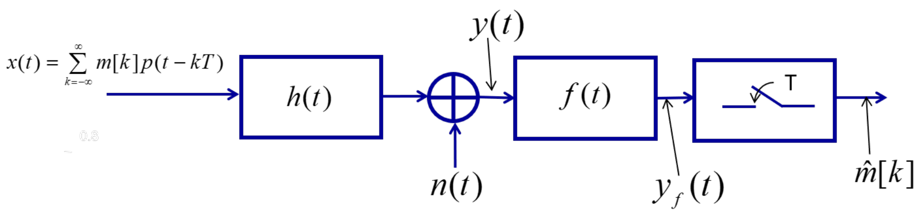


Figure 8.1: An generalized PAM system with additive noise before the receiving filter.

Assume that the noise $n(t)$ is a white Gaussian noise process with PSD $\frac{N_0}{2}$ and, for simplicity, that the channels are ideal.

$$f(t) = h(t) = \delta(t) \tag{8.1}$$

Even if the zero-ISI condition is met, it is incredibly unlikely that the receiver estimates $\hat{m}[k] = m[k]$ because of the noise, as seen in Figure 8.2.

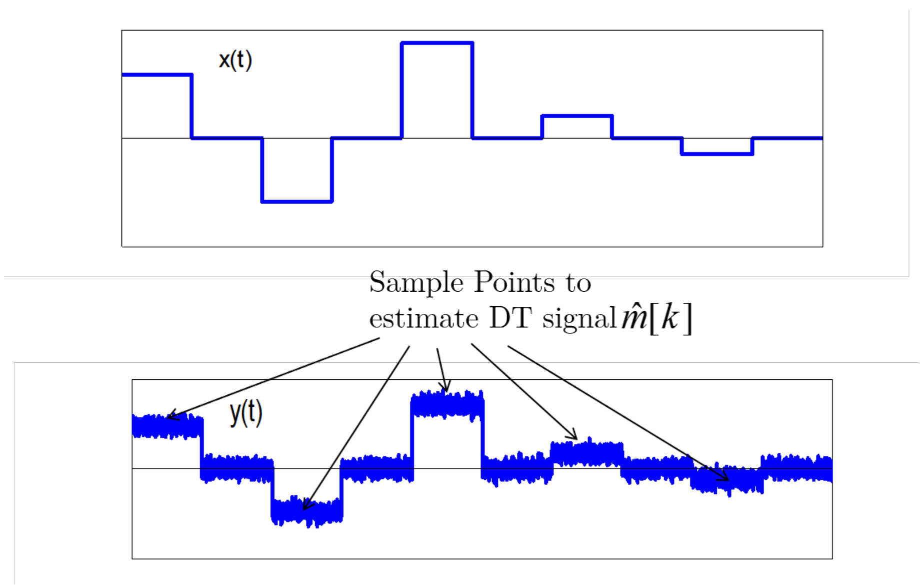


Figure 8.2: A PAM system can be corrupted by noise. Sampling at distinct points introduces significant error depending on the amount of noise.

Sampling at distinct points comes with error in the final estimated data. As shown in Figure 8.3, if the received signal is averaged over each symbol period, a better estimate can be made. However, this may still not be sufficient for real-world systems.

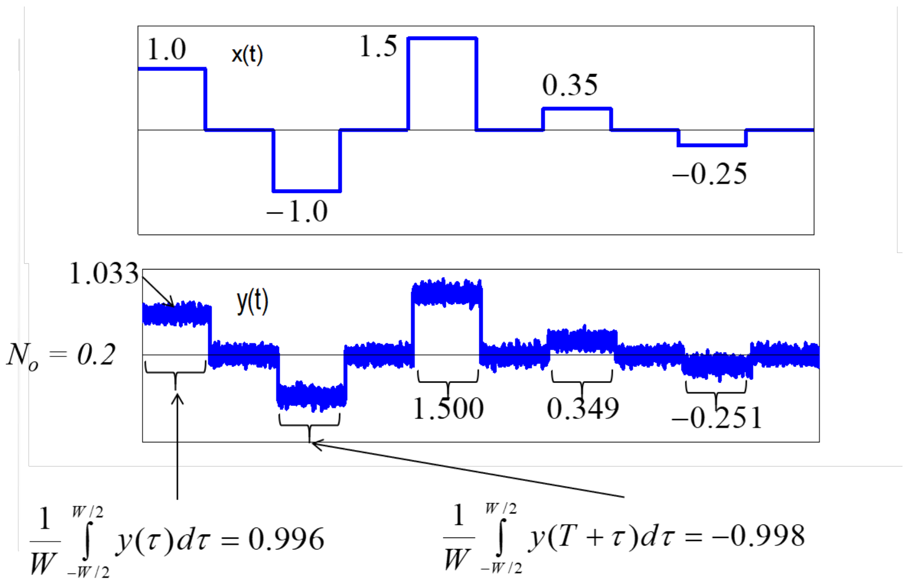


Figure 8.3: A PAM system can be corrupted by noise. Averages of each symbol period are an improvement to sampling at distinct points.

In the following example, we explore the impact of noise on the probability of error for a BPSK system using Matlab.

Example 8.1

Probability of Error for 2-PAM/BPSK in MATLAB

This example simulates the transmission of a Binary Phase Shift Keying (BPSK) signal with additive noise between the transmitter and receiver, then finds the error rate. Because this is performed in MATLAB, the signals are not strictly continuous. The sample rate is high enough that the difference is not important to understand a real-world continuous analogue. The carrier frequency `fc` and the low pass filter bandwidth `w` were chosen such that the carrier frequency is much higher than the bandwidth, but the exact values are not important.

```
% number of data points to transmit
samples = 1E3;
% spacing between points
period = 50;
% transmit pulse amplitude
V = 1;
% transmit frequency, about 5 cycles per data point
fc = pi/5;
% low pass filter width, significantly lower than transmit frequency
```

```
w = fc / 50;

% generate random +-1 values
tx_data = sign(randn(1, samples));

% insert zeros so data occurs every period points
data_upsample = upsample(tx_data, period);

% fill the zeros with the same data, and amplify by a factor of V
pulse = V*ones(period, 1);
pulse_data = conv(data_upsample, pulse, 'same');
% the 'same' argument ensures the correct array length
% (without the extra points on the front/back ends)

% Modulate to carrier frequency
cosine = cos(fc*(1:period*samples));
tx = pulse_data .* cosine;

% simulate transmission through a lossless but noisy channel
% the noise is a standard gaussian (mu=1, sigma=1)
rx = tx + randn(size(tx));

% demodulate and apply a low pass filter
demod = rx .* cosine;
lpf = sinc(w*(-period/2:period/2));
rx_lpf = conv(demod, lpf, 'same');

% normalize to amplitude V
rx_lpf = V*rx_lpf / max(rx_lpf);

% sample at intervals to recover original data
rx_data = sign(downsample(rx_lpf, period, period/2));

% estimate errors, then find the bit error rate (BER)
errs = rx_data ~= tx_data;
BER = sum(errs) / samples;
fprintf(Error Rate:
```

The result of this script is an error of approximately 27.3%. The waveforms in Figure 8.4 show how the noise tampers with the signal.

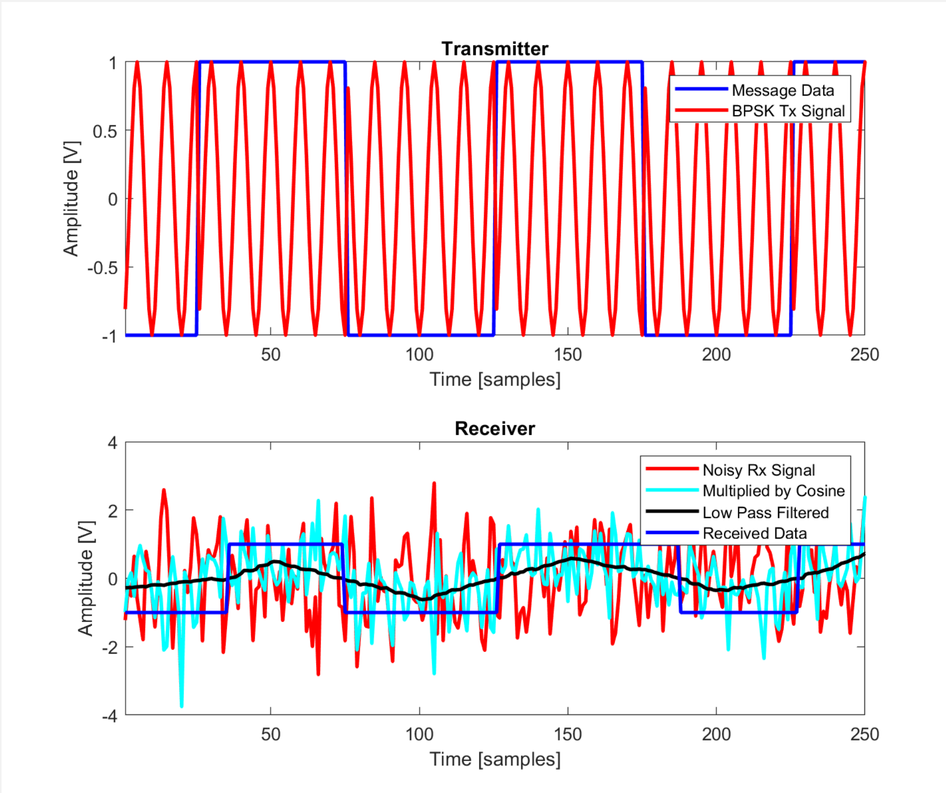


Figure 8.4: Waveforms in the transmitter (top) and receiver (bottom) of a binary phase shift keying system with additive noise from a standard Gaussian variable.

Figure 8.4 was generated with the following code:

```
subplot(211);
plot(inds, pulse_data, 'b', ...
    inds, tx, 'r', ...
    'LineWidth', 2)
legend('Message Data', 'BPSK Tx Signal')
xlim([1, period*5])
xlabel(Time [samples])
ylabel(Amplitude [V])
title(Transmitter)

subplot(212);
plot(inds, rx, 'r', ...
    inds, demod, 'c', ...
    inds, rx_lpf(inds), 'k', ...
    inds, sign(rx_lpf(inds)), 'b', ...
    'LineWidth', 2)
legend('Noisy Rx Signal', ...
    'Multiplied by Cosine', ...
```

```
    'Low Pass Filtered', ...
    'Received Data')
xlim([1, period*5])
xlabel(Time [samples])
ylabel(Amplitude [V])
title(Receiver)
```

8.2 *Introducing Noise to Pulse Amplitude Modulation (PAM) Systems*

Consider the system as shown in Figure 8.5, in which a DT signal $m[k]$ combined with a rectangular pulse, $p(t)$ is communicated through a noisy channel.

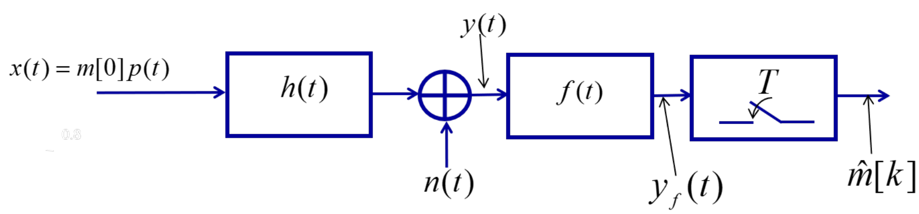


Figure 8.5: A PAM system corrupted by noise and transmitting only a single symbol.

Recall the probability of bit-error calculation for binary data $b[k]$, transmitted signal level V, noise power σ^2, and estimated bit $\hat{b}[k]$.

$$P\{b \neq \hat{b}\} = Q\left(\frac{V-\mu}{\sigma}\right) \tag{8.2}$$

since $\mu = 0$, we are left with $Q\left(\frac{V}{\sigma}\right)$.

To model noise levels in a PAM system, write the output $\hat{m}[k]$ of a PAM system in terms of a signal component $\hat{m}_s[k]$ and noise component $\hat{m}_n[k]$ assuming zero-ISI.

$$\hat{m}[k] = \hat{m}_s[k] + \hat{m}_n[k] \tag{8.3}$$

For noise power, it is sufficient to analyze a single symbol because PAM systems, assuming no ISI, are typically designed with equal probability of error for all symbols. The probability of error for one symbol (or any given symbol) depends on how much signal and how much noise is in $\hat{m}[k]$. To determine the ratio of signal to noise, begin by decomposing the filtered received signal $y_f(t)$ into signal and noise parts

$$y_f(t) = s_f(t) + n_f(t) \tag{8.4}$$

$$s_f(t) = (h * f * x)(t) \tag{8.5}$$

$$n_f(t) = (f * n)(t) \tag{8.6}$$

Assume zero-ISI and consider a single pulse at $t = 0$ to get

$$\hat{m}[0] = \hat{m}_s[0] + \hat{m}_n[0] \tag{8.7}$$

$$\hat{m}_s[0] = s_f(0) \qquad \text{(signal component)} \tag{8.8}$$

$$\hat{m}_n[0] = n_f(0) \qquad \text{(noise component)} \tag{8.9}$$

To find the probability of error, we are interested in the ratio of the average power of the signal component to that of the noise component. Recalling that $s_f(0)$ and $n_f(0)$ are random variables, denote their average powers as $\mathcal{E}_S$ and $\mathcal{E}_N$, respectively. The probability of error is in terms of the signal and noise levels, analogous to V and σ in Equation 8.2, which are the square root of their respective powers

$$P_{err} = Q\left(\sqrt{\frac{\mathcal{E}_S}{\mathcal{E}_N}}\right) \tag{8.10}$$

If $n(t)$ is a Gaussian noise process with PSD $\frac{N_0}{2}$ then

$$\hat{m}_n[0] \sim \mathcal{N}(0, \sigma_N^2) \tag{8.11}$$

The variance of the noise, σ_N^2, is required to find $\mathcal{E}_N$. Using statistical tools from Chapter 5, write the noise power in terms of the filter $f(t)$ and the random variable .

$$\mathcal{E}_N = \sigma_N^2 \tag{8.12}$$

$$= E[(\hat{m}_n[0])^2] \tag{8.13}$$

$$= E[(n_f(0))^2] \tag{8.14}$$

$$= R_{N_f}(0) \qquad \text{Autocorrelation of Filtered Noise} \tag{8.15}$$

$$= \int_{-\infty}^{\infty} S_{N_f N_f}(f) df \tag{8.16}$$

$$= \frac{N_0}{2} \int_{-\infty}^{\infty} |F(f)|^2 df \qquad \text{PSD of Filtered Noise} \tag{8.17}$$

$$= \frac{N_0}{2} \int_{-\infty}^{\infty} |f(\tau)|^2 d\tau \qquad \text{Parseval's Theorem} \tag{8.18}$$

Observe that the receive filter impacts the noise power but **not** the transmit pulse or the channel. Next, perform a similar analysis on the signal level, which is the square root of signal power $\mathcal{E}_S$.

$$\sqrt{\mathcal{E}_S} = \hat{m}_s[0] \tag{8.19}$$

$$= s_f(0) \tag{8.20}$$

$$= m[0](p * h * f)(0) \tag{8.21}$$

$$= m[0](f * g)(0) \qquad \text{Substitute } g(t) = (p * h)(t) \tag{8.22}$$

$$= m[0] \int_{-\infty}^{\infty} f(\tau) g(0 - \tau) d\tau \tag{8.23}$$

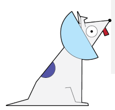
$$= m[0] \int_{-\infty}^{\infty} f(\tau) g(-\tau) d\tau \tag{8.24}$$

Unlike noise power, the receive filter, $f(t)$, transmit pulse, $p(t)$, and channel, $h(t)$, all impact the signal level. Also, the integral is not data-dependent, while $m[0]$ is a data-dependent value. Set $m[0] = \pm V$, i.e., 2-PAM, 1 bit/symbol. To summarize, the two analyses produce equations for average noise **power** and an equation for average signal **level**.

$$\mathcal{E}_N = \frac{N_0}{2} \int_{-\infty}^{\infty} |f(\tau)|^2 d\tau \tag{8.25}$$

$$\sqrt{\mathcal{E}_S} = V \int_{-\infty}^{\infty} f(\tau) g(-\tau) d\tau \tag{8.26}$$

Fun Facts from Doggo



Signal and noise power, $\mathcal{E}_S$ and $\mathcal{E}_N$, refer to the average received signal/noise energy per symbol. The signal or noise level is found by taking the square root of the signal or noise power. The power, in units of Watts, must also consider the symbol rate, however, but this is not required for the error analysis.

Recall that in Equation 8.2, the probability of error was found by applying the Q-function to the ratio of bits transmitted at a signal level (V) to the noise level (σ). Applying the Q-function in this noisy 2-PAM system in the same way suggests the ratio $\frac{\sqrt{\mathcal{E}_S}}{\sqrt{\mathcal{E}_N}}$.

$$P_{err} = Q\left(\frac{\text{signal level}}{\text{noise level}}\right) \tag{8.27}$$

$$= Q\left(\frac{\sqrt{\mathcal{E}_S}}{\sqrt{\mathcal{E}_N}}\right) \tag{8.28}$$

$$= Q\left(\sqrt{\frac{\mathcal{E}_S}{\mathcal{E}_N}}\right) \tag{8.29}$$

The quantity $\frac{\mathcal{E}_S}{\mathcal{E}_N}$ is called the *signal-to-noise ratio (SNR)*. Using Equations 8.25 and 8.26, the probability of error for the 2-PAM system in terms of V, $\frac{N_0}{2}$, and the pulse/channel/filter frequency responses is

$$P_{err} = Q\left(\frac{\left|V \int_{-\infty}^{\infty} f(\tau) g(-\tau) d\tau\right|}{\sqrt{\frac{N_0}{2} \int_{-\infty}^{\infty} |f(\tau)|^2 d\tau}}\right) \tag{8.30}$$

Recall that the Q-function calculates the probability that a standard Gaussian exceeds a given value. The a larger the argument to the Q-function is, the less likely an error is to occur. In other words, to minimize the probability of error, maximize the contents of the Q-function, which is the SNR, by finding a receive filter $f(t)$ to maximize

$$\frac{\left|V \int_{-\infty}^{\infty} f(\tau) g(-\tau) d\tau\right|^2}{\frac{N_0}{2} \int_{-\infty}^{\infty} |f(\tau)|^2 d\tau} \tag{8.31}$$

It's useful to note that scaling our filter, $f(t)$, does not impact our result because we are squaring a square root. What matters is the shape of our filter $f(t)$.

8.3 The Cauchy-Schwarz Inequality

To determine the shape of our filter, $f(t)$, we can utilize the Cauchy-Schwarz Inequality, which is considered one of the most important inequalities in all of mathematics!

This section utilizes the Cauchy-Schwarz Inequality to derive the matched filter.

In two dimensions, the Cauchy-Schwarz Inequality states that the magnitude of the dot product of two vectors $\vec{x}$ and $\vec{y}$ is less than or equal to the product of their magnitudes.

$$|\vec{x} \cdot \vec{y}|^2 \leq |\vec{x}|^2 |\vec{y}|^2 \tag{8.32}$$

Equality is achieved if and only if $\vec{x}$ and $\vec{y}$ are in the same direction, which is when $\vec{x} = \alpha\vec{y}$. This places a fundamental limit on the length of dot products. It will soon be seen how this applies to the optimization problem from the previous chapter. Expanding the inequality into scalar form gives

$$(x_1 y_1 + x_2 y_2)^2 \leq (x_1^2 + x_2^2)(y_1^2 + y_2^2) \tag{8.33}$$

Observing the pattern in the subscripts, write in n-dimensions

$$\left(\sum_{i=1}^{n} x_i y_i \right)^2 \leq \sum_{i=1}^{n} x_i^2 \sum_{i=1}^{n} y_i^2 \tag{8.34}$$

There is an analogous equation for real functions that also holds when the integrals are finite. The vectors become real functions, and the summations become integrals to switch to a form of the inequality on continuous dimensions.

$$\left| \int_{-\infty}^{\infty} \phi_1(\tau)\phi_2(\tau) d\tau \right|^2 \leq \int_{-\infty}^{\infty} |\phi_1(\tau)|^2 \, d\tau \int_{-\infty}^{\infty} |\phi_2(\tau)|^2 \, d\tau \tag{8.35}$$

Like before, equality is achieved if and only if one function is a scaled version of the other:

$$\phi_1(t) = \alpha\phi_2(t) \tag{8.36}$$

Rearranging Equation 8.35, by dividing the right hand side by $\phi_1(\tau)$ gives us the Cauchy-Schwarz inequality

$$\frac{\left| \int_{-\infty}^{\infty} \phi_1(\tau)\phi_2(\tau) d\tau \right|^2}{\int_{-\infty}^{\infty} |\phi_1(\tau)|^2 \, d\tau} \leq \int_{-\infty}^{\infty} |\phi_2(\tau)|^2 \, d\tau \tag{8.37}$$

Does that equation look familiar? Recall that a filter $f(t)$ that maximizes Equation 8.31 is desired.

$$\frac{\left| V \int_{-\infty}^{\infty} f(\tau) g(-\tau) d\tau \right|^2}{\frac{N_0}{2} \int_{-\infty}^{\infty} |f(\tau)|^2 d\tau} \tag{8.38}$$

From the Cauchy-Schwarz Inequality, shown in Equation 8.37, along with the assumption that $g(\tau)$ is real, it can be shown that

$$\frac{\left|V\int_{-\infty}^{\infty} f(\tau)g(-\tau)d\tau\right|^2}{\frac{N_0}{2}\int_{-\infty}^{\infty}|f(\tau)|^2 d\tau} \leq \frac{V}{N_0/2}\int_{-\infty}^{\infty}|g(-\tau)|^2 d\tau \tag{8.39}$$

Thus, if the filter $f(t)$ meets this bound (i.e., it achieves equality as in Equation 8.36), then it must be the best choice! If $g(t)$ is real and $f(t) = \alpha g(-t)$, then it can be shown that this is the best choice for $f(t)$

$$\frac{\left|V\int_{-\infty}^{\infty} \alpha g(-\tau)g(-\tau)d\tau\right|^2}{\frac{N_0}{2}\int_{-\infty}^{\infty}|\alpha g(-\tau)|^2 d\tau} = \frac{V}{N_0/2}\int_{-\infty}^{\infty}|g(-\tau)|^2 d\tau \tag{8.40}$$

$$\frac{\alpha^2\left|V\int_{-\infty}^{\infty} |g(-\tau)|^2 d\tau\right|^2}{\alpha^2\frac{N_0}{2}\int_{-\infty}^{\infty}|g(-\tau)|^2 d\tau} = \frac{V}{N_0/2}\int_{-\infty}^{\infty}|g(-\tau)|^2 d\tau \tag{8.41}$$

$$\frac{V}{N_0/2}\int_{-\infty}^{\infty}|g(-\tau)|^2 d\tau = \frac{V}{N_0/2}\int_{-\infty}^{\infty}|g(-\tau)|^2 d\tau \tag{8.42}$$

Therefore, $f(t) = \alpha g(-t)$ minimizes the probability of error. This is known as the **matched filter**! If $g(t)$ is complex, the matched filter is instead the time-reversed complex conjugate of $g(t)$

$$f(t) = \alpha g^*(-t) \tag{8.43}$$

With the matched filter, the probability of error becomes

$$P_{err} = Q\left(\frac{\left|V\sqrt{\int_{-\infty}^{\infty}|g(\tau)|^2 d\tau}\right|}{\sqrt{\frac{N_0}{2}}}\right) \tag{8.44}$$

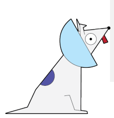
Fun Facts from Doggo

When a 2-PAM receiver uses the matched filter

$$f(t) = \alpha g^*(-t)$$

where α is a constant scale factor and $g(t) = p(t) * h(t)$, the probability of error is

$$P_{err} = Q\left(|V|\sqrt{\frac{2}{N_0}}\sqrt{\int_{-\infty}^{\infty}|g(\tau)|^2 d\tau}\right)$$



You may have noticed the scale factor α disappeared. This corresponds to the fact that the receiver filter will treat noise and signal equally, so a large gain factor will not improve the SNR. Practically, receiver gain brings the received signal to a range appropriate for the hardware. All of this analysis has produced method to design a pulse and filter that minimizes errors in communication systems.

8.4 *Practical Considerations of the Matched Filter*

Suppose that a system is described by the frequency response in Figure 8.6a. The matched filter is 8.6b).

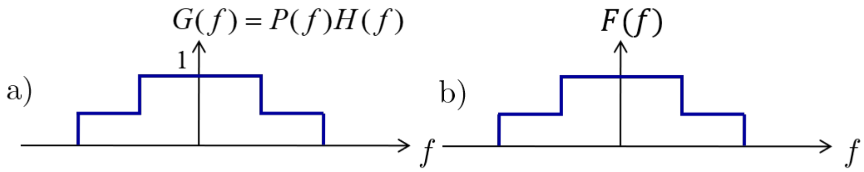


Figure 8.6: A PAM system with assumed zero-ISI. The combined frequency response (left) and the matched filter (right) are the same shape.

To minimize the symbol period T with zero-ISI, the effective channel $R(f) = P(f)H(f)F(f)$ is ideally a box. Making $P(f)H(f)$ and $F(f)$ boxes is a usual choice because it is easy to find the smallest zero-ISI symbol period when working with these boxes. Using boxes requires accurate synchronization for symbol samples, however. The raised cosine can be used instead.

To minimize the probability of error (assuming $\beta = 1$, $g(t)$ and $G(f)$ are real and symmetric), the following should be true

$$F(f) = G(f) = P(f)H(f) \tag{8.45}$$

Using the matched filter and to have $R(f)$ be a raised cosine, the following must also be a raised cosine:

$$\begin{aligned} R(f) &= P(f)H(f)F(f) && (8.46) \\ &= P(f)H(f)P(f)H(f) && (8.47) \\ &= (P(f)H(f))^2 && (8.48) \end{aligned}$$

This means that $P(f)H(f)$ and $F(f)$ need to be the square-root of a raised cosine. This is sometimes called a root raised cosine filter. In many wired communications systems, $H(f)$ is well-modeled as an ideal low pass filter. $R(f)$ is usually still going to be a raised cosine to reduce sensitivity to timing error. In such cases, make $P(f)$ a root-raised-cosine that fits in the pass-band of $H(f)$. The matched filter then becomes the root-raised-cosine filter again.

Chapter 9: M-PAM and System Design

Consider the M-PAM system shown in Figure 9.1. Thus far, noisy versions of systems capable of transmitting one of four different values have been considered. Expressions for the probability of error for these systems required use of the the $Q(\cdot)$ function.

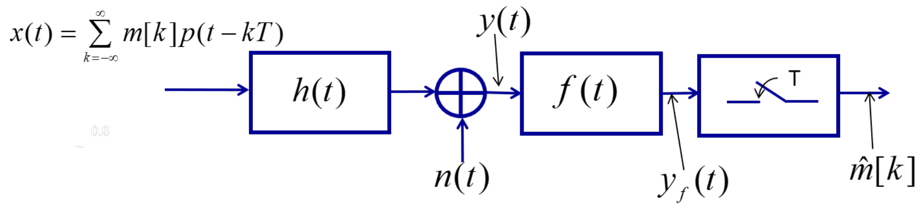


Figure 9.1: An M-PAM system with noise

Here, a system where the random variable X can take one of M different values, evenly spaced in the range [-V, V] is explored. Suppose that you observe Y, which is a noisy version of X, where

$$Y = X + N$$

and $N \sim \mathcal{N}(0, \sigma^2)$.

Let Figure 9.2 illustrate the different possible values of X, which can take the values illustrated by the dots. Estimates of X, denoted by $\hat{X}$ are made by comparing Y to different thresholds. Y is used to estimate X by choosing the possible value nearest to Y. The thresholds where the estimates of X change from one level to another are shown using the dashed lines in Figure 9.2.

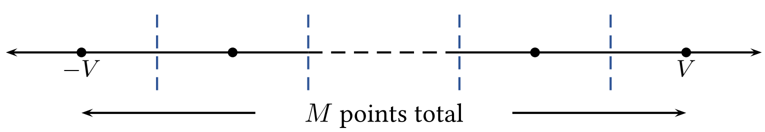


Figure 9.2: The region from $-V$ to $+V$ is divided over M different points where the estimates for X are the nearest points. Thresholds are marked with dashed lines and are halfway between points.

To show that the probability of error for this system is

$$P\{\text{Error}\} = \frac{2M-2}{M} Q\left(\frac{V}{\sigma(M-1)}\right)$$

Consider that the transmitted values are spaced by $\frac{2V}{M-1}$, in which case the spacing between a transmitted value and the nearest threshold is $\frac{V}{M-1}$ as illustrated in the following figure.

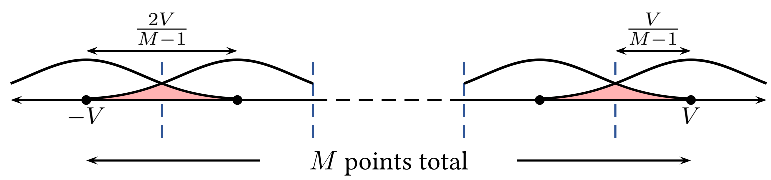


Figure 9.3: The spacing between points is $\frac{2V}{M-1}$, so each point is $\frac{V}{M-1}$ away from the nearest threshold. The shaded regions show where errors occur.

The values at the edge (i.e. the values of $\pm V$) have one half the probability of error compared to the values in the middle. The probability of error given that V or $-V$ is transmitted corresponds to one of the shaded areas in Figure 9.3, which has the following value

$$\begin{aligned} P\left\{\text{Error}|X=V\right\} &= P\left\{\text{Error}|X=-V\right\} \\ &= Q\left(\frac{V}{\sigma(M-1)}\right) \end{aligned}$$

The probability of error given one of the middle points was transmitted is twice that of the edges

$$P\left\{\text{Error}|X=\text{ middle point}\right\} = 2Q\left(\frac{V}{\sigma(M-1)}\right)$$

The total probability of error is then

$$\begin{aligned} P\left\{\text{Error}\right\} = &\; P\left\{\text{Error}|X=V\right\}P\{X=\pm V\} \\ &+ (M-2)P\left\{\text{Error}|X=\text{ a middle point}\right\}P\{X=\text{ a middle point }\} \end{aligned}$$

where we have used the fact that there are $M-2$ middle points with identical probabilities of error. Similarly, there are two edge points. Since the probability of transmitting any one point is $\frac{1}{M}$, the total probability of error is finally

$$\begin{aligned} P\left\{\text{Error}\right\} &= \frac{1}{M}Q\left(\frac{V}{\sigma(M-1)}\right) + \frac{1}{M}Q\left(\frac{V}{\sigma(M-1)}\right) + \frac{(M-2)}{M}2Q\left(\frac{V}{\sigma(M-1)}\right) \\ &= \frac{2}{M}Q\left(\frac{V}{\sigma(M-1)}\right) + \frac{(2M-4)}{M}Q\left(\frac{V}{\sigma(M-1)}\right) \\ &= \frac{2M-2}{M}Q\left(\frac{V}{\sigma(M-1)}\right) \end{aligned}$$

9.1 *Designing with M-PAM*

Consider the system above in which a DT signal $m[k]$ using a rectangular pulse $p(t)$ is communicated. Assume that $n(t)$ is a white Gaussian noise process with PSD $\frac{N_0}{2}$ and that the zero-ISI condition is met. $m[k]$ takes one of M values distributed evenly in $[-V,V]$. With this general case for M and $g(t)=(p*h)(t)$, the probability of error is

$$P_{err} = \left(\frac{2M-2}{M}\right)Q\left(\frac{\left|V\int_{-\infty}^{\infty}f(\tau)g(-\tau)d\tau\right|}{(M-1)\sqrt{\frac{N_0}{2}\int_{-\infty}^{\infty}|f(\tau)|^2d\tau}}\right) \tag{9.1}$$

With the matched filter $f(t) = \beta g(-t)$ where β is a scale factor, the probability of error is

$$P_{err} = \left(\frac{2M-2}{M}\right) Q\left(\frac{|V|\sqrt{2}}{(M-1)\sqrt{N_0}}\sqrt{\int_{-\infty}^{\infty}|g(\tau)|^2 d\tau}\right) \tag{9.2}$$

$$= \left(\frac{2M-2}{M}\right) Q\left(\frac{|V|\sqrt{2}}{(M-1)\sqrt{N_0}}\sqrt{\int_{-\infty}^{\infty}|G(f)|^2 df}\right) \tag{9.3}$$

Recall the main design parameters for a digital communication system (besides cost, system complexity, etc.)

- Transmit Power
$$P = \frac{1}{T}E[|m[k]|^2]\int_{-\infty}^{\infty}|p(t)|^2 dt \tag{9.4}$$
- Bandwidth - controls maximum symbol rate for zero-ISI and impacts noise variance
- Probability of Error - application dependent (wireless $\sim 10^{-4}$, wired $\sim 10^{-7}$
- Data rate - usually maximized, $\frac{\log_2 M}{T}$ bits/second

Typically, a designer must find the largest value for M that meets all the constraints then determine the average transmit power. Given Constraints: Bandwidth $\leq B$, $P_{err} \leq \epsilon$, and Transmit Power $\leq P_{lim}$, here are steps to maximize data rate:

1. Find $P(f)$ to make $G(f) = P(f)H(f)$ a box from $[-B, B]$
2. Use the matched filter, $f(t) = \frac{1}{\beta}g(-t)$. The probability of error is then
$$P_{err} = \left(\frac{2M-2}{M}\right) Q\left(\frac{|V|\sqrt{2}}{(M-1)\sqrt{N_0}}\sqrt{\int_{-\infty}^{\infty}|g(\tau)|^2 d\tau}\right) \tag{9.5}$$
$$= \left(\frac{2M-2}{M}\right) Q\left(\frac{|V|\sqrt{2}}{(M-1)\sqrt{N_0}}\sqrt{\int_{-\infty}^{\infty}|G(f)|^2 df}\right) \tag{9.6}$$
3. Find the largest value of M such that the power constraint is met, and
$$\left(\frac{2M-2}{M}\right) Q\left(\frac{|V|\sqrt{2}}{(M-1)\sqrt{N_0}}\sqrt{\int_{-\infty}^{\infty}|g(\tau)|^2 d\tau}\right) \leq \epsilon \tag{9.7}$$

Transmit Power

Transmit power is typically the power output by an amplifier into an antenna. The transmit power for M-PAM is covered here and can be used for the specific 2-PAM case in BPSK. For any natural number M, Figure 9.4 illustrates the possible values of $m[k]$ in an M-PAM system.

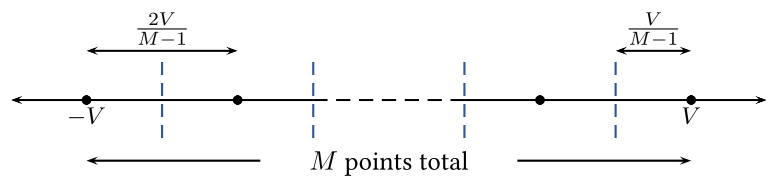


Figure 9.4: The voltage range from $[-V, V]$ is divided into M parts in an M-PAM system.

The energy of a unit pulse with unit impedance/resistance can be evaluated in either frequency domain or time domain, whichever is easier.

$$\int_{-\infty}^{\infty} |P(f)|^2 df = \int_{-\infty}^{\infty} |p(\tau)|^2 d\tau \tag{9.8}$$

The energy of a pulse directly to the right of $-V$ requires the following energy (also into a unit impedance/resistance). This is the unit pulse energy scaled by the height of the example pulse.

$$(-V + \frac{2V}{M-1})^2 \int_{-\infty}^{\infty} |P(f)|^2 df = (-V + \frac{2V}{M-1})^2 \int_{-\infty}^{\infty} |p(\tau)|^2 d\tau \tag{9.9}$$

It follows that the average energy of transmitting into a unit impedance/resistance for all M pulse levels is found with an arithmetic mean of all possible pulse heights.

$$\frac{1}{M}\sum_{l=0}^{M-1}\left(-V + l\frac{2V}{M-1}\right)^2 \int_{-\infty}^{\infty} |P(f)|^2 df = \frac{V^2}{M}\sum_{l=0}^{M-1}\left(\frac{2l}{M-1} - 1\right)^2 \int_{-\infty}^{\infty} |P(f)|^2 df \tag{9.10}$$

The average power of transmitting into a unit impedance/resistance depends on the symbol rate, the number of pulse sent per second, to have units of Watts. If a greater number of pulses per second with some energy are transmitted, then it makes sense for the power to increase.

$$P_{avg} = \frac{V^2}{MT}\sum_{l=0}^{M-1}\left(\frac{2l}{M-1} - 1\right)^2 \int_{-\infty}^{\infty} |P(f)|^2 df \tag{9.11}$$

Antennae are rarely only 1Ω (50Ω is a typical value), so we should also find the transmit power if we vary the antenna impedance. If the transmit signal is a voltage going into an impedance/resistance of R, the average power is

$$P_{avg} = \frac{V^2}{MTR}\sum_{l=0}^{M-1}\left(\frac{2l}{M-1} - 1\right)^2 \int_{-\infty}^{\infty} |P(f)|^2 df \tag{9.12}$$

In most standard applications, we want to use the largest V possible by setting the transmit power equal to the limit

$$\frac{V^2}{MTR}\sum_{l=0}^{M-1}\left(\frac{2l}{M-1} - 1\right)^2 \int_{-\infty}^{\infty} |P(f)|^2 df = P_{lim} \tag{9.13}$$

and solving for V. The largest V possible is then

$$V = \sqrt{\frac{P_{lim} \cdot RTM}{\sum_{l=0}^{M-1}\left(\frac{2l}{M-1} - 1\right)^2 \int_{-\infty}^{\infty} |P(f)|^2 df}} \tag{9.14}$$

We now have a way to find transmit power, so we update the design parameters:

- Bandwidth - controls maximum symbol rate for zero-ISI, impacts noise power
  Ideally, $T = \frac{1}{2B}$ if signal must live in $[-B, B]$
- Probability of Error - application dependent (wireless $\sim 10^{-4}$, wired $\sim 10^{-7}$)

$$P_{err} = \left(\frac{2M-2}{M}\right) Q\left(\frac{|V|\sqrt{2}}{(M-1)\sqrt{N_0}}\sqrt{\int_{-\infty}^{\infty}|g(\tau)|^2 d\tau}\right) \tag{9.15}$$

- Transmit Power - Constraint is met when

$$V = \sqrt{\frac{P_{lim} \cdot RTM}{\sum_{l=0}^{M-1}\left(\frac{2l}{M-1}-1\right)^2 \int_{-\infty}^{\infty}|P(f)|^2 df}} \tag{9.16}$$

- Data Rate - Usually maximized, is $\frac{\log_2 M}{T}$ bits/second

We can evaluate all of this by increasing M until these constraints fail, then take the largest passing value of M.

PAM Digital Processing

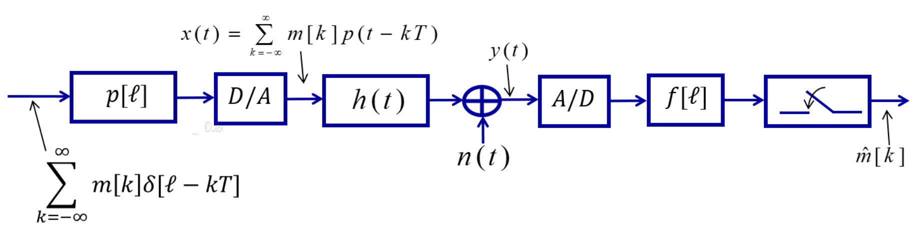


Figure 9.5: An M-PAM system with digital processing on both the transmitter and receiver.

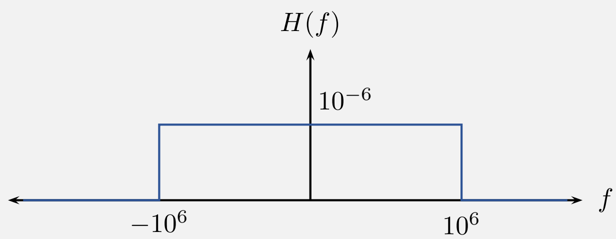
Much of the processing for PAM transmissions is done digitally. The process of design and analysis of the system is similar to the previous section, except that there is a digital to analog converter (D/A; DAC) and an analog to digital converter (A/D; ADC) at the two ends of the link that are interfacing with the channel.

9.2 *Example Design Problem*

Example 9.1

Suppose you have a system with the following parameters:

- Transmit Power Limit $P_{lim} = 0.18$ W, signal is a voltage.
- Impedance (resistance) of transmit system (e.g. antenna, cable) is 50 Ω
- Error probability less than or equal to 10^{-4}
- Noise power spectral density is 2×10^{-20}
- Channel frequency response:



Design a pulse, a filter, and M to maximize the data rate subject to the above constraints.

Steps:

1. Pick a pulse shape $P(f)$ so that $P(f)H(f)$ is the widest box possible/practical
2. Pick $F(f)$ to be a box of the same width. Height does not matter, so make it 1.
3. Find T for zero-ISI by inspecting the effective channel $R(f) = P(f)H(f)F(f)$
4. Calculate $\int_{-\infty}^{\infty} |P(f)|^2 df$, $\int_{-\infty}^{\infty} |F(f)|^2 df$, and $\int_{-\infty}^{\infty} P(f)H(f)F(f)df$.
5. Loop through all possible M, starting with $M = 2$
   a) Calculate the largest possible V to meet power limit P_{lim} with resistance/impedance R
   $$V = \sqrt{\frac{P_{lim} \cdot RTM}{\sum_{l=0}^{M-1} \left(\frac{2l}{M-1} - 1\right)^2 \int_{-\infty}^{\infty} |P(f)|^2 df}} \tag{9.17}$$
   b) If the probability of error is above threshold, set $M = M - 1$ and break. Otherwise, continue the loop, $M = M + 1$

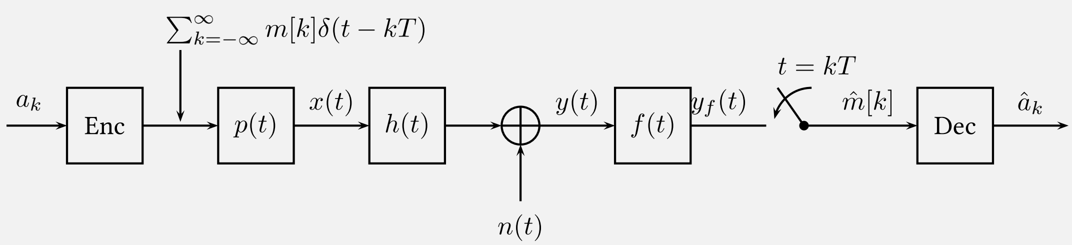
The solution for these parameters is $M = 6$. The work that shows this is left as an exercise.

9.3 *Problem Set 6: M-PAM Design and Root Raised Cosine*

Problem	Topic	Points
1	2PAM with Noise and Matched Filter	4
2	Designing M-PAM with Matched Filter	6
3	Simulating zero-ISI with Root Raised Cosine	6
Total:		16

Exercise 9.1

(4 points) Consider the following block diagram of a Pulse-Amplitude-Modulation (PAM) system which is used to communicate symbols a_k across a channel modeled as a Linear-Time-Invariant (LTI) filter of impulse response $h(t)$ with additive White-Gaussian-Noise of Power-Spectral-Density $\frac{N_o}{2}$ given by $n(t)$.



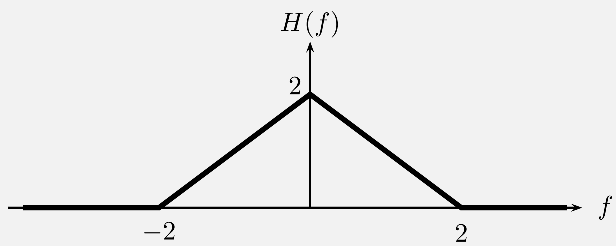
Additionally, assume that $a_k = 1$ or $a_k = 0$ with equal probability,

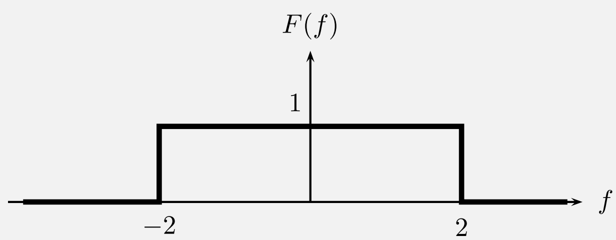
$$m[k] = \begin{cases} 1 & \text{if } a_k = 1 \\ -1 & \text{if } a_k = 0 \end{cases}$$

and

$$\hat{a} = \begin{cases} 0 & \text{if } \hat{m}[k] < 0 \\ 1 & \text{if } \hat{m}[k] \geq 0. \end{cases}$$

a) Suppose that $h(t)$ and $f(t)$ have the following Fourier Transforms:



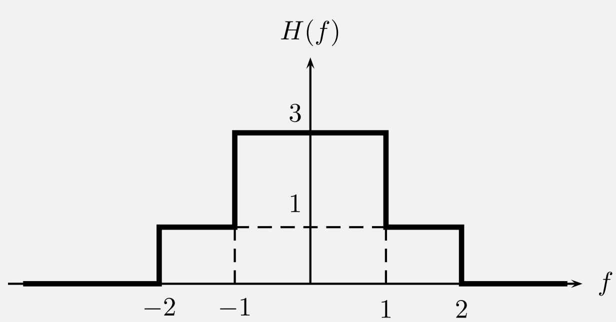


i. If $p(t) = \delta(t)$, what is the minimum symbol period T that ensures zero Inter-Symbol-Interference (ISI)?

ii. If $p(t) = \delta(t)$, what is the average probability of error of this system assuming no ISI?

   Hint: Consider the transmission of a single bit ($m[0]$) and write $y_f(0)$ as a combination of signal and noise parts.

b) Suppose that $h(t)$ has the following Fourier Transform:



and $f(t)$ is the matched filter.

i. Please write an expression for the Fourier Transform of $f(t)$ in terms of $H(f)$ and $P(f)$ (Fourier Transform of $p(t)$).

ii. Please find a combination of $p(t)$, $f(t)$ and T such that there is zero ISI in the system AND the symbol period T is the smallest possible. Briefly justify why you think this is the smallest possible T. Note that $f(t)$ is required to be the matched-filter.

iii. Please find the probability of error of the system with the parameters you found in (b) ii.

Exercise 9.2

(6 points) Consider a Pulse-Amplitude-Modulation (PAM) communications system as we saw in class and illustrated in Figure 9.6.

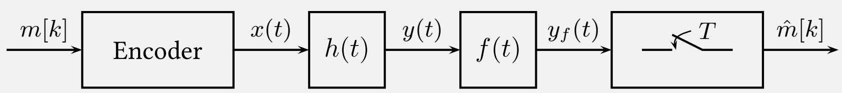


Figure 9.6: PAM System

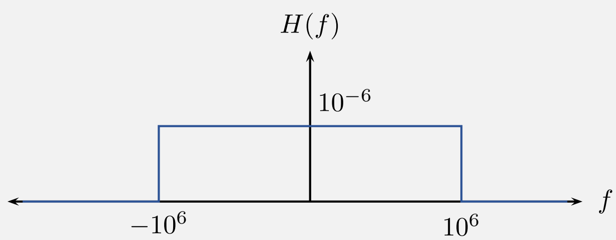


Figure 9.7: Channel frequency response $H(f)$ with width 2×10^6 and height 10^{-6}

Suppose that $m[k]$ can take one of M possible levels distributed uniformly in [-V, V]. Assume that a pulse $p(t)$ is used to communicate data in this channel, with symbol period T seconds. Suppose that the transmitted signal in this system is a voltage, which is transmitted into an impedance (resistance) of R. Assume that the transmitted signal $x(t)$ has zero frequency content in frequencies $|f| > B$.

a) What is the smallest symbol period possible for zero ISI?

b) What is the average transmit power of the system as a function of $p(t), R, T, M$, and V.

c) Find the V which makes the average transmit power equal to P_{lim}. Please express this as a function of $p(t), R, T, M$, and P_{lim}.

d) You will now optimize the data rate of a system subject to the following constraints

- The maximum transmitted power is $P_{lim} = 0.18W$.
- The maximum probability of error is 10^{-4}
- The channel is an ideal low-pass filter with cutoff frequency $B = 10^6$ Hz as shown in Figure 9.7
- $R = 50\Omega$
- Noise PSD, $\frac{N_0}{2} = 2 \times 10^{-20}$ W/Hz

i. What is the smallest value of T possible for zero ISI?

ii. Suppose that the system you designed has the smallest T with no ISI. What is the variance of the noise component in $\hat{m}[k]$ if the height of $F(f)$ = 1?

iii. Find the largest value of M that meets the criteria above. You will likely have to write a computer program to do this, recalling that M has to be an integer.

Exercise 9.3

(6 points) In this problem, you will simulate transmitting a PAM signal through a channel which has a timing error. Note that such an error is inevitable in real systems as transmitters and receivers have different sampling clocks.
Download the MATLAB function `TimingErrorChannel.m` and the script `ps6Helper.m` This file contains a function which simulates the effect of transmitting a signal through a channel with a timing offset (which is of a fractional number of samples), and additive noise.

a) Run the script `ps6Helper.m`. It creates an impulse response h, the impulse response of a raised-cosine, and the impulse response of a root-raised-cosine pulse. The latter two use a roll-off factor of 0.5. The target symbol rate (which controls where the zero crossings are of the pulses) is set by the parameter T. The code starts with $T = 5$. The code generates two plots. One with the channel frequency response, and the other with the channel frequency response as well as the Fourier transforms of the raised cosine and root raised cosine filters. Additionally, the code also generates a vector `dat` with $N = 100000$ random, independent, identically distributed (IID) ± 1 values.

b) By examining the frequency response of the channel, try to estimate the smallest value of T that will result in zero ISI. Estimate this based on the shape of the frequency response of the channel. Note that T must be an integer here.

c) By using the `upsample` command in MATLAB, generate a vector `x` which has `T` zeros between consecutive data values, where T is the value you found in the previous part. Normalize it by dividing it with its root-mean-square value, i.e. `x = x/rms(x);`

d) Simulate running `x` through the channel by running `y = TimingErrorChannel(x);`. `y` now contains the received signal after being run through the channel with a timing offset applied.

e) Note that the channel has inserted some unknown delay to your signal. You will need to find the best times to sample. The following segment of MATLAB code estimates the best time for the first sample of `y`. We are assuming here that the first 1000 samples of `x` are known by the receiver which can cross correlate the received signal with the first 1000 samples of `x`.

```
[Ryx lags] = xcorr(y, x(1:1000));
[mm, ii] = max(abs(Ryx));
StartIndex = lags(ii)+1;
```

The samples of `y` taken at the best times then are `y(StartIndex:T:end)`.

f) By comparing the sign of the entries of `y(StartIndex:T:end)` and the sign of the entries of `dat`, find the probability of error.

g) Create a new transmitted signal which uses the root-raised-cosine pulse, by convolving `x` with `rrc`. Call this vector `xrrc`. i.e. `xrrc = conv(x, rrc)`. Normalize this vector by its rms value: `xrrc = xrrc/rms(xrrc);`

h) Simulate transmitting this vector through the channel, and then convolve the resulting signal with `rrc` to simulate applying the matched filter at the receiver. Find the best sampling point and find the probability of error as you did in Parts e) to f). .

i) Repeat Parts c) to h) until you find the smallest value of T such that the system with the root-raised cosine filter has a Probability of Error less than or equal to 10^{-4} and T is minimized.

Chapter 10: Generalized Modulation Techniques

10.1 *Generalizing Pulse Amplitude Modulation*

Instead of transmitting a pulse whose amplitude changes with the transmit data, why not transmit different signals (e.g., pulses) for different symbols? Suppose we have M different possible symbols (or $K = \log_2 M$ bits). We would transmit one choice from the M different pulses

$$p_1(t), p_2(t), ..., p_M(t) \tag{10.1}$$

to represent each of the M possible symbols. For PAM, each of these would be the same pulse scaled by different values. In any case, our encoder becomes the mapper shown below to turn bits into pulses.

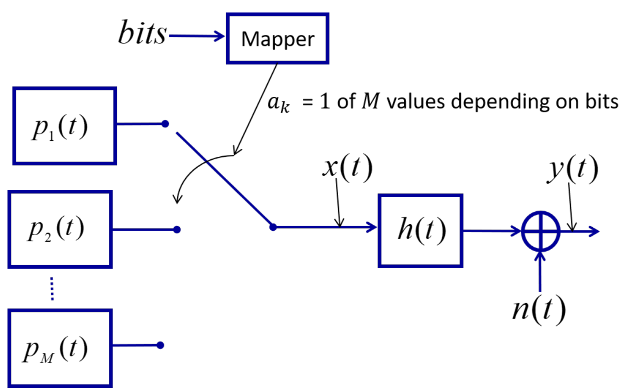


Figure 10.1: A mapper turns bits into different pulses by connecting to one of the pulse generators at the right time for the current bit group.

The mapper groups bits into symbols of $\log_2 M$ bits each. The M different possibilities for each symbol means each can represent multiple bits. With this generalization, we can consider numerous new modulation techniques in familiar terms.

Modulation Examples

Pulse-Position Modulation (PPM)

In Pulse Position Modulation (PPM), the position of pulses within a window of width W encodes information. We can detect symbols by matching a filter to the pulse and calculating the peak location in a cross-correlation. Clearly, if we can precisely detect the position of a pulse in the window, then we can have a higher number of possible positions and therefore a higher bit-rate. Figure 10.2 shows a set of pulses for PPM with a symbol period of W and dividing the period into four pulse positions.

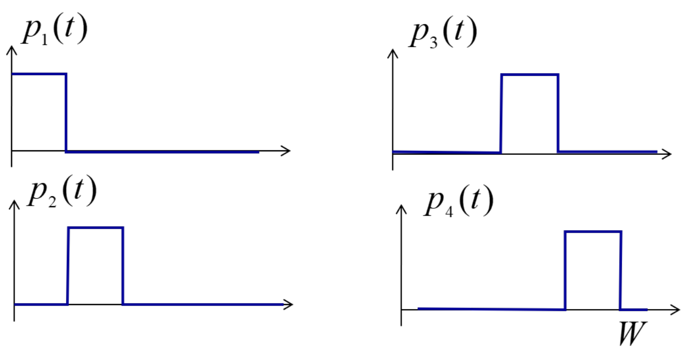


Figure 10.2: Waveforms for Pulse-Position Modulation place a box somewhere in the width of the pulse.

In PPM applications, synchronizing the reading frame is a necessary step and is typically accomplished by sending a known sequence of bits at the beginning of the transmission.

On-Off Keying (OOK)

In On-Off Keying (OOK), the existence of a pulse encodes a 1. Non-existence encodes a 0. To detect, we look for a pulse. If there is a peak in the output after the matched filter, 1, otherwise, 0. Figure 10.3 shows the on and off pulses for on-off keying.

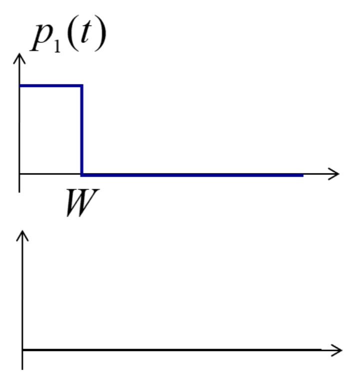


Figure 10.3: Waveforms for On-Off keying.

This method has the weakness of not distinguishing transmission of all zeros from a lack of transmission. It is implemented in optical communication systems by toggling a laser to communicate digital information.

10.2 Implementation of General Modulation Techniques

Different modulation techniques have similarities and differences. Generally, some kind of matched filtering is followed by sampling and a final decision. The exact form depends on the transmit scheme:

- PAM - matched filter with "unit" pulse

- OOK - matched filter with pulse
- PPM - matched filter with pulse

We will generalize our analysis of the system. First, we define the transmission of a symbols more generally with a convolution of the pulse through a channel to form a received pulse.

$$\begin{aligned} q_1(t) &= p_1(t) * h(t) \\ q_2(t) &= p_2(t) * h(t) \\ &\vdots \\ q_M(t) &= p_M(t) * h(t) \end{aligned} \tag{10.2}$$

We can design $p_1(t), p(2), \cdots, p_M(t)$ such that the received pulses $q_i(t)$ are all orthogonal to each other, e.g., the Walsh functions in Figure 10.4. Choosing orthogonal pulses is almost always the right decision because of the receiver implementation. These and other orthogonal functions enable Code-Division Multiple Access (CDMA) where multiple messages can be transmitted simultaneously using orthogonal functions.

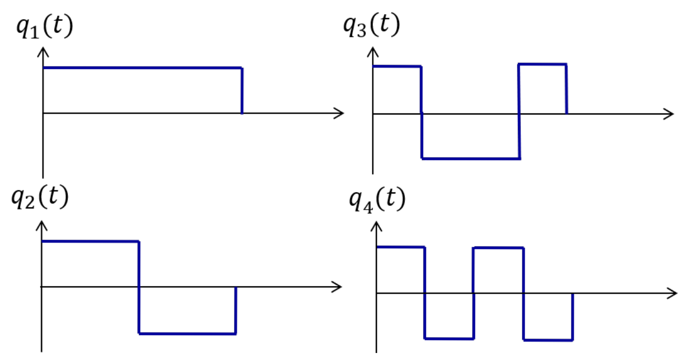


Figure 10.4: The Walsh functions are an example of orthogonal functions

In other words, orthogonality means the convolution with other functions in the set is zero at $t = 0$

$$(\overleftarrow{q_i} * q_j)(t=0) = \int_{-\infty}^{\infty} \overleftarrow{q_i}(-\tau) q_j(\tau) d\tau = \int_{-\infty}^{\infty} q_i(\tau) q_j(\tau) d\tau = \begin{cases} 0, & i \neq j \\ A, & i = j \end{cases} \tag{10.3}$$

for $A \neq 0$, real $p(t)$, and real $h(t)$. This means that we can identify which symbol we received by calculating Equation 10.3 for every possible received signal and picking the result which is most correlated to one of our symbols. Due to the effect of the channel, the others may not be perfectly zero. We also must ensure that the sampling time is synchronized because Equation 10.3 does not necessarily hold when $t \neq 0$ or even when $t \approx 0$. With this strategy, we feed the received signals into a bank of M filters, each a matched filter to a received symbol q_i, as in Figure 10.5, that identifies which symbol was most likely transmitted through the channel.

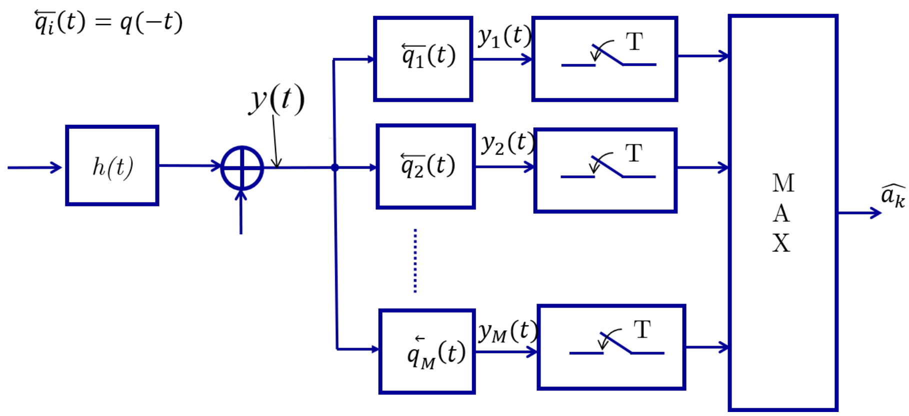


Figure 10.5: A bank of matched filters, applied to an orthogonal set of symbol, can be used to identify which symbols was transmitted. Synchronization is still critical for the orthogonality condition to work properly.

Mathematically, what is the output of a matched filter $q_j(t)$ when the transmitted symbol is $p_i(t)$?

$$y_j(t) = \overleftarrow{q}_j * (q_i(t) + n(t)) \tag{10.4}$$

Assuming zero-ISI and considering the representative zeroth symbol, we find the output

$$\begin{aligned}
y_j(0) &= \overleftarrow{q}_j * (q_i(0) + n(0)) && (10.5)\\
&= \int_{-\infty}^{\infty} q_i(\tau)\overleftarrow{q}_j(0-\tau)d\tau + \int_{-\infty}^{\infty} n(\tau)\overleftarrow{q}_j(0-\tau)d\tau && (10.6)\\
&= \int_{-\infty}^{\infty} q_i(\tau)\overleftarrow{q}_j(-\tau)d\tau + \int_{-\infty}^{\infty} n(\tau)\overleftarrow{q}_j(-\tau)d\tau && (10.7)\\
&= \int_{-\infty}^{\infty} q_i(\tau)q_j(\tau)d\tau + \int_{-\infty}^{\infty} n(\tau)q_j(\tau)d\tau && (10.8)
\end{aligned}$$

With Equation 10.3, we have an equation with signal/noise components.

$$y_j(0) = \begin{cases} 0, & i \neq j \\ A, & i = j \end{cases} + N \tag{10.9}$$

This allows us to analyze the probability of error and other parameters using a similar approach to what we did for PAM. We can determine the signal-to-noise ratio and evaluate the probability that a symbol is confused for another from there.

10.3 Carrier Transmission

Our analysis of PAM signals and the generalization in Section 10.1 considered only baseband signals (occupy frequencies around zero). The transmitted signal (for PAM) is a pulse train

$$x_m(t) = \sum_{k=-\infty}^{\infty} m[k]p(t-kT) \tag{10.10}$$

However, many channels are bandpass because of the ability of different EM wavelengths to propagate, so we cannot transmit in the baseband. From our work with AM radio, we know how to move our transmissions to the pass-band using a cosine at the carrier frequency f_c

$$x(t) = x_m(t)\cos(2\pi f_c t + \theta_c) \tag{10.11}$$

At the receiver, we know to multiply by the same cosine and apply an LPF to get the message signal back. After demodulating, we can continue our analysis as normal!

Binary Phase Shift Keying (BPSK)

BPSK is defined by the following modulation scheme for the transmitted signal $x(t)$

$$x_m(t) = \sum_{k=-\infty}^{\infty} m[k]p(t-kT) \tag{10.12}$$

$$x(t) = x_m(t)\cos(2\pi f_c t + \theta_c) \tag{10.13}$$

where $m[k] = \pm 1$ for a symbol period T. We may recognize that this is the familiar Binary/2-PAM with carrier modulation, so why call it BPSK? We need only look at the transmitted signal waveform in Figure 10.6 for an answer.

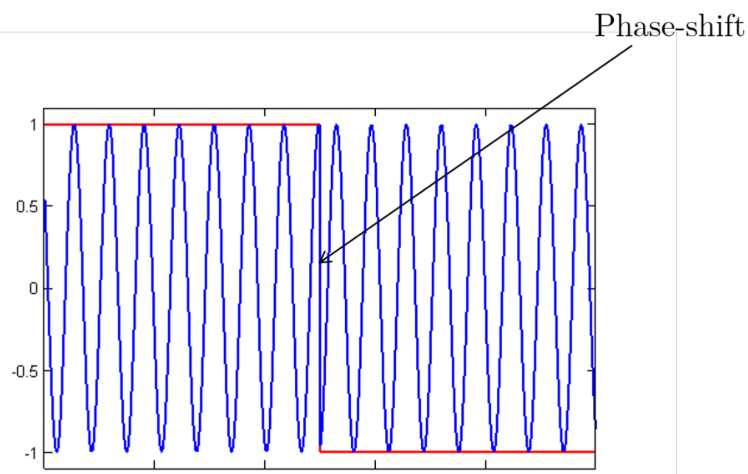


Figure 10.6: In Binary Phase-Shift Keying, sudden phase shifts in the modulated signal encode the data.

Chapter 11: Quadrature Amplitude Modulation

In this chapter, we are going to seemingly perform magic by turning a real-valued channel (a regular old waveform in time) into a complex-valued one! The magical outcome of this is that we essentially can double the bit rate of our system by pushing both real and imaginary values through simultaneously. We call this quadrature amplitude modulation (QAM), and it relies on the orthogonality of sines and cosines.

One way we can use the complex channel we gain in QAM is to send two individual PAM streams that do not interfere with each other. The result is a 2-dimensional (real and imaginary axes) constellation of received values. The simplest form of this is to implement BPSK on the real (I) and imaginary (Q) channels for a 4-QAM system (two 2-PAM systems).

11.1 Quadrature Channels

Consider a system where we transmit two independent baseband signals, one modulated by cosine and the other by sine. We will assume that we have an ideal channel $h(t) = \delta(t)$ and that the signals are band-limited to $[-f_M, f_M]$. The carrier frequency must also be much larger than the bandwidth, $f_M << f_c$. The transmitter for this 4-QAM system is shown in Figure 11.1.

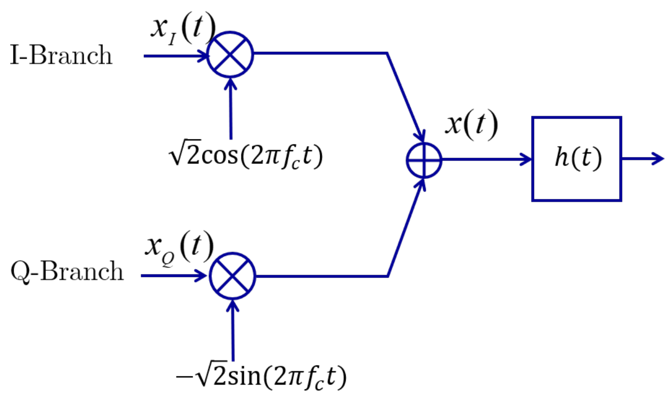


Figure 11.1: A 4-QAM transmitter block diagram. The transmitted signal combines two message signals modulated by a cosine and sine of the same frequency (effectively the same signal 90° out of phase, to which is what the word quadrature refers).

The receiver for this QAM system is shown in Figure 11.2 below.

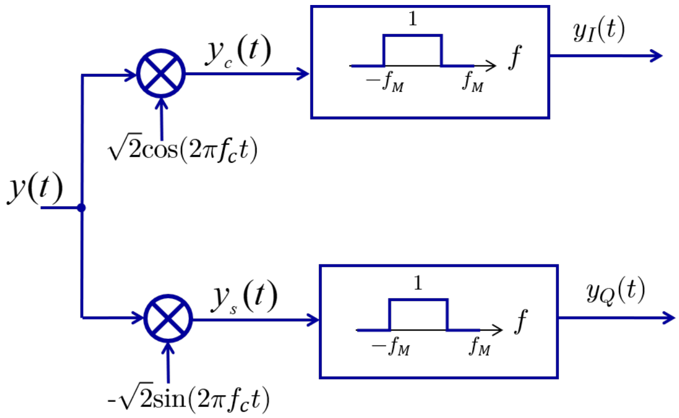


Figure 11.2: A 4-QAM receiver block diagram. The received signal separates two message signals by demodulating one by a sine and the other by a cosine at the same frequency.

Assuming synchronized transmitter and receiver, we have

$$y_I(t) = x_I(t) \tag{11.1}$$
$$y_Q(t) = x_Q(t) \tag{11.2}$$

11.2 *Baseband Equivalent Channel*

When we perform quadrature modulation like above, the entire signal chain from $x_I(t)$ and $x_J(t)$ to $y_I(t)$ and $y_Q(t)$ can be bundled into one complex valued process in the baseband, shown in Figure 11.3. Modulating by orthogonal cosine/sines, thereby permitting simultaneous real- and imaginary-valued signals, is the defining feature of this signal chain. The inputs and outputs are defined accordingly,

$$x_b(t) = x_I(t) + x_Q(t) \tag{11.3}$$
$$y_b(t) = y_I(t) + x_Q(t) \tag{11.4}$$

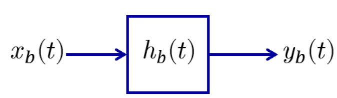


Figure 11.3: The baseband equivalent model. The input $x_b(t)$ and output y_b are complex signals, and the hidden intermediate impulse responses are complex as well.

If signals are sampled at a rate greater than the Nyquist frequency ($\geq 2(f_c + f_M)$, since the maximum frequency will be slightly above the carrier frequency), we have the DT system in Figure 11.4.

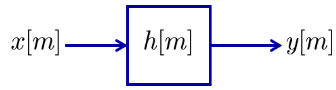


Figure 11.4: The baseband equivalent model as a DT system when the Nyquist sampling criteria is met for the maximum frequency present in the signal. All inputs, outputs, and impulse responses remain complex.

Deriving the Baseband Equivalent Model

Even if $h(t)$ is not a unit impulse, we can create this simple model of the system which enables us to apply the analysis techniques from Chapter 7 on Pulse Amplitude Modulation. Consider the frequency response of a channel $H(f)$, which is defined for all frequencies. Our transmitted signal only lives in the ranges $[-f_c - f_M, -f_c + f_M]$ and $[f_c - f_M, f_c + f_M]$, so we only care about $H(f)$ on those intervals. Thus, we define $H_{bp}(f)$ (not to be confused with $H_b(f)$) with the following equation and depict an example in Figure 11.5.

$$H_{bp}(f) = \begin{cases} H(f), & f_c - f_M < f \leq f_c + f_M \\ H(f), & -f_c - f_M < f \leq -f_c + f_M \\ 0, & \text{otherwise} \end{cases} \tag{11.5}$$

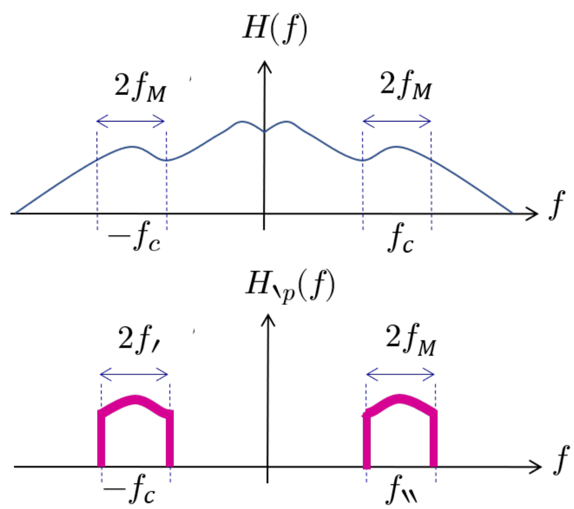


Figure 11.5: When our modulated signals only live in a certain range, we can model the channel frequency response as only the part that lives in that range.

When we make this approximation of $H(f)$, the change has no impact on the transmitted data and we can write

$$Y(f) = X(f)H(f) = X(f)H_{bp}(f) \tag{11.6}$$

Our mathematical representation of the baseband equivalent model relies only on this subset of $H(f)$. Recall our transmitter from before, except we will now use Equation 11.3 and the equivalent channel response given in Figure 11.6.

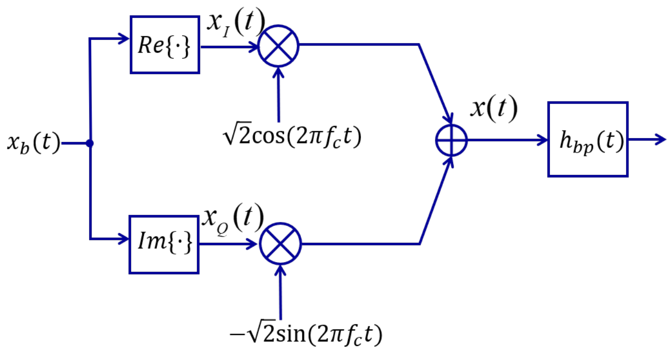


Figure 11.6: The QAM transmitter with an equivalent channel response and a single complex-valued input.

We notice that our model sums together a real cosine part and an imaginary sine part at the same frequency and with the same absolute magnitude. With Euler's formula, $e^{ix} = \cos x + j \sin x$, we can write

$$\Re\{\sqrt{2}x_b(t)e^{j2\pi f_c t}\} = \Re\{\sqrt{2}(x_I(t) + jx_Q(t))(\cos 2\pi f_c t + j\sin 2\pi f_c t)\} \tag{11.7}$$

$$\Re\{\sqrt{2}x_b(t)e^{j2\pi f_c t}\} = \sqrt{2}x_I(t)\cos 2\pi f_c t - \sqrt{2}x_Q(t)\sin 2\pi f_c t = x(t) \tag{11.8}$$

Writing sines and cosines with a complex exponential simplifies the arithmetic and our system, as seen in Figure 11.2.

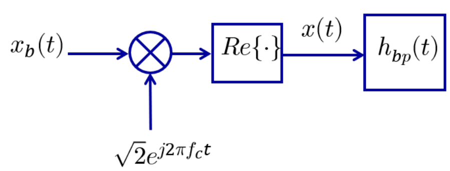
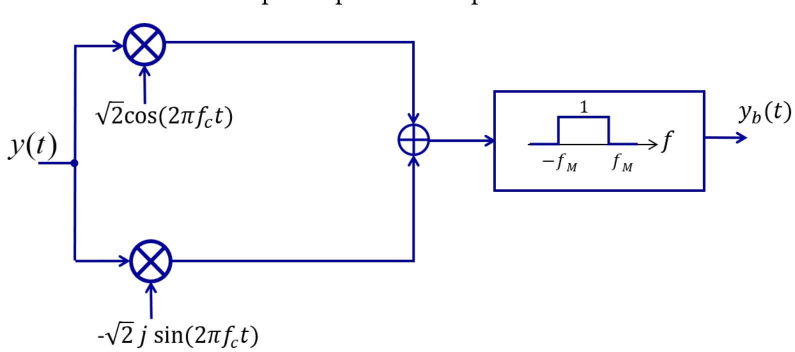


Figure 11.7: The QAM transmitter in terms of a single multiplication with a complex exponential given in Eq. 11.8.

The receiver when written with complex inputs and outputs becomes



Invoking linearity, it does not matter if the low-pass filtering comes before or after the sum. Therefore, we can then move the two LPF stages to a single (complex) LPF after the sum, as shown in Figure 11.8.

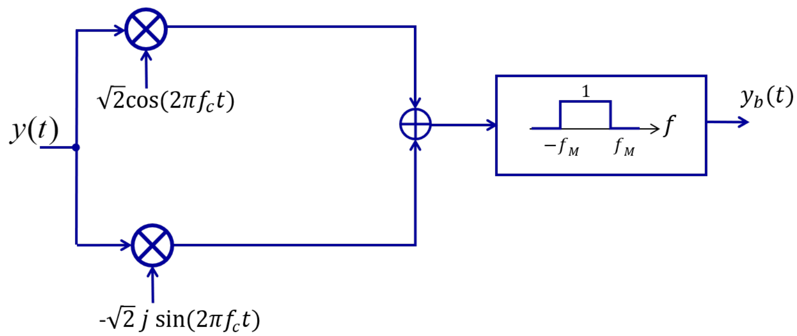


Figure 11.8: The QAM receiver with a complex LPF and single complex output.

Remembering what we did in the transmitter, we write

$$\sqrt{2}\cos 2\pi f_c t - \sqrt{2}\sin 2\pi f_c t = \sqrt{2}e^{-j2\pi f_c t} \tag{11.9}$$

With Equation 11.9, the receiver can be simplified to a single complex multiplication and LPF, shown in Figure 11.9.

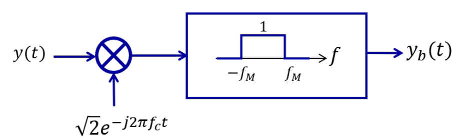


Figure 11.9: The QAM receiver can be written in terms of a single multiplication with a complex exponential and a complex low-pass filter.

Combining the transmitter in Figure 11.2 and receiver in Figure 11.9 into one system gives the system in Figure 11.10.

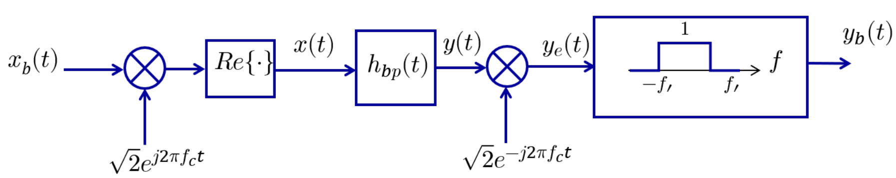


Figure 11.10: The combined Tx/Rx baseband equivalent system written using complex signals and operations.

Next, we will derive an equation for $y_e(t)$. From the system diagram, we can write $y_e(t)$

$$y_e(t) = (x * h_{bp}(t))\,\sqrt{2}e^{-j2\pi f_c t} \tag{11.10}$$

We expand the convolution operator into its integral form then move the complex exponential into the integrand.

$$y_e(t) = \int_{-\infty}^{\infty} x(\tau) h_{bp}(t-\tau) d\tau \sqrt{2} e^{-j2\pi f_c t} \tag{11.11}$$

$$= \sqrt{2} \int_{-\infty}^{\infty} x(\tau) h_{bp}(t-\tau) e^{-j2\pi f_c t} d\tau \tag{11.12}$$

We can make the following substitution

$$e^{-j2\pi f_c t} = e^{-j2\pi f_c (t-\tau)} e^{-j2\pi f_c \tau} \tag{11.13}$$

into Equation 11.12 to yield

$$y_e(t) = \sqrt{2} \int_{-\infty}^{\infty} x(\tau) e^{-j2\pi f_c \tau} h_{bp}(t-\tau) e^{-j2\pi f_c (t-\tau)} d\tau \tag{11.14}$$

We define the signals

$$h_b(t) = h_{bp}(t) e^{-j2\pi f_c t} \tag{11.15}$$

$$x_e(t) = x(t) e^{-j2\pi f_c t} \tag{11.16}$$

Replacing Equations 11.15 and 11.16 into Equation 11.14

$$y_e(t) = \sqrt{2} \int_{-\infty}^{\infty} x_e(t) h_b(t-\tau) d\tau \tag{11.17}$$

This integral is equivalent to the definition of convolution shown in Equation 2.50,

$$y_e(t) = \sqrt{2} x_e(t) * h_b(t) \tag{11.18}$$

This changes the channel $h_{bp}(t)$ before the demodulating multiplier to the baseband channel $h_b(t)$ **after** the multiplier.

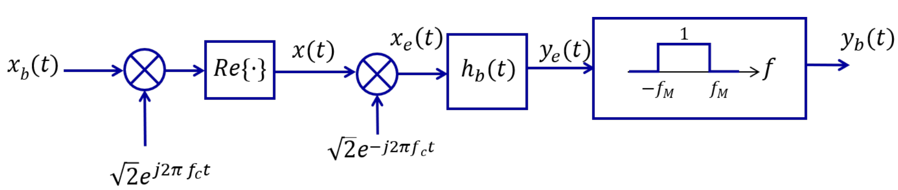


Figure 11.11: The base band system with derived channel h_b placed after the second multiplying stage.

Because the system is linear, we can switch the low-pass filter and the channel $h_b(t)$

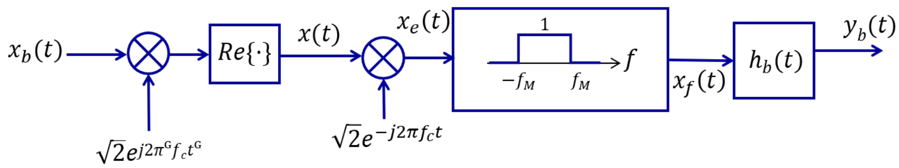


Figure 11.12: The base band system with derived channel h_b placed at the very end using the linearity of the LPF.

Why have we gone to all of this trouble to get the channel impulse response on the tail end? Because this allows us to apply the LPF to the signal independent of the channel. Given that we can write the time-domain signal $x_e(t)$ in terms of separate and simple cosines/sines, we can identify which ones are attenuated and therefore removed. Beginning with the definition of $x_e(t)$, we convert to cosines and sines.

$$\begin{aligned} x_e(t) &= x(t)e^{-j2\pi f_c t} \\ &= (2x_I(t)\cos(2\pi f_c t) - 2x_Q(t)\sin(2\pi f_c t))(\cos(2\pi f_c t) - j\sin(2\pi f_c t)) \end{aligned}$$

We then expand the product and apply a product-to-sum identity on the mixed sine/cosine products.

$$\begin{aligned} x_e(t) &= 2x_I(t)\cos^2(2\pi f_c t) - jx_I(t)\cos(2\pi f_c t)\sin(2\pi f_c t) \\ &\quad - x_Q(t)\cos(2\pi f_c t)\sin(2\pi f_c t) + 2jx_Q(t)\sin^2(2\pi f_c t) \\ &= 2x_I(t)\cos^2(2\pi f_c t) - jx_I(t)\sin(2\pi 2 f_c t) \\ &\quad - x_Q(t)\sin(2\pi 2 f_c t) + 2jx_Q(t)\sin^2(2\pi f_c t) \end{aligned}$$

We know that the squared cosine and sine terms put the signals $2x_I(t)$ and $2x_Q(t)$ in the baseband with half amplitude and at $2f_c$ either from our frequency analysis of AM or from applying product-to-sum identities. Either can be shown to justify the resulting unity gain on $x_I(t)$ and $x_Q(t)$

$$\begin{aligned} x_e(t) &= x_I(t) + x_I(t)\cos(2\pi 2 f_c t) - jx_I(t)\sin(2\pi 2 f_c t) \\ &\quad - x_Q(t)\sin(2\pi 2 f_c t) + jx_Q(t) + jx_Q(t)\sin(2\pi 2 f_c t) \end{aligned}$$

Now, we can apply the LPF to each term, which removes those for which $f > f_M$, remembering that $f_c >> f_M$.

$$\begin{aligned} x_f(t) &= x_I(t) + x_I(t)\cos(2\pi 2 f_c t) - jx_I(t)\sin(2\pi 2 f_c t) \\ &\quad - x_Q(t)\sin(2\pi 2 f_c t) + jx_Q(t) + jx_Q(t)\sin(2\pi 2 f_c t) \\ &= x_I(t) + jx_Q(t) \\ &= x_b(t) \end{aligned}$$

Since we have shown that $x_f(t) = x_b(t)$, we can justify the simplification of the system to the baseband equivalent $h_b(t)$.

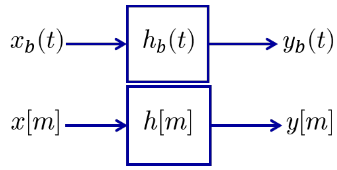


Figure 11.13: The baseband equivalent models (same as Figure 11.3 and Figure 11.4), which we have now proven are valid simplifications.

In real world systems, no one really looks at the in-phase (I) and quadrature (Q) components separately. Instead they are considered jointly as the baseboard equivalent model and are encoded and decoded together.

11.3 *Quadrature Phase Shift Keying (QPSK)*

In the previous section, we showed that a QAM system can be simplified to a complex system with a complex input $x_b(t)$, complex baseband equivalent channel $h_b(t)$, and complex output $y_b(t)$. This is a complex channel that we will now use for our QAM analysis. We can model noise as the addition of a complex Gaussian, add a complex receiver filter $f(t)$, and sample at a symbol period T to create our complex PAM system.

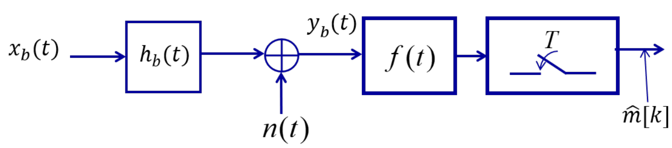


Figure 11.14: A Quadrature Amplitude Modulation system with additive noise and a receiver filter.

Suppose that we transmit the complex binary message signal

$$m[k] = \pm V \pm jV \tag{11.19}$$

$$x_b(t) = \sum_{k=-\infty}^{\infty} m[k]p(t - kT) \tag{11.20}$$

This is called 4-QAM, or Quadrature Phase Shift Keying since it works the same as BPSK but with a quadrature modulation scheme. The signal **constellation** is a figure representing all possible values of each symbol. A Tx and Rx constellation for QPSK is shown in Figure 11.15.

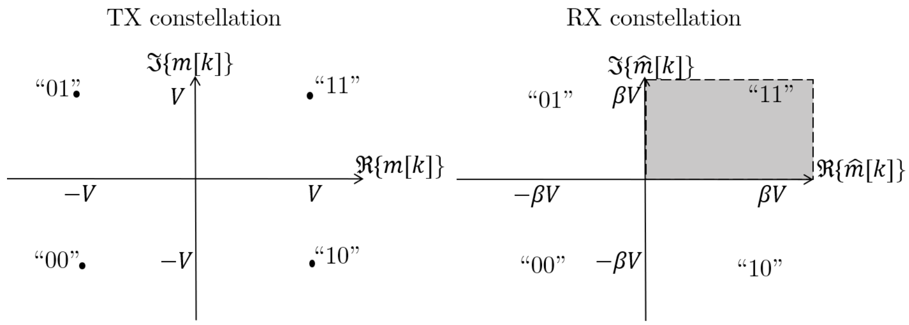


Figure 11.15: A QPSK constellation at the transmitter (left) and receiver (right).

In QPSK model, the receiver can decode using at the sign of the real and imaginary parts of the received, sampled signal as depicted in the constellation diagram, e.g., both parts positive corresponds to the bit group "11". We can also consider the probability of error much like we did for 2-PAM. Assuming that $h_b(t)$ and $f(t)$ are real (for now, we will generalize for complex values later), and zero-ISI, we can consider a representative zeroth symbol:

$$\begin{aligned}\hat{m}[0] &= m[0](p * h_b * f)(0) \\ &= \pm V \int_{-\infty}^{\infty} g(\tau) f(\tau) d\tau \pm jV \int_{-\infty}^{\infty} g(\tau) f(\tau) d\tau + (n * f)(0)\end{aligned} \tag{11.21}$$

where $g(t) = (p * h_b)(t)$. If $f(t)$ is the matched filter, then it is the time-reversed complex conjugate

$$f(t) = g^*(-t) \tag{11.22}$$

Plugging the matched filter into Equation 11.21

$$\begin{aligned}\hat{m}[0] &= \pm V \int_{-\infty}^{\infty} g(\tau) g^*(-\tau) d\tau \pm jV \int_{-\infty}^{\infty} g(\tau) g^*(-\tau) d\tau + (n * f)(0) \\ &= \pm V \int_{-\infty}^{\infty} |g(\tau)|^2 d\tau \pm jV \int_{-\infty}^{\infty} |g(\tau)|^2 d\tau + (n * f)(0)\end{aligned} \tag{11.23}$$

If we let $\beta = \int_{-\infty}^{\infty} |g(\tau)|^2 d\tau$, then Equation 11.23 is

$$\hat{m}[0] = \pm \beta V \pm j\beta V + (n * f)(0) \tag{11.24}$$

The noise $n(t)$ is complex, and we will assume that the real and imaginary part are independent, white Gaussian noise processes with PSD $\frac{N_0}{2}$.

$$\sigma_{n,I}^2 = \text{var}\{\Re\{(n * f)(0)\}\} = \beta \frac{N_0}{2} \tag{11.25}$$

$$\sigma_{n,Q}^2 = \text{var}\{\Im\{(n * f)(0)\}\} = \beta \frac{N_0}{2} \tag{11.26}$$

The probability of error on the I-channel should be the same as that on the Q-channel since both have level βV and are subject to noise with variance $\beta \frac{N_0}{2}$ (level $\sqrt{\beta \frac{N_0}{2}}$). The resulting error probability on either channel is written in terms of the Q-function

$$\begin{aligned} P\{\text{error on I-channel}\} &= P\{\text{error on Q-channel}\} \\ &= Q\left(\frac{\beta V}{\sqrt{\beta \frac{N_0}{2}}}\right) \\ &= Q\left(\frac{\sqrt{2\beta}V}{\sqrt{N_0}}\right) \end{aligned} \tag{11.27}$$

The probability that the whole 4-QAM symbol is **correct** is the probability that **both** I- and Q-channels are correct.

$$\begin{aligned} P\{\text{Correct for 4-QAM}\} &= P\{\text{Correct for Q- channel}\} \cdot P\{\text{Correct for Q- channel}\} \\ &= (P\{\text{Correct for I- channel}\})^2 \end{aligned} \tag{11.28}$$

We do not know the probability of correctness, only for error, but these should sum to one.

$$P\{\text{Correct for Q- channel}\} = 1 - P\{\text{Error for I- channel}\} \tag{11.29}$$

We then can write 11.28 as

$$P\{\text{Correct for 4-QAM}\} = (1 - P\{\text{Error for I- channel}\})^2 \tag{11.30}$$

$$= (1 - Q\left(\frac{\sqrt{2\beta}V}{\sqrt{N_0}}\right))^2 \tag{11.31}$$

The overall probability of error for 4-QAM is then

$$\begin{aligned} P\{\text{Error for 4-QAM}\} &= 1 - (1 - P\{\text{Error for I- channel}\})^2 \\ &= (1 - Q\left(\frac{\sqrt{2\beta}V}{\sqrt{N_0}}\right))^2 \end{aligned} \tag{11.32}$$

An alternative expression can be derived from Equation 11.27 directly using an OR operation.

$$P\{\text{Error for 4-QAM}\} = Q\left(\frac{\sqrt{2\beta}V}{\sqrt{N_0}}\right)^2 - 2Q\left(\frac{\sqrt{2\beta}V}{\sqrt{N_0}}\right) \tag{11.33}$$

M-QAM

If we allow ourselves to have multiple levels on both the real and imaginary analysis, the probability of error analysis extends much like it did for PAM. In 16-QAM, for example, the symbol encodes 4 bits, and the system has the constellation shown in Figure 11.16.

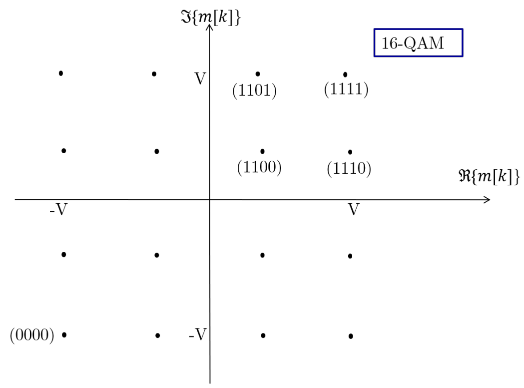


Figure 11.16: A 16-QAM constellation. Each sample encodes 4 bits for 16 possible symbols. A received sample is decoded to the nearest point in the constellation.

The error analysis for M-QAM considers the average of M possible signal levels, with noise remaining the same.

$$P\{\text{Correct for M-QAM}\} = (1 - P\{\text{Error for } \sqrt{M} - PAM\})^2 \tag{11.34}$$

$$P\{\text{Error for } \sqrt{M} - PAM\} = \frac{2\sqrt{M}-2}{\sqrt{M}} Q\left(\frac{\sqrt{\beta}V}{\sqrt{\frac{N_0}{2}(M-1)}}\right) \tag{11.35}$$

$$P\{\text{Error for M-QAM}\} = 1 - \left(1 - \frac{2\sqrt{M}-2}{\sqrt{M}} Q\left(\frac{\sqrt{\beta}V}{\sqrt{\frac{N_0}{2}(M-1)}}\right)\right)^2 \tag{11.36}$$

Part IV

Appendix

Appendix A: Reference Sheet

You may find the following information useful while working through problems in this book.

A.1 *Helpful Equations*

$$
\begin{aligned}
(a+b)^3 &= a^3 + 3a^2b + 3ab^2 + b^3 \\
(a+b)^4 &= a^4 + 4a^3b + 6a^2b^2 + 4ab^3 + b^4 \\
\sum_{k=0}^{\infty} a^k &= \frac{1}{1-a} \quad \text{if } |a| < 1 \\
\sin(\alpha \pm \beta) &= \sin\alpha\cos\beta \pm \cos\alpha\sin\beta \\
\cos(\alpha \pm \beta) &= \cos\alpha\cos\beta \mp \sin\alpha\sin\beta \\
\sin 2\theta &= 2\sin\theta\cos\theta \\
\cos 2\theta &= \cos^2\theta - \sin^2\theta \\
2\cos\theta\cos\varphi &= \cos(\theta-\varphi) + \cos(\theta+\varphi) \\
2\sin\theta\sin\varphi &= \cos(\theta-\varphi) - \cos(\theta+\varphi) \\
2\sin\theta\cos\varphi &= \sin(\theta+\varphi) + \sin(\theta-\varphi) \\
\sin(\theta + \tfrac{\pi}{2}) &= +\cos\theta \\
\cos(\theta + \tfrac{\pi}{2}) &= -\sin\theta \\
\cos^2(\theta) &= \frac{1}{2}(1 + \cos 2\theta) \\
\cos^3(\theta) &= \frac{1}{4}(3\cos\theta + \cos 3\theta) \\
\cos^4(\theta) &= \frac{1}{8}(3 + 4\cos 2\theta + \cos 4\theta) \\
e^x &= \sum_{n=0}^{\infty} \frac{x^n}{n!} = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \cdots
\end{aligned}
$$

Consider a stationary random process $X(t)$ that is transmitted through a Linear-Time-Invariant (LTI) system with impulse response $h(t)$. Suppose that the output of this system is the random process $Y(t)$. The following hold:

$$
\begin{aligned}
R_{YX}(\tau) &= R_{XX} * h(\tau) \\
R_{XY}(\tau) &= R_{XX} * \overleftarrow{h}(\tau) \\
R_{YY}(\tau) &= R_{XX} * h * \overleftarrow{h}(\tau)
\end{aligned}
$$

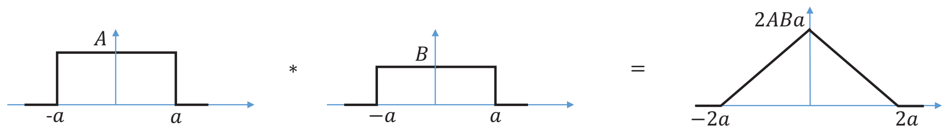
where $\overleftarrow{h}(\tau) = h(-\tau)$.

A.2 Fourier Transform Pairs and Properties

Time Domain Signal	Fourier Transform
$x(t)$	$X(f)$
$y(t)$	$Y(f)$
$\delta(t)$	1
$\delta(t-t_o)$	$e^{-j2\pi f t_o}$
$e^{j2\pi f_o t}$	$\delta(f-f_o)$
$\cos(2\pi f_o t)$	$\frac{1}{2}\delta(f-f_o)+\frac{1}{2}\delta(f+f_o)$
$\sin(2\pi f_o t)$	$\frac{1}{2j}\delta(f+f_o)-\frac{1}{2j}\delta(f-f_o)$
$ax(t)+by(t)$	$aX(f)+bY(f)$
$x(at)$	$\frac{1}{\lvert a\rvert}X\left(\frac{f}{a}\right)$
$x(t)y(t)$	$X(f)*Y(f).$
$x(t)*y(t)$	$X(f)Y(f)$
$e^{j2\pi f_o t}x(t)$	$X(f-f_o)$
$x(t-t_o)$	$e^{-j2\pi f t_o}X(f)$
$\frac{d}{dt}x(t)$	$j2\pi f X(f)$
$\frac{d^n}{dt^n}x(t)$	$(j2\pi f)^n X(f)$
$\frac{\sin(2\pi K t)}{\pi t}$	Box that equals 1 from $-K$ to K and zero elsewhere
Box that equals 1 from $-T$ to T and zero elsewhere	$\frac{\sin(2\pi T f)}{\pi f}$
$\sum_{k=-\infty}^{\infty}\delta(t-kT)$	$\frac{1}{T}\sum_{k=-\infty}^{\infty}\delta\left(f-\frac{k}{T}\right)$

A.3 Convolution

The convolution of two boxes results in a triangle as follows:



A.4 *Gaussian Random Variables and Q-function*

A Gaussian distributed random variable X with mean μ and variance σ^2 has the following Probability Density Function (PDF):

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

The Q-function (which does not have a closed-form) is defined as follows:

$$Q(t) = \int_t^\infty \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}\, dx$$

For your reference, here are some values for the Q-function.

x	$Q(x)$	x	$Q(x)$
$\frac{1}{4}$	4.01×10^{-1}	1	1.59×10^{-1}
$\frac{1}{3}$	3.69×10^{-1}	$\frac{5}{4}$	1.06×10^{-1}
$\frac{1}{2}$	3.09×10^{-1}	$\frac{4}{3}$	9.12×10^{-2}
$\frac{2}{3}$	2.52×10^{-1}	$\frac{3}{2}$	6.68×10^{-2}
$\frac{3}{4}$	2.27×10^{-1}	$\frac{5}{3}$	4.78×10^{-2}

Appendix B: Logarithms and Decibels

Decibels (dBs) are useful when working with values that span multiple orders of magnitudes. Decibels can be used to express:

- Ratio of one value of a physical property to another. Eg: dBW is power measured relative to 1 Watt.
- A change in a value (+1dB or -3dB)

Mathematically, a decibel represents the logarithmic ratio of measured quantity to a reference quantity, i.e.

$$X\ dB \quad = \quad 10\ log_{10}\left(\frac{Quantity_{measured}}{Quantity_{referenced}}\right) \tag{B.1}$$

Sometimes, the decibel symbol (dB) is followed by a letter like dBm or dBW. These letters represent a physical reference for the decibel ratio like power for dBm, dBW and dBc or gain for dBi and dBd. This physical reference, like 1mW of power for dBm or gain of an isotropic antenna for dBi, represent the denominator values in Eq. (B.1). Table B.1 below expands lists some commonly used references.

Table B.1: Commonly used dB symbols and their meanings. Please note that gain is a unitless quantity, ergo dBi and dBd are inherently unitless.

Symbol	**References**
dBW	Ratio for which the reference is 1 Watt (W) of power dissipated by a 50 Ω load
dBm	Ratio for which the reference is 1 milliwatt (mW) of power dissipated by a 50 Ω load
dBc	Ratio for which the reference is the power of a carrier wave (may be in mW or W)
dBi	Ratio for which the reference is the gain of an isotropic antenna
dBd	Ratio for which the reference is the gain of a half-wave dipole antenna
dB-Hz	Ratio for which the reference bandwidth is 1 Hz
dB/K	Ratio relates gain to system noise temperature (in Kelvins)

Table B.2 shows some key values used while working with decibels.

Linear (X)	**Base 10 Logarithm**	$dB = 10\mathbf{log}_{10}$**(X)**
1/1000	10^{-3}	-30
1/100	10^{-2}	-20
1/10	10^{-1}	-10
1/2	$10^{-0.3}$	-3
1	10^{0}	0
2	$10^{0.3}$	3
3	$10^{0.5}$	5
5	$10^{0.7}$	7
10	10^{1}	10
20	$10^{1.3}$	13
100	10^{2}	20
1000	10^{3}	30

Table B.2: Linear to logarithmic conversion for decibels.

Decibel Arithmetic

The decibel system is based on logarithms, and follows the logarithm arithmetic rules. A linear quantity can be expressed in dBs by using equation 1, also shown below:

$$X\ dB \quad = \quad 10\ log_{10}\left(\frac{P_{measured}}{P_{reference}}\right)$$

To convert a ratio expressed in dBs into a linear scale, you may use:

$$\text{Power Ratio} \quad = \quad 10^{\frac{X\ dB}{10}} \tag{B.2}$$

Because decibels are logarithmic, multiplication of linear values becomes addition in logarithmic space and division becomes subtraction. Rules used in dB arithmetic are listed in the Table B.3 below.

Table B.3: Arithmetic rules for logarithms.

Rule	Formula
Log of 1	$log(1) = 0$
Log Reciprocal	$log(1/x) = -log(x)$
Product	$log(xy) = log(x) + log(y)$
Quotient	$log(x/y) = log(x) - log(y)$

Exercise B.1

Convert the following ratios presented in Table B.4 into decibels and vice-versa.

Table B.4: Linear to logarithmic conversion for decibels.

Ratio	Ratio factored in linear space	Ratio in logarithmic space	dB
20	2×10	$3 + 10$	13
25	$100/(2 \times 2)$	$20 - (3 + 3)$	14
500			
0.4			
			-3
			-13
			-30
20 W			(in dBm)
50 mW			(in dBm)
50 mW			(in dBW)

HINT: To convert a ratio in decibel, Eq. (B.2) can be used. Alternatively, the steps presented below can be followed to convert from linear space to decibel space by using the table:

1. Express linear ratio in terms of common factors. Eg: $20 = 2 \times 10$.
2. Redefine the linear factors Using the rules of logarithm. Eg: $20 = 2 \times 10 = 3 + 10$.

3. Perform arithmetic in logarithm space to arrive at your answer. Eg: $20 = 2 \times 10 = 3 + 10 = 13\,dB$.

Appendix C: Software Defined Radio (SDR): Frequency Modulation (FM) Lab

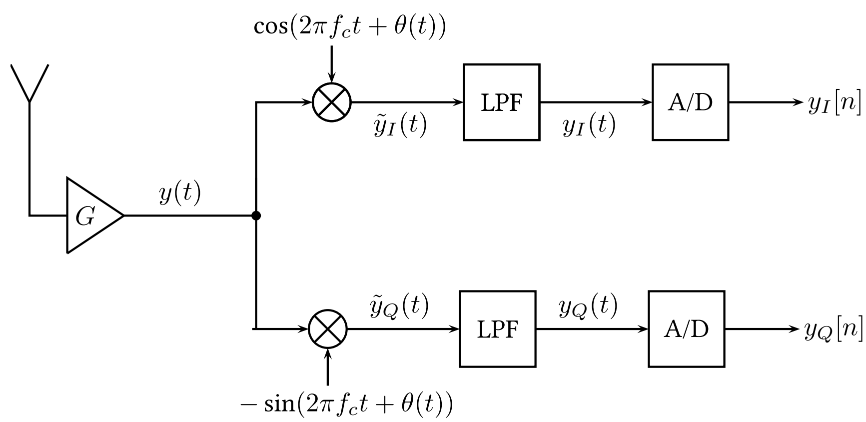
C.1 Introduction

In this assignment, you will use a low-cost software-radio peripheral to decode an FM modulated signal. A software-radio peripheral is a device that receives an analog radio signal and converts it into a digital signal (typically at baseband), to be processed by a computer. You will be using the RTL-SDR which is a receive-only software defined radio peripheral. In other words, this device can be used to receive but not transmit signals.

Please answer all parts labeled **Exercise** in your lab reports. Include all plots, MATLAB code, and a sample of your audio data.

C.2 RTL-SDR Receiver

A functional block diagram for the RTL-SDR system is shown in the following figure.



A gain, typically large, is applied to the signal from an external antenna. You can specify this gain in the tool you use to interface with the RTL-SDR. The resulting signal $y(t)$ is split into two branches. On one branch, $y(t)$ is multiplied by $\cos(2\pi f_c t + \theta(t))$, where $\theta(t)$ is used to model a small, possibly time-varying phase offset between the cosines at the transmitter and the receiver. The resulting signal, $\tilde{y}_I(t) = y(t)\cos(2\pi f_c t + \theta(t))$ is then low pass filtered through a filter whose bandwidth you can specify. That signal is then converted into a digital signal $y_I[n]$ using some sampling rate (sampling frequency) which you can also specify. You should ensure that the bandwidth of the filter is less than $\frac{1}{2}$ the sampling frequency to avoid aliasing. The signal obtained from this branch is called the in-phase component of $y(t)$.

On the second branch, the signal is multiplied by $-\sin(2\pi f_c t + \theta(t))$, resulting in $\tilde{y}_Q(t) = -y(t)\sin(2\pi f_c t + \theta(t))$. $\tilde{y}_Q(t)$ is filtered through a low-pass filter to produce $y_Q(t)$ which is then sampled to produce a digital signal $y_Q[n]$. This signal obtained from this branch is called the quadrature-phase component of $y(t)$.

It turns out that doing things in this manner (i.e. multiplying by cosine and -sine) will simplify a lot of the notation and the math when we are dealing with digital communications. As you saw in problem sets 1 and 2, multiplying by both of these carriers allows us to recover radio signals even when the transmitter and receiver are not synchronized. If the carriers are synchronized, we can instead send two independent data streams on each carrier simultaneously.

$y_I[n]$ and $y_Q[n]$ are the *only* signals you directly obtain from the RTL-SDR.

C.3 Installing RTL-SDR tools

There are several different tools to interface with the RTL-SDR. The most convenient is to use MATLAB. You can install the tools following the instructions on this Google Doc. We have also provided starter code that captures 3 million samples from the RTL-SDR at a particular frequency and puts the data into two vectors `y_I` and `y_Q`. You can change the parameters of the system by modifying the starter code appropriately. The gain parameter may have to be changed depending on the strength of the signal for your desired station, where you collect the data. This number is specified in decibels (dB). If you'd like more practice with decibels check out Appendix B: Logarithms and Decibels.

C.4 FM Demodulation

Exercise 1: Envelope Detector Approach

We have seen how we can cascade a differentiator with an envelope detector (diode + RC circuit) to decode an FM signal using a cheaper circuit. We cannot do this directly with the RTL-SDR as it does not provide direct access to raw FM signals (i.e. it only gives you the $y_I[n]$ and $y_Q[n]$ signals). In other words, you cannot access the FM signal $y(t)$ directly. One work-around for this is to set f_c to a frequency slightly lower than the frequency of the station you would like to access, with a sufficiently high sample rate so that the signal for the targeted station is acquired. E.g. to decode the station at $f_t = 90.9$ MHz, you can set $f_c = 90.8$ MHz, and set the sampling rate to 300 000. Assume that $\theta(t) \approx 0$ for this exercise and neglect its effect in the analysis.

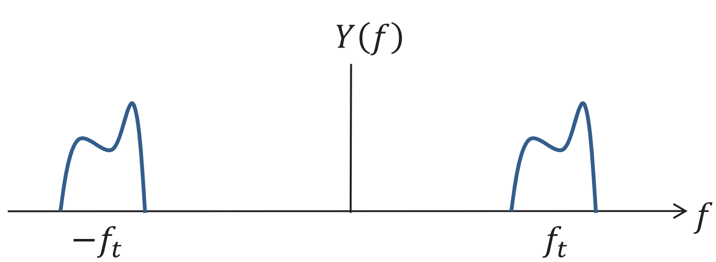


Figure C.1: Frequency domain illustration of a radio station at frequency f_t.

Parts (a) - (d) implement the work-around described above to acquire a received (but not demodulated) FM signal of the form

$$y_r(t) = G\cos\left(2\pi f t + \phi(t)\right)$$

where

$$\phi(t) = 2\pi k_0 \int_{-\infty}^{t} m(\tau)d\tau$$

Parts (e) and (f) demodulate the received FM signal using a derivative, ideal diode, and low pass filter. This envelope detector produces a signal $m(t)+C$ where C is some constant. Part (g) removes the constant term and adjusts the volume to make the received signal speaker friendly.

(a) Suppose that the signal you wish to decode is a station broadcasting at $f_t = 90.9$ MHz, and has a bandwidth[1] of 80kHz. In the frequency domain, this would correspond to two blobs at ± 90.9 MHz, as illustrated in Figure D.1. If you set $f_c = 90.8$ MHz, and assuming $\theta(t)$ is negligible, please sketch a representative picture of what $Y_I(f)$ looks like.

(b) Using appropriate trigonometric identities, show that $y_I(t)$ approximately equals an FM signal with a lower carrier frequency than f_t, and determine the numeric value of that carrier frequency.

(c) Using the RTL-SDR tuned to $f_c = 90.8$ MHz, and with a sample rate of 300,000, please collect 5-10 seconds worth of data. Plot the FFT of y_I. Does it match what you predicted in (a)?

(d) Plot the time domain signal y_I from Part (c). By zooming into different parts of the time-domain signal, find a portion of the signal that clearly indicates that it is an FM signal (you should be able to find a portion of the signal which has varying frequencies). You should submit a plot of this with your report.

(e) Take the derivative of the signal, and zero out its negative components. This operation simulates the derivative followed by the diode portion of the envelope detector. Please normalize this signal so that its maximum value is 1. Please plot the resulting signal.

(f) Low pass filter the signal from Part (e), and normalize it so that its maximum value is 1. Please plot this signal on the **same axes as the previous part**. Zoom in to a portion of the signal which illustrates how the diode followed by low pass filter tracks the envelope of the derivative FM signal.

(g) Subtract the mean out of the signal in the previous part[2], normalize it so that its maximum value is 0.1 to control volume, and decimate it by a factor of 4 using e.g. MATLAB's `decimate` function. The decimation reduces the sample rate to a value that your computer can handle comfortably for audio. Using MATLAB's `sound` command listen to the resulting signal, and make note of its features to compare to the next part. Note that you will have to specify the sampling frequency in the call to `sound`, which will be 300000/4 here.

[1] *Note that actual FM transmissions have a total bandwidth in excess of 200kHz because the transmissions include other information such as stereo channel information and some digital data (e.g. song title). But the mono audio signal occupies less than 30kHz of bandwidth, so these numbers will work for decoding audio*

[2] *Removing the mean helps ensure an inaudible DC current does not ruin your speakers*

Exercise 2: FM Decoding using I and Q channels

Here we will explore a different approach to FM decoding. Now, adjust your SDR MATLAB code to set f_c exactly to the frequency of a desired radio station, e.g. 107.9 MHz, or some other station you like to listen to.

(a) Assume that $\theta(t)$ is small and slowly varying (i.e. both $\theta(t) \approx 0$ and $\frac{d}{dt}\theta(t) \approx 0$)[3]. Using calculus and trigonometric identities, please show why the following is approximately proportional to the original message signal $m(t)$. **Be sure to distinguish $\phi(t)$, the phase offset encoding $m(t)$, from $\theta(t)$, the phase offset between the transmitter and receiver.**

$$\hat{m}(t) = \left(\frac{d}{dt}y_Q(t)\right) y_I(t) - \left(\frac{d}{dt}y_I(t)\right) y_Q(t)$$

(b) Implement the approach described in Part (a) in software and include all your code. Please listen to the resulting signal and compare it to the signal found using the approach in Exercise 1.

C.5 Deliverables

Please turn in an electronic report as a PDF. The report should contain:

- Your name(s)
- Answers to each prompt
- Descriptive title
- Introduction section including research on FM systems, including block diagrams for modulation and demodulation, how we extract $m(t)$, and envelope detection. These findings should be properly cited.
- Methodology section with block diagrams for both exercises
- Results section with all required plots and links to both decoded audio files.
- Conclusion and reflection sections
- Reference section
- An appendix or attached repository with your neatly organized and commented code. Please upload all MATLAB code to Canvas such that it can be downloaded, run and produces the results in your report.

[3] *Optional: Also show that $\theta(t)$ does not have to be small, only slowly varying, for this to work.*

C.6 Tips

i. Collect your data outdoors or by a window to improve signal quality.

ii. Helpful MATLAB commands. Filter design: `sinc`. Filter implementation (i.e. running the filter): `conv filter`. Sample rate reduction: `decimate`. Playing audio: `sound`. Writing audio files: `audiowrite`

iii. You will have to play around with the gain parameter to figure out the right value. There is also an automatic gain control (AGC) which has the RTL-SDR set the gain on its own. You can try using the AGC if you do not wish to set the gain by hand, but it may not work as well as setting it yourself.

iv. Sound cards on your laptops will typically not be able to use very high sampling frequencies. Many laptops can support 192kHz or less. So you should decimate your signal before playing through your speakers. A sample rate below 100kHz should work with most laptops.

v. Remember to remove any DC offset and scale your signal to be below 0.1 in amplitude before playing using `sound`. This will ensure that the volume is appropriate and that a DC current does not ruin your speakers.

Appendix D: Software Defined Radio (SDR) Frequency Modulation (FM) Lab Solutions

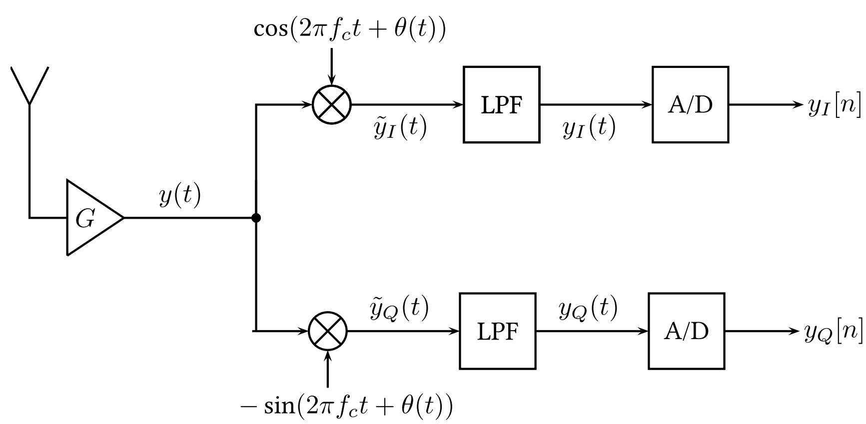
D.1 Introduction

In this assignment, you will use a low-cost software-radio peripheral to decode an FM modulated signal. A software-radio peripheral is a device that receives an analog radio signal and converts it into a digital signal (typically at baseband), to be processed by a computer. You will be using the RTL-SDR which is a receive-only software defined radio peripheral. In other words, this device can be used to receive but not transmit signals.

Please answer all parts labeled **Exercise** in your lab reports. Include all plots, MATLAB code, and a sample of your audio data.

D.2 RTL-SDR Receiver

A functional block diagram for the RTL-SDR system is shown in the following figure.



A (typically large) gain is applied to the signal from an external antenna. You can specify this gain in the tool you use to interface with the RTL-SDR. The resulting signal $y(t)$ is split into two branches. On one branch, $y(t)$ is multiplied by $\cos(2\pi f_c t + \theta(t))$, where $\theta(t)$ is used to model a small, possibly time-varying phase offset between the cosines at the transmitter and the receiver. The resulting signal, $\tilde{y}_I(t) = y(t)\cos(2\pi f_c t + \theta(t))$ is then low pass filtered through a filter whose bandwidth you can specify. That signal is then converted into a digital signal $y_I[n]$ using some sampling rate (sampling frequency) which you can also specify. You should ensure that the bandwidth of the filter is less than $\frac{1}{2}$ the sampling frequency to avoid aliasing. The signal obtained from this branch is called the in-phase component of $y(t)$.

On the second branch, the signal is multiplied by $-\sin(2\pi f_c t + \theta(t))$, resulting in $\tilde{y}_Q(t) = -y(t)\sin(2\pi f_c t + \theta(t))$. $\tilde{y}_Q(t)$ is filtered through a low-pass filter to produce $y_Q(t)$ which is then sampled to produce a digital signal $y_Q[n]$. This signal obtained from this branch is called the quadrature-phase component of $y(t)$.

It turns out that doing things in this manner (i.e. multiplying by cosine and -sine) will simplify a lot of the notation and the math when we are dealing with digital communications. As you saw in problem sets 1 and 2, multiplying by both of these carriers allows us to recover radio signals even when the transmitter and receiver are not synchronized. If the carriers are synchronized, we can instead send two independent data streams on each carrier simultaneously.

$y_I[n]$ and $y_Q[n]$ are the *only* signals you directly obtain from the RTL-SDR.

D.3 Installing RTL-SDR tools

There are several different tools to interface with the RTL-SDR. The most convenient is to use MATLAB. You can install the tools following the instructions on MATLAB RTL-SDR Tools. We have also provided starter code that captures 3 million samples from the RTL-SDR at a particular frequency and puts the data into two vectors `y_I` and `y_Q`. You can change the parameters of the system by modifying the starter code appropriately. The gain parameter may have to be changed depending on the strength of the signal for your desired station, where you collect the data. This number is specified in decibels (dB). If you'd like more practice with decibels check out Appendix B: Logarithms and Decibels.

D.4 FM Demodulation

Exercise 1: Envelope Detector Approach

We have seen how we can cascade a differentiator with an envelope detector (diode + RC circuit) to decode an FM signal using a cheaper circuit. We cannot do this directly with the RTL-SDR as it does not provide direct access to raw FM signals (i.e. it only gives you the $y_I[n]$ and $y_Q[n]$ signals). In other words, you cannot access the FM signal $y(t)$ directly. One work-around for this is to set f_c to a frequency slightly lower than the frequency of the station you would like to access, with a sufficiently high sample rate so that the signal for the targeted station is acquired. E.g. to decode the station at $f_t = 90.9$ MHz, you can set $f_c = 90.8$ MHz, and set the sampling rate to 300 000. Assume that $\theta(t) \approx 0$ for this exercise and neglect its effect in the analysis.

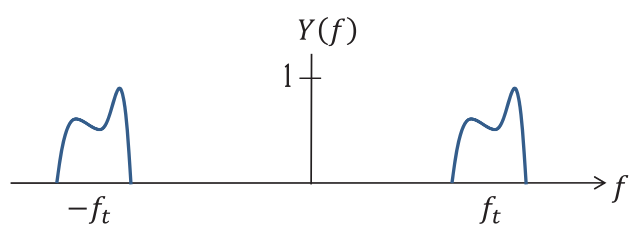


Figure D.1: Frequency domain illustration of a radio station at frequency f_t.

Parts (a) - (d) implement the work-around described above to acquire a received (but not demodulated) FM signal of the form

$$y_r(t) = G\cos\left(2\pi f t + \phi(t)\right)$$

where

$$\phi(t) = 2\pi k_0 \int_{-\infty}^{t} m(\tau)d\tau$$

Parts (e) and (f) demodulate the received FM signal using a derivative, ideal diode, and low pass filter. This envelope detector produces a signal $m(t)+C$ where C is some constant. Part (g) removes the constant term and adjusts the volume to make the received signal speaker friendly.

Solution

In this exercise, we first work in Parts (a) - (d) to acquire a received (but not demodulated) FM signal in the form of $y_r(t) = G\cos(2\pi f t + \phi(t))$. In doing so, we set f_c to a slightly lower frequency than the station we would like to access, with a sufficiently high sample rate so that the signal for the target station is acquired. For example, we can decode the station at $f_t = 90.9$ MHz for an $f_c = 90.8$ MHz and a sampling rate to 300 000.

The second part of this exercise (Parts e and f) demodulate the received FM signal using a derivative, diode, and LPF. This envelope detector produces a signal $m(t)$ where C is some constant. Part (g) removes the constant term and adjusts the volume to make the received signal speaker friendly.

(a) Suppose that the signal you wish to decode is a station broadcasting at $f_t = 90.9$ MHz, and has a bandwidth[1] of 80kHz. In the frequency domain, this would correspond to two blobs at ± 90.9 MHz, as illustrated in Figure D.1. If you set $f_c = 90.8$ MHz, and assuming $\theta(t)$ is negligible, please sketch a representative picture of what $Y_I(f)$ looks like.

Solution

Given that our desired station has a frequency $f_t = 90.9$ MHz with a bandwidth of 80 kHz. The frequency domain representation would be two blobs at ± 90.9 MHz. If we set f_c to 90.8 MHz, the blobs should result in the figure of $Y_I(f)$ below.

[1] *Note that actual FM transmissions have a total bandwidth in excess of 200kHz because the transmissions include other information such as stereo channel information and some digital data (e.g. song title). But the mono audio signal occupies less than 30kHz of bandwidth, so these numbers will work for decoding audio*

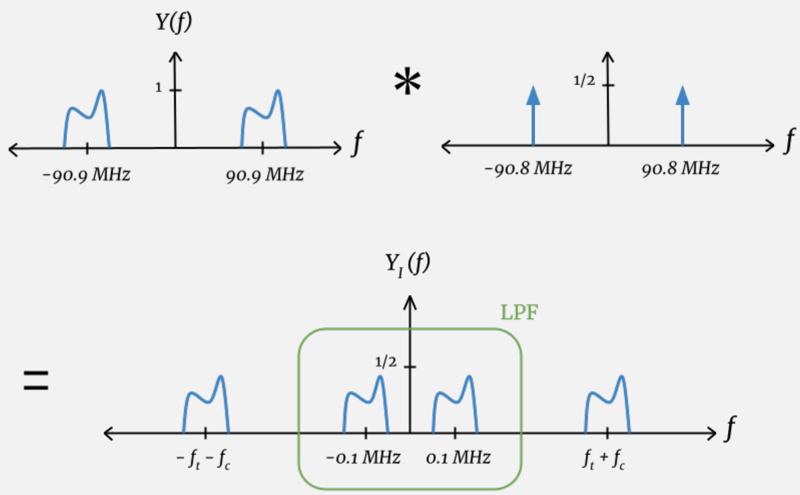


Figure D.2: Depiction of the convolution process to obtain $Y_I(f)$

Since there was an offset of 100 kHz between f_t and f_c, and theta is negligible in this scenario, it is expected for convolution to work differently. What would have otherwise been a single blob at the center of the cosine convolution is instead two smaller blobs relatively close to each other compared to the two outermost blobs. Before getting to $Y_I(f)$, we also need to apply a low pass filter to the signal, which means dropping the two outermost blobs. In the end, there are two relatively close center blobs, shown in the green box in Figure D.2.

(b) Using appropriate trigonometric identities, show that $y_I(t)$ approximately equals an FM signal with a lower carrier frequency than f_t, and determine the numeric value of that carrier frequency.

Solution

The information signal is embedded in the varying phase of the transmitted FM signal, u(t), where

$$u(t) = A_c \cos(2\pi f_t t + \phi(t))$$

u(t) can then be mixed with $\cos(2\pi f_c t + \theta(t))$ to get $\bar{y}_I(t)$

$$\bar{y}_I(t) = u(t) * \cos(2\pi f_c t + \theta(t))$$

By plugging in u(t), $\bar{y}_I(t)$ becomes

$$\bar{y}_I(t) = A_c \cos(2\pi f_t t + \phi(t)) * \cos(2\pi f_c t + \theta(t))$$

For this exercise, $\theta(t)$ is assumed to be negligible, so it can me zeroed out to give

$$\bar{y}_I(t) = A_c \cos(2\pi f_t t + \phi(t)) * \cos(2\pi f_c t)$$

By using trigonometric identities, the equation above can be written as

$$\bar{y}_I(t) = \frac{A_c}{2} \cos(2\pi t(f_t - f_c) + \phi(t)) * \cos(2\pi t(f_t + f_c))$$

A low pass filter can get rid of the $\cos(2\pi(f_t + f_c)t)$ term, which results in $y_I(t)$

$$y_I(t) = \frac{A_c}{2} \cos(2\pi t(f_t - f_c) + \phi(t))$$

$y_I(t)$ is similar to $u(t)$ except the carrier frequency for $y_I(t)$ is $f_t - f_c$, whereas the carrier frequency of $u(t)$ is simply f_t.

(c) Using the RTL-SDR tuned to $f_c = 90.8$ MHz, and with a sample rate of 300,000, please collect 5-10 seconds worth of data. Plot the FFT of y_I. Does it match what you predicted in (a)?

Solution

The FFT plot of y_I should look similar to your prediction in (a), with two peaks occurring at around $\pm f_t - f_c$, or 100 kHz. This follows Part (b), showing that a pseudo FM signal is present.

```
%% Starter Code
fs = 300e3; % this is the sample rate
fc = 90.8e6; % this is the center frequency

t = linspace(1,fs*5,fs*5+160);
t = t.';
x = zeros(fs*5,1); % empty vector to store data

% create object for RTL-SDR receiver
rx = comm.SDRRTLReceiver('CenterFrequency',fc, 'EnableTunerAGC', false, '
    TunerGain', 35,  'SampleRate', fs);

counter = 1; % initialize a counter
while(counter < length(x)) % while the buffer for data is not full
    rxdata = rx();   % read from the RTL-SDR
    x(counter:counter + length(rxdata)-1) = rxdata; % save the samples
        returned
    counter = counter + length(rxdata); % increment counter
```

```
end
% the data are returned as complex numbers
% separate real and imaginary part, and remove any DC offset
y_I = real(x)-mean(real(x));
y_Q = imag(x)-mean(imag(x));
y_I2 = y_I;

%% Part c
figure
plot_FT(y_I,fs);
xlabel('Frequency (Hz)')
ylabel('Amplitude')
```

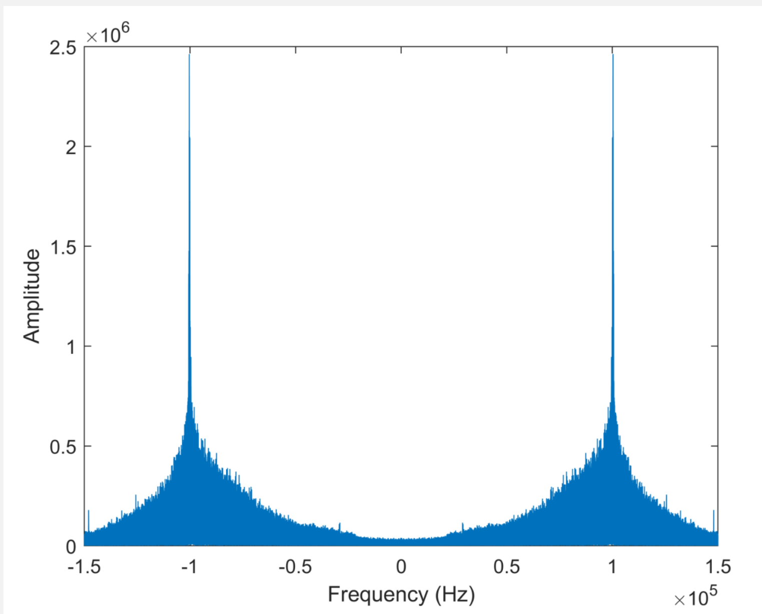


Figure D.3: Plot of y_I signal in frequency domain (FFT).

(d) Plot the time domain signal y_I from Part (c). By zooming into different parts of the time-domain signal, find a portion of the signal that clearly indicates that it is an FM signal (you should be able to find a portion of the signal which has varying frequencies). You should submit a plot of this with your report.

Solution

In this part, convert the signal into the time domain. Then zoom in to show a difference in wavelength which represented the modulated signal over time. Plots may vary based on the time scale that is zoomed into.

```
%% Part d
t2 = linspace(218500,224499,6000)'./fs;
plot(t2,y_I(218500:224499));
xlabel('Time (Seconds)');
ylabel('Amplitude');
```

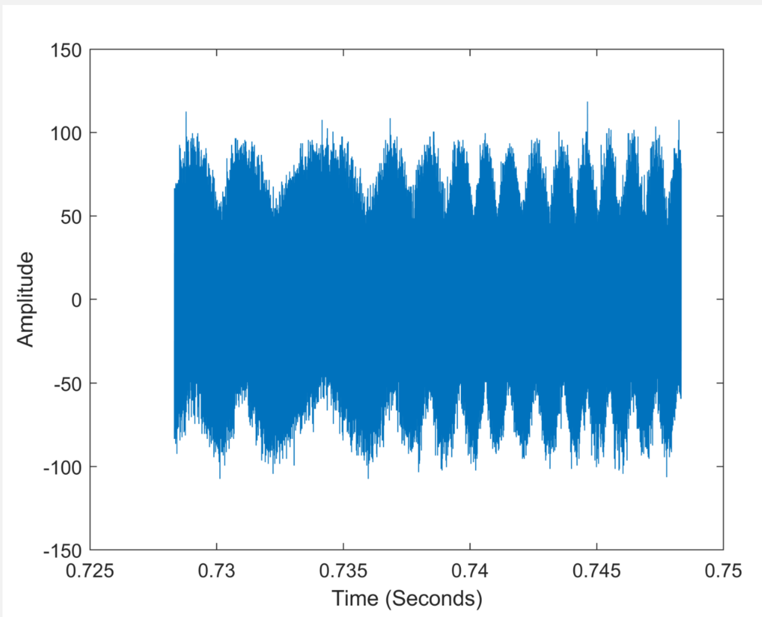


Figure D.4: Plot of a subsection of the y_I signal in time domain.

(e) Take the derivative of the signal, and zero out its negative components. This operation simulates the derivative followed by the diode portion of the envelope detector. Please normalize this signal so that its maximum value is 1. Please plot the resulting signal.

Solution

```
%% Part e
dy_I = diff(y_I);

len = size(dy_I,1);
maxval = max(dy_I);

for i = 1:len
    dy_I(i) = dy_I(i)./maxval;
    if dy_I(i) < 0
        dy_I(i) = 0;
    end
end

plot(linspace(218500,224499,6000)'./fs, dy_I(218500:224499));
xlabel('Time (Seconds)');
ylabel('Amplitude');
```

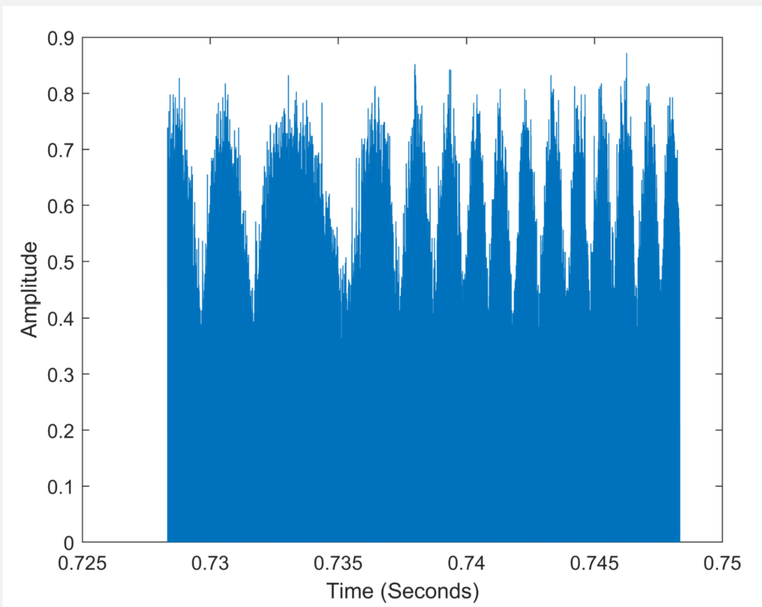


Figure D.5: Plot of differentiated and normalized y_I signal in time domain.

(f) Low pass filter the signal from Part (e), and normalize it so that its maximum value is 1. Please plot this signal on the **same axes as the previous part**. Zoom in to a portion of the signal which illustrates how the diode followed by low pass filter tracks the envelope of the derivative FM signal.

Solution

```
%% Part f
wpass = 10000;
dy_ILPF = lowpass(dy_I,wpass,fs); % creating lowpass filter
dy_ILPFnorm = dy_ILPF/max(dy_ILPF); % normalizing signal

plot(linspace(218500,224499,6000)'./fs, dy_ILPFnorm(218500:224499);
xlabel('Time (Seconds)');
ylabel('Amplitude');
```

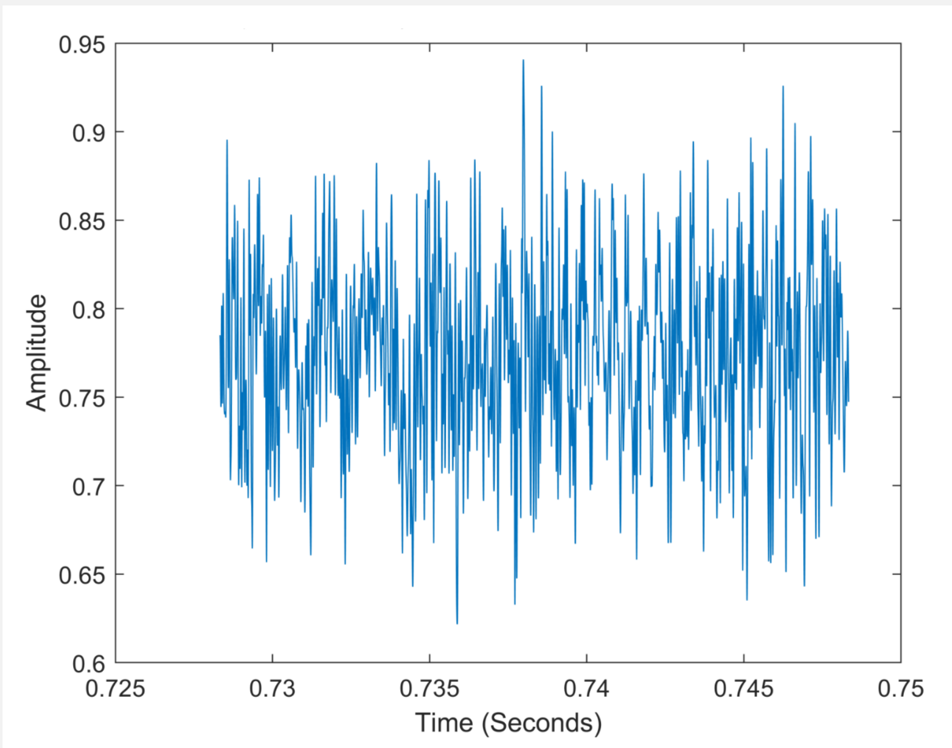


Figure D.6: Plot of differentiated, normalized, and filtered y_I signal subsection in time domain.

(g) Subtract the mean out of the signal in the previous part[2], normalize it so that its maximum value is 0.1 to control volume, and decimate it by a factor of 4 using e.g. MATLAB's `decimate` function. The decimation reduces the sample rate to a value that your computer can handle comfortably for audio. Using MATLAB's `sound` command listen to the resulting signal, and make note of its features to

[2] *Removing the mean helps ensure an inaudible DC current does not ruin your speakers*

compare to the next part. Note that you will have to specify the sampling frequency in the call to `sound`, which will be 300000/4 here.

Solution

```
%% Part g
dy_Isubmean = dy_ILPFnorm - mean(dy_ILPFnorm);
dy_Inorm10 = dy_Isubmean / (10*max(dy_Isubmean));
dy_Idecimated = decimate(dy_Inorm10,4);

plot(linspace(1,length(dy_Idecimated),length(dy_Idecimated))'./fs,
    dy_Idecimated);
xlabel('Time (Seconds)');
ylabel('Amplitude');

% Play and save audio file
sound(dy_Idecimated,fs/4);
audiowrite(Exercise1.wav, dy_Idecimated,fs/4);
```

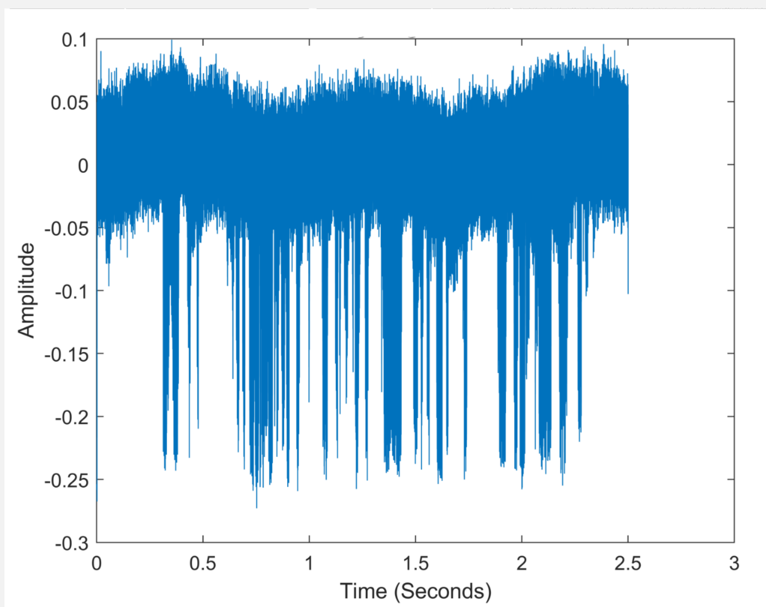


Figure D.7: Plot of message signal y_I in time domain.

Exercise 2: FM Decoding using I and Q channels

Here we will explore a different approach to FM decoding. Now, adjust your SDR MATLAB code to set f_c exactly to the frequency of a desired radio station, e.g. 107.9 MHz, or some other station you like to listen to.

(a) Assume that $\theta(t)$ is small and slowly varying (i.e. both $\theta(t) \approx 0$ and $\frac{d}{dt}\theta(t) \approx 0$)[3]. Using calculus and trigonometric identities, please show why the following is approximately proportional to the original message signal $m(t)$. **Be sure to distinguish $\phi(t)$, the phase offset encoding $m(t)$, from $\theta(t)$, the phase offset between the transmitter and receiver.**

$$\hat{m}(t) = \left(\frac{d}{dt}y_Q(t)\right) y_I(t) - \left(\frac{d}{dt}y_I(t)\right) y_Q(t)$$

Solution

Starting with the I-branch, write $y_I(t)$, which is the signal before sampling but after the low pass filter:

$$y_I(t) = LPF\{G\cos(2\pi f_c t + \phi(t))\cos(2\pi f_c t + \theta(t))\}$$

Because $\theta(t)$ is small, it can be removed, leaving the product. The receive gain factor G can also be factored outside the LPF:

$$y_I(t) = G * LPF\{\cos(2\pi f_c t + \phi(t))\cos(2\pi f_c t)\}$$

Using the product-to-sum identity $\cos(A)\cos(B) = \frac{1}{2}[\cos(A+B) + \cos(A-B)]$, write the expression on the right in a form that allows the low pass filter to be applied , where $A = 2\pi f_c t + \phi(t)$ and $B = 2\pi f_c t$.

$$y_I(t) = G * \frac{1}{2}[\cos((2\pi f_c t + \phi(t)) + (2\pi f_c t)) + \cos((2\pi f_c t + \phi(t)) - (2\pi f_c t))]$$
$$y_I(t) = \frac{G}{2}[\cos(4\pi f_c t + \phi(t)) + \cos(\phi(t))]$$

The term at $4\pi f_c t$ is removed by the LPF, leaving the relatively simple equation for $y_I(t)$:

$$y_I(t) = \frac{G}{2}\cos(\phi(t))$$

For the Q-branch, write $y_Q(t)$ similarly:

$$y_Q(t) = LPF\{G\cos(2\pi f_c t + \phi(t))\sin(2\pi f_c t + \theta(t))\}$$

Remove $\theta(t)$ and factor out G like before.

$$y_Q(t) = G * LPF\{\cos(2\pi f_c t + \phi(t))\sin(2\pi f_c t)\}$$

Use the product-to-sum identity $\cos(A)\sin(B) = \frac{1}{2}[\sin(A+B) + \sin(A-B)]$ to write:

$$y_Q(t) = G * \frac{1}{2}[\sin(2\pi f_c t + \phi(t) + 2\pi f_c t) + \sin(2\pi f_c t + \phi(t) - 2\pi f_c t)]$$
$$y_Q(t) = \frac{G}{2}[\sin(4\pi f_c t + \phi(t)) + \sin(\phi(t))]$$

[3] *Optional: Also show that $\theta(t)$ does not have to be small, only slowly varying, for this to work.*

The LPF removes the $2f_c$ term, giving $y_Q(t)$.

$$y_Q(t) = \frac{G}{2}\sin(\phi(t))$$

With equations for both $y_I(t)$ and $y_Q(t)$, rewrite the $\hat{m}(t)$.

$$\hat{m}(t) = \left[\frac{d}{dt}\left(\frac{G}{2}\sin(\phi(t))\right)\right]\left[\frac{G}{2}\cos(\phi(t))\right] - \left[\frac{d}{dt}\left(\frac{G}{2}\cos(\phi(t))\right)\right]\left[\frac{G}{2}\sin(\phi(t))\right]$$

Factor out the $\frac{G}{2}$ terms.

$$\hat{m}(t) = \frac{G^2}{4}(\frac{d}{dt}\sin(\phi(t)))\cos(\phi(t)) - \frac{G^2}{4}(\frac{d}{dt}\cos(\phi(t)))\sin(\phi(t))$$

Using the chain rule, apply the derivatives.

$$\hat{m}(t) = \frac{G^2}{4}\phi'(t)\cos(\phi(t))\cos(\phi(t)) - \frac{G^2}{4}(-\phi'(t)\sin(\phi(t))\sin(\phi(t))$$

Factor the $\frac{G^2}{4}$ and $\phi'(t)$ terms.

$$\hat{m}(t) = \frac{G^2}{4}\phi'(t)[\cos^2(\phi(t)) + \sin^2(\phi(t))]$$

Using the Pythagorean identity, $\hat{m}(t)$ is now:

$$\hat{m}(t) = \frac{G^2}{4}\phi'(t)$$

At this point, recall that an FM radio signal encodes the message in $\phi(t)$ as the following:

$$\phi(t) = 2\pi k_0 \int_{-\infty}^{t} m(\tau)d\tau$$

The derivative of this integral is just $m(t)$, which means $\hat{m}(t)$ is finally:

$$\hat{m}(t) = \frac{2\pi k_0 G^2}{4}m(t)$$

Thus, we have shown that $\hat{m}(t)$ is proportional to $m(t)$.

(b) Implement the approach described in Part (a) in software and include all your code. Please listen to the resulting signal and compare it to the signal found using the approach in Exercise 1.

Solution

```
m_hat = (y_I(1:length(y_I)-1).*diff(y_Q)) - (y_Q(1:length(y_Q)-1).*diff(y_I));

len = size(m_hat,1);
maxval = max(m_hat);

for i = 1:len
    m_hat(i) = m_hat(i)/maxval;
    if m_hat(i) < 0
        m_hat(i) = 0;
    end
end

wpass = 10000;
m_hatLPF = lowpass(m_hat,wpass,fs);
m_hatLPFnorm = m_hatLPF/max(m_hatLPF);

m_submean = m_hatLPFnorm - mean(m_hatLPFnorm);
m_norm10 = m_submean / (10*max(m_submean));
m_decimated = decimate(m_norm10,4);

plot(linspace(1,length(m_decimated),length(m_decimated))'./fs,m_decimated);
xlabel('Time (Seconds)');
ylabel('Amplitude');
```

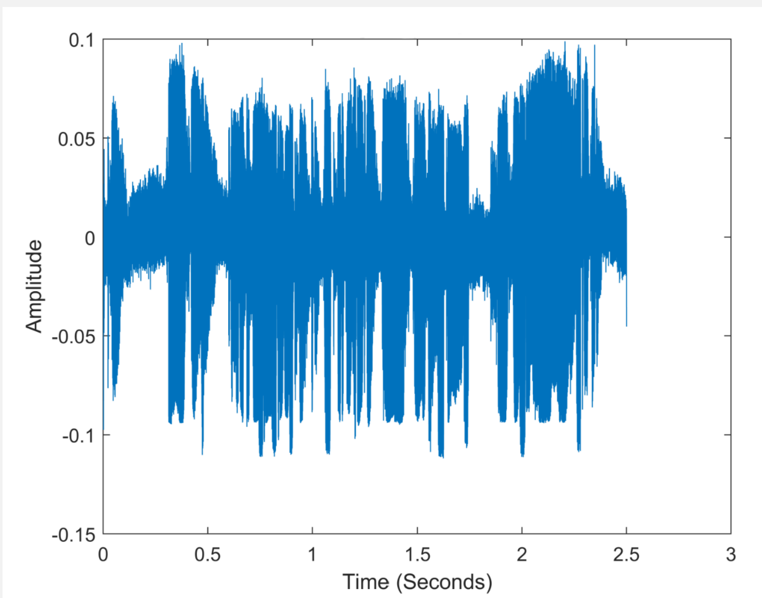


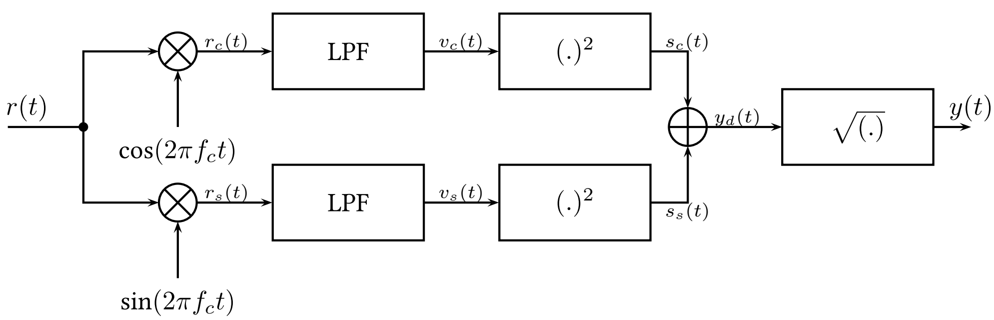
Figure D.8: Plot of message signal $\hat{m}$ in time domain.

The figure above differs from the figure in Exercise 1 since $\hat{m}(t)$ does not need its negative components to be zeroed out, nor does it require the magnitude of the signal to be normalized. In comparing the audio quality between Exercises 1 and 2, there is a negligible difference.

Appendix E: In-Class Worksheets

Problem 1: Amplitude Modulation (Adapted from Oppenheim and Willsky '97)

Consider the following system, which takes an input signal of the form $r(t) = (A + m_n(t))\cos(2\pi f_c t + \theta_c)$, where $A + m_n(t) > 0$ for all t. Let $m(t)$ be band-limited to f_M, i.e. $M(f) = 0$ for all $|f| > f_M$. Furthermore, assume that $f_M \ll f_c$, and the low-pass filters (LPF) have cutoff frequencies of f_c and that θ_c is an unknown but constant phase-offset.



Note that the blocks marked $(.)^2$ are devices which square the input signals and the block marked $\sqrt{(.)}$ takes the square-root of the input signal. Show that $y(t) = Gx(t)$, where G is a scale factor and $x(t) = (A + m_n(t))$.

Hint: You can work in the time domain using trigonometric identities or the frequency domain using pictures and Fourier Transform properties or both. In the frequency domain, using some representative, asymmetric plot for $X(f)$, you don't have to explicitly convolve anything, just represent the results of any convolutions with representative plots that are clearly labeled.

Problem 1 Solution:

Time Domain:

The input signal

$$r(t) = (A + m_n(t))\cos(2\pi f_c t + \theta_c)$$

is split into two signals, one multiplied by cosine $r_c(t)$ and the other multiplied by sine $r_s(t)$. The time domain equation after multiplication by cosine is

$$r_c(t) = (A + m_n(t))\cos(2\pi f_c t + \theta_c)\cos(2\pi f_c t)$$

Using the identity

$$\cos(\theta)\cos(\phi) = \frac{\cos(\theta - \phi) + \cos(\theta + \phi)}{2}$$

$r_c(t)$ can be written as a sum instead of a product of cosines

$$r_c(t) = \frac{1}{2}(A + m_n(t))(\cos\theta_c + \cos(4\pi f_c t))$$

As a sum, it is clearer that the cosine at $2f_c$ is filtered out by the lowpass filter, giving $v_c(t)$

$$v_c(t) = \frac{1}{2}(A + m_n(t))\cos\theta_c$$

The squared signal following the LPF is then

$$s_c(t) = \frac{1}{4}(A + m_n(t))^2\cos^2\theta_c$$

On the other branch, multiplication by sine gives

$$r_s(t) = (A + m_n(t))\cos(2\pi f_c t + \theta_c)\sin(2\pi f_c t)$$

Using the identity

$$\sin(\theta)\cos(\phi) = \frac{\sin(\theta + \phi) + \sin(\theta - \phi)}{2}$$

$r_s(t)$ can also be written as a sum instead of a product

$$r_s(t) = \frac{1}{2}(A + m_n(t))(\sin(4\pi f_c t) + \sin\theta_c)$$

The high frequency sine at $2f_c$ is filtered out, giving $v_s(t)$

$$v_s(t) = \frac{1}{2}(A + m_n(t))\sin\theta_c$$

The squared signal following the LPF is then

$$s_s(t) = \frac{1}{4}(A + m_n(t))^2\sin^2\theta_c$$

After the multiplication, lowpass filtering, and squaring the signals $s_c(t)$ and $s_s(t)$ are added together

$$\begin{aligned}
y_d(t) &= s_c(t) + s_s(t) \\
&= \frac{1}{4}(A + m_n(t))^2\cos^2\theta_c + \frac{1}{4}(A + m_n(t))^2\sin^2\theta_c \\
&= \frac{1}{4}(A + m_n(t))^2\left[\cos^2\theta_c + \sin^2\theta_c\right]
\end{aligned}$$

The Pythagorean identity $1 = \cos^2\theta + \sin^2\theta$ gives

$$y_d(t) = \frac{1}{4}(A + m_n(t))^2$$

Finally, the square root gives the output $y(t)$

$$y(t) = \frac{1}{2}(A + m_n(t))$$

Therefore $y(t) = Gx(t)$ with $G = \frac{1}{2}$.

Frequency Domain:

Let $X(f)$ be represented by the blob

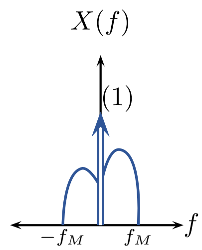


Figure E.1: Sketch of a blob to represent $X(f)$

The modulated signal is convolved by a cosine with a phase offset θ_c to give $R(f)$

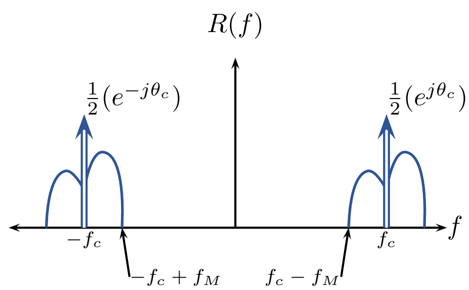


Figure E.2: Sketch of $R(f)$

$r(t)$ is multiplied by the cosine, so $R(f)$ is convolved by cosine, giving $R_c(f)$ on the top branch

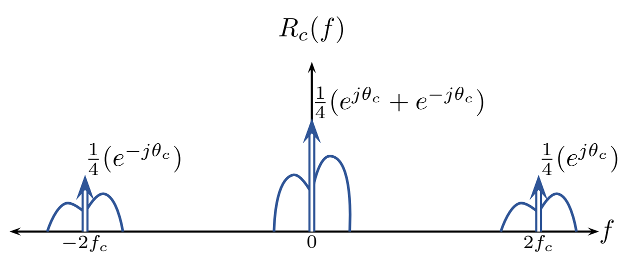


Figure E.3: Sketch of $R_c(f)$

Similarly, convolving $R(f)$ by sine gives $R_s(f)$ on the lower branch

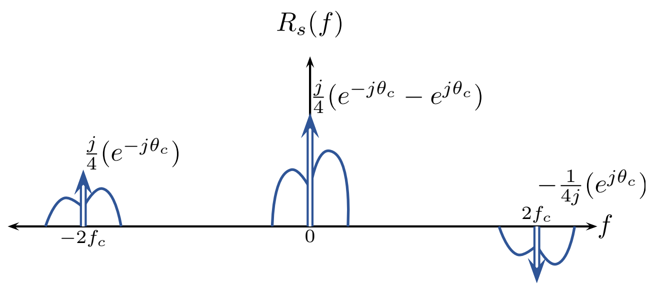


Figure E.4: Sketch of $R_s(f)$

Let $F[x^2(t)] = X(f) * X(f)$ be represented by the blob

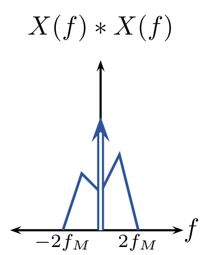


Figure E.5: Sketch of a blob to represent $F[x^2(t)] = X(f) * X(f)$

After the high frequency components are removed by the LPF, the squared signals on each branch $S_c(f)$ and $S_s(f)$ are

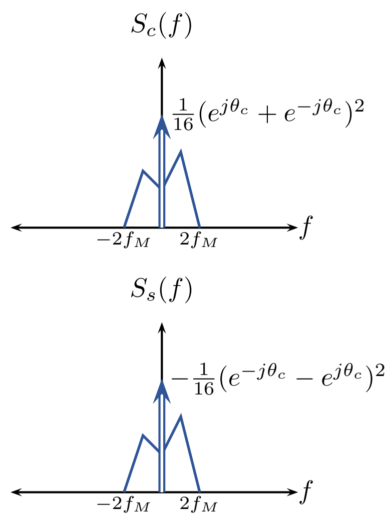


Figure E.6: Sketch of $S_c(f)$ and $S_s(f)$

$S_c(f)$ and $S_s(f)$ are added together to get $Y_s(f)$. Using the identities

$$\cos\theta = \frac{e^{j\theta} - e^{-j\theta}}{2} \text{ and}$$
$$j\sin\theta = \frac{e^{j\theta} - e^{-j\theta}}{2}$$

the height of $Y_s(f)$ is the sum of the heights of $S_c(f)$ and $S_s(f)$ and is simplified to

$$\begin{aligned}\frac{1}{16}\left[(e^{j\theta_c} + e^{-j\theta_c})^2 - (e^{-j\theta_c} - e^{j\theta_c})^2\right] &= \frac{1}{16}\left[4\cos^2\theta_c + 4\sin^2\theta_c\right] \\ &= \frac{1}{16}(4) \\ &= \frac{1}{4}\end{aligned}$$

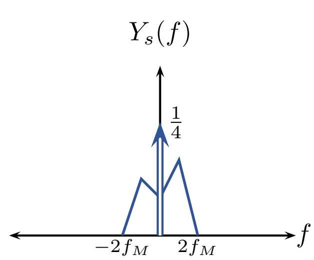


Figure E.7: Sketch of $Y_s(f) = S_c(f) + S_s(f)$ which has height $\frac{1}{4}$

$\sqrt{X(f) * X(f)} = X(f)$ when $X(f)$ is positive. Therefore, the square root finally gives the original signal $X(f)$ from $Y_s(f)$ with a scale factor $G = \frac{1}{2}$.

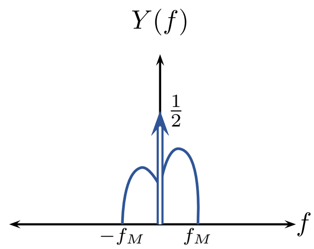


Figure E.8: Sketch of $Y(f) = \frac{1}{2}X(f)$

Bibliography

[1] International Telecommunications Union. *Statistics*. 2019. URL: https://www.itu.int/en/ITU-D/Statistics/Pages/stat/default.aspx (visited on 09/01/2021).

[2] J. G. Proakis and M. Salehi. *Fundamentals of Communications Systems*. Hyde Park, MA: Pearson, 2004.

[3] GitHub User phiresky. *Convolution Demo*. URL: https://phiresky.github.io/convolution-demo/ (visited on 06/06/2021).

[4] La tour Eiffel. *The Eiffel Tower and Science*. URL: https://www.toureiffel.paris/en/the-monument/eiffel-tower-and-science (visited on 01/13/2022).

[5] C. Nave. *AM and FM Radio Frequencies*. URL: http://hyperphysics.phy-astr.gsu.edu/hbase/Audio/radio.html.

Index