Communication Systems

Collection Editor:
Janko Calic

Communication Systems

Collection Editor:
Janko Calic

Authors:

Thanos Antoulas
Richard Baraniuk
Dan Calderon
Catherine Elder
Anders Gjendemsjø
Michael Haag
Matthew Hutchinson
Don Johnson
Stephen Kruzick
Robert Nowak
Ricardo Radaelli-Sanchez
Justin Romberg
Phil Schniter
Melissa Selik
JP Slavinsky

Online:
< http://cnx.org/content/col10631/1.3/ >

OpenStax-CNX

Collection structure revised: January 21, 2009
PDF generated: October 29, 2015
For copyright and attribution information for the modules contained in this collection, see p. 145.

Table of Contents

Chapter 1

Signals and Systems in Communications

1.1 Signals

1.1.1 Signal Classifications and Properties[1]

1.1.1.1 Introduction

This module will begin our study of signals and systems by laying out some of the fundamentals of signal classification. It is essentially an introduction to the important definitions and properties that are fundamental to the discussion of signals and systems, with a brief discussion of each.

1.1.1.2 Classifications of Signals

1.1.1.2.1 Continuous-Time vs. Discrete-Time

As the names suggest, this classification is determined by whether or not the time axis is **discrete** (countable) or **continuous** (Figure 1.1). A continuous-time signal will contain a value for all real numbers along the time axis. In contrast to this, a discrete-time signal (Section 1.1.6), often created by sampling a continuous signal, will only have values at equally spaced intervals along the time axis.

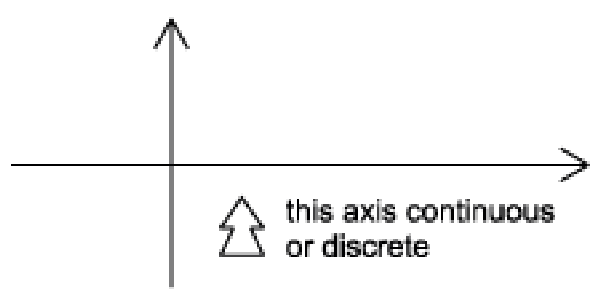


Figure 1.1

[1]This content is available online at <http://cnx.org/content/m10057/2.23/>.

1.1.1.2.2 Analog vs. Digital

The difference between **analog** and **digital** is similar to the difference between continuous-time and discrete-time. However, in this case the difference involves the values of the function. Analog corresponds to a continuous set of possible function values, while digital corresponds to a discrete set of possible function values. An common example of a digital signal is a binary sequence, where the values of the function can only be one or zero.

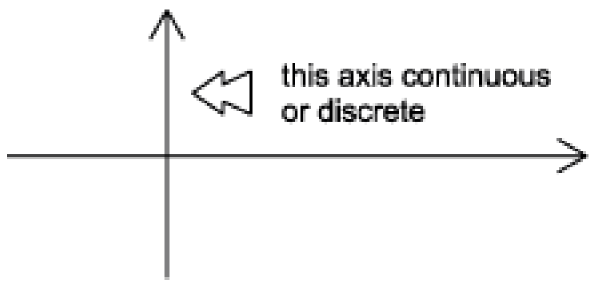


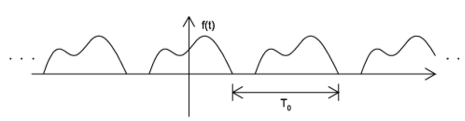
Figure 1.2

1.1.1.2.3 Periodic vs. Aperiodic

Periodic signals[2] repeat with some **period** T, while aperiodic, or nonperiodic, signals do not (Figure 1.3). We can define a periodic function through the following mathematical expression, where t can be any number and T is a positive constant:

$$f(t) = f(t+T) \tag{1.1}$$

fundamental period of our function, $f(t)$, is the smallest value of T that the still allows (1.1) to be true.



(a)

(b)

Figure 1.3: (a) A periodic signal with period T_0 (b) An aperiodic signal

[2]"Continuous Time Periodic Signals" <http://cnx.org/content/m10744/latest/>

1.1.1.2.4 Finite vs. Infinite Length

Another way of classifying a signal is in terms of its length along its time axis. Is the signal defined for all possible values of time, or for only certain values of time? Mathematically speaking, $f(t)$ is a **finite-length signal** if it is **defined** only over a finite interval

$$t_1 < t < t_2$$

where $t_1 < t_2$. Similarly, an **infinite-length signal**, $f(t)$, is defined for all values:

$$-\infty < t < \infty$$

1.1.1.2.5 Causal vs. Anticausal vs. Noncausal

Causal signals are signals that are zero for all negative time, while **anticausal** are signals that are zero for all positive time. **Noncausal** signals are signals that have nonzero values in both positive and negative time (Figure 1.4).

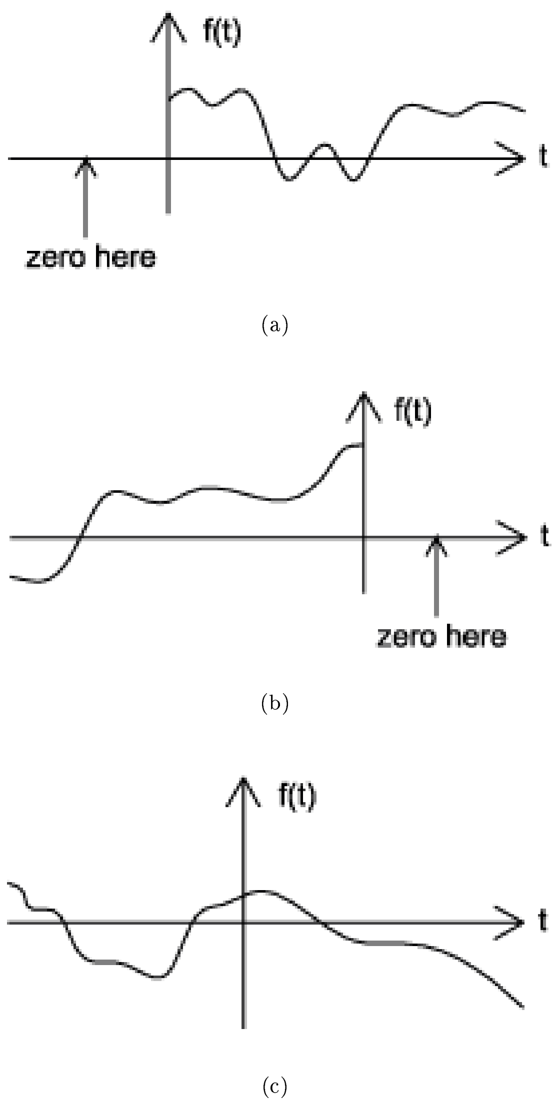
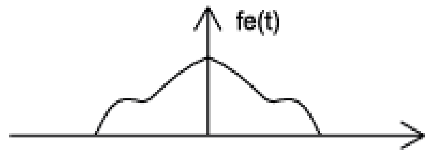


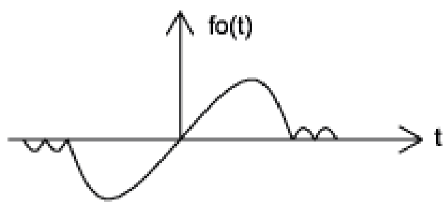
Figure 1.4: (a) A causal signal (b) An anticausal signal (c) A noncausal signal

1.1.1.2.6 Even vs. Odd

An **even signal** is any signal f such that $f(t) = f(-t)$. Even signals can be easily spotted as they are **symmetric** around the vertical axis. An **odd signal**, on the other hand, is a signal f such that $f(t) = -f(-t)$ (Figure 1.5).



(a)



(b)

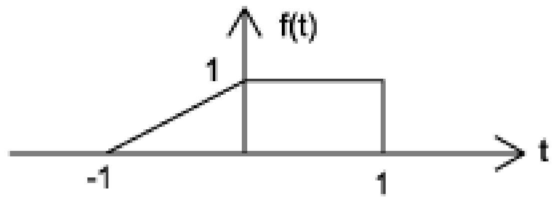
Figure 1.5: (a) An even signal (b) An odd signal

Using the definitions of even and odd signals, we can show that any signal can be written as a combination of an even and odd signal. That is, every signal has an odd-even decomposition. To demonstrate this, we have to look no further than a single equation.

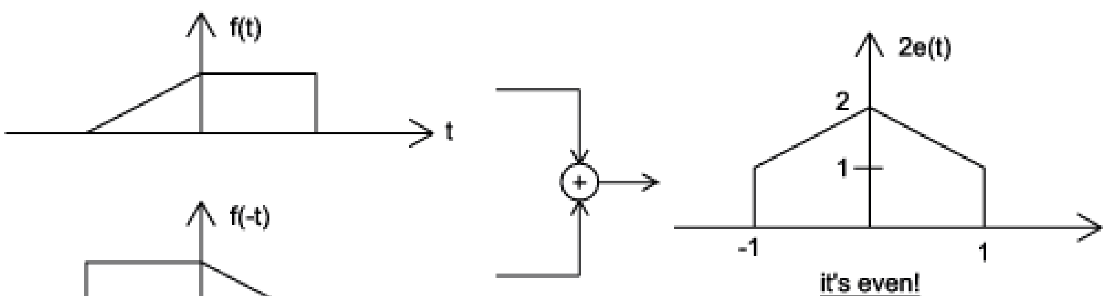
$$f(t) = \frac{1}{2}(f(t) + f(-t)) + \frac{1}{2}(f(t) - f(-t)) \quad (1.2)$$

By multiplying and adding this expression out, it can be shown to be true. Also, it can be shown that $f(t) + f(-t)$ fulfills the requirement of an even function, while $f(t) - f(-t)$ fulfills the requirement of an odd function (Figure 1.6).

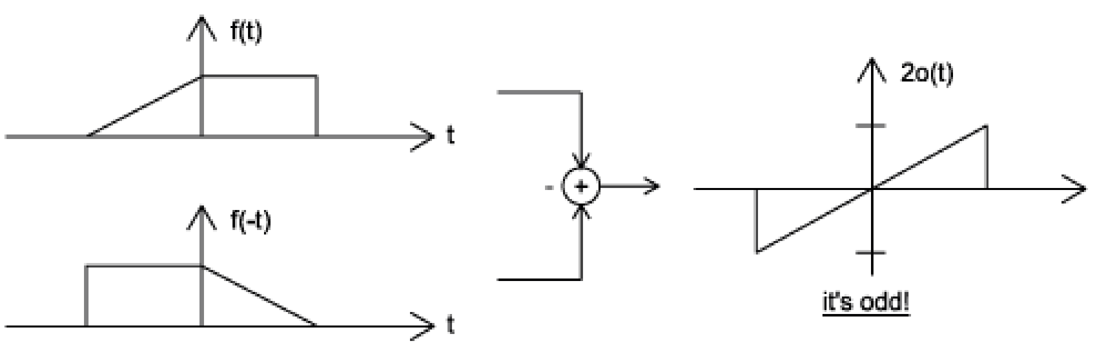
Example 1.1



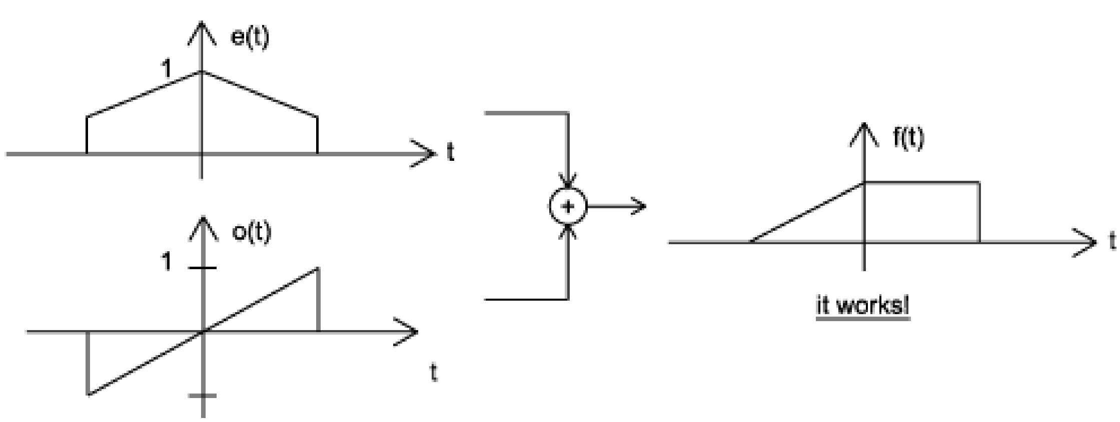
(a)



(b)



(c)

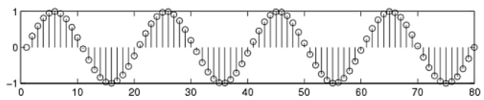


(d)

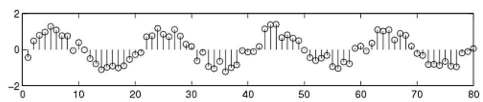
Figure 1.6: (a) The signal we will decompose using odd-even decomposition (b) Even part: $e(t) = \frac{1}{2}(f(t) + f(-t))$ (c) Odd part: $o(t) = \frac{1}{2}(f(t) - f(-t))$ (d) Check: $e(t) + o(t) = f(t)$

1.1.1.2.7 Deterministic vs. Random

A **deterministic signal** is a signal in which each value of the signal is fixed, being determined by a mathematical expression, rule, or table. On the other hand, the values of a **random signal**[3] are not strictly defined, but are subject to some amount of variability.



(a)



(b)

Figure 1.7: (a) Deterministic Signal (b) Random Signal

Example 1.2
Consider the signal defined for all real t described by

$$f(t) = \begin{cases} sin(2\pi t)/t & t \geq 1 \\ 0 & t < 1 \end{cases} \tag{1.3}$$

This signal is continuous time, analog, aperiodic, infinite length, causal, neither even nor odd, and, by definition, deterministic.

1.1.1.3 Signal Classifications Summary

This module describes just some of the many ways in which signals can be classified. They can be continuous time or discrete time, analog or digital, periodic or aperiodic, finite or infinite, and deterministic or random. We can also divide them based on their causality and symmetry properties.

1.1.2 Signal Operations[4]

1.1.2.1 Introduction

This module will look at two signal operations affecting the time parameter of the signal, time shifting and time scaling. These operations are very common components to real-world systems and, as such, should be understood thoroughly when learning about signals and systems.

[3]"Introduction to Random Signals and Processes" <http://cnx.org/content/m10649/latest/>
[4]This content is available online at <http://cnx.org/content/m10125/2.18/>.

1.1.2.2 Manipulating the Time Parameter

1.1.2.2.1 Time Shifting

Time shifting is, as the name suggests, the shifting of a signal in time. This is done by adding or subtracting a quantity of the shift to the time variable in the function. Subtracting a fixed positive quantity from the time variable will shift the signal to the right (delay) by the subtracted quantity, while adding a fixed positive amount to the time variable will shift the signal to the left (advance) by the added quantity.

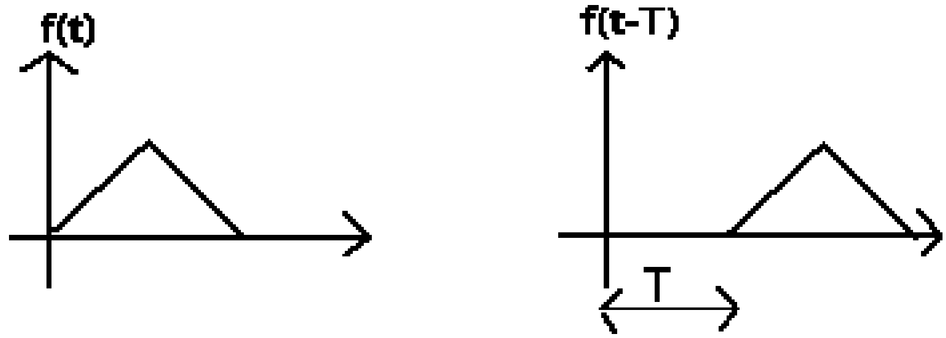


Figure 1.8: $f(t-T)$ moves (delays) f to the right by T.

1.1.2.2.2 Time Scaling

Time scaling compresses or dilates a signal by multiplying the time variable by some quantity. If that quantity is greater than one, the signal becomes narrower and the operation is called compression, while if the quantity is less than one, the signal becomes wider and is called dilation.

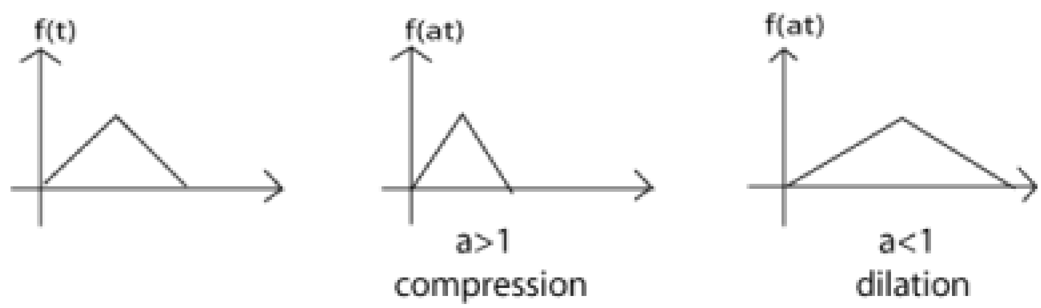


Figure 1.9: $f(at)$ compresses f by a.

Example 1.3
Given $f(t)$ we woul like to plot $f(at-b)$. The figure below describes a method to accomplish this.

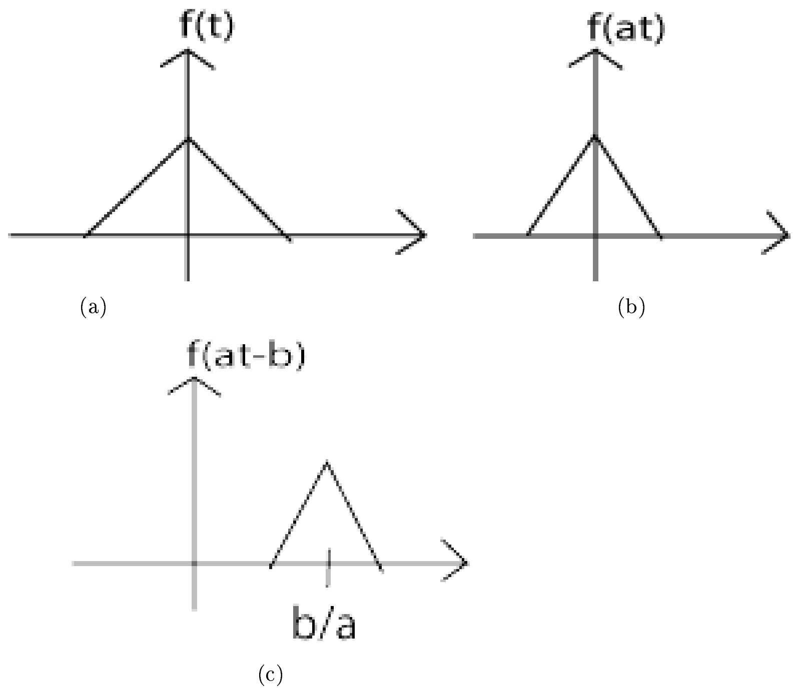


Figure 1.10: (a) Begin with $f(t)$ (b) Then replace t with at to get $f(at)$ (c) Finally, replace t with $t-\frac{b}{a}$ to get $f\left(a\left(t-\frac{b}{a}\right)\right)=f(at-b)$

1.1.2.2.3 Time Reversal

A natural question to consider when learning about time scaling is: What happens when the time variable is multiplied by a negative number? The answer to this is time reversal. This operation is the reversal of the time axis, or flipping the signal over the y-axis.

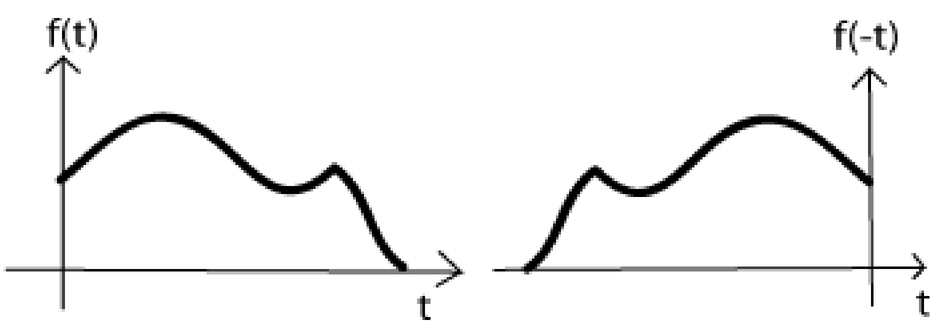


Figure 1.11: Reverse the time axis

1.1.2.3 Time Scaling and Shifting Demonstration

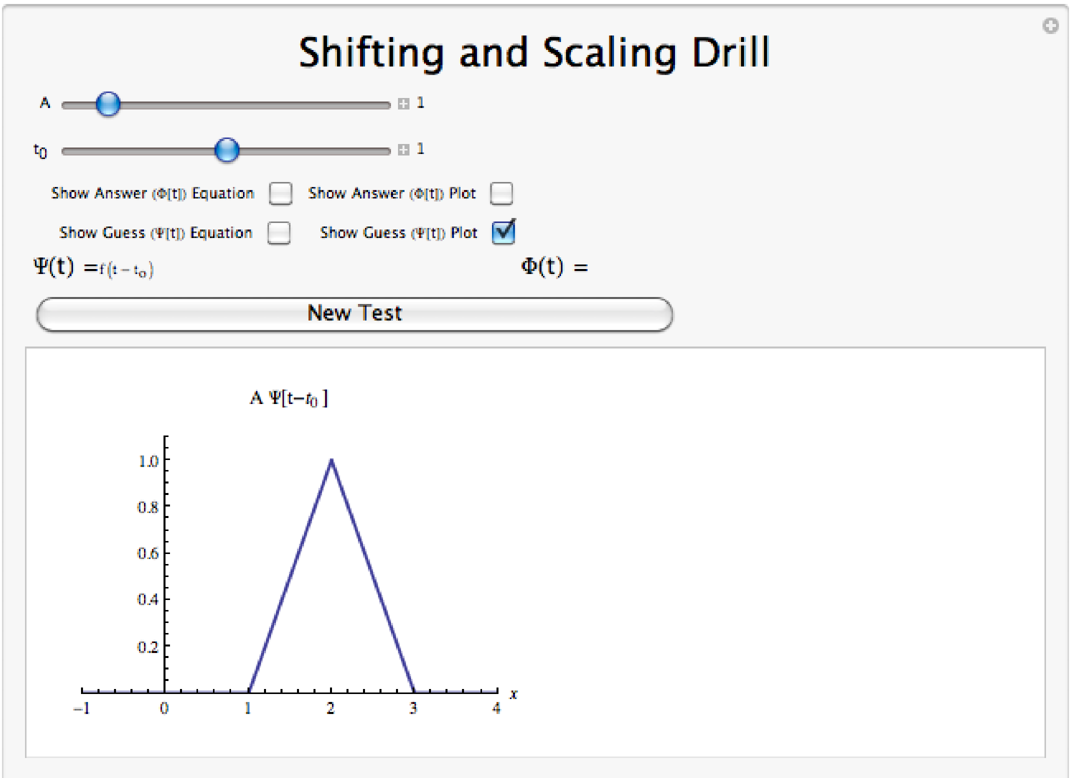


Figure 1.12: Download[5] or Interact (when online) with a Mathematica CDF demonstrating Discrete Harmonic Sinusoids.

1.1.2.4 Signal Operations Summary

Some common operations on signals affect the time parameter of the signal. One of these is time shifting in which a quantity is added to the time parameter in order to advance or delay the signal. Another is the time scaling in which the time parameter is multiplied by a quantity in order to dilate or compress the signal in time. In the event that the quantity involved in the latter operation is negative, time reversal occurs.

1.1.3 Common Continuous Time Signals[6]

1.1.3.1 Introduction

Before looking at this module, hopefully you have an idea of what a signal is and what basic classifications and properties a signal can have. In review, a signal is a function defined with respect to an independent variable. This variable is often time but could represent any number of things. Mathematically, continuous

[5] See the file at <http://cnx.org/content/m10125/latest/TimeshifterDrill_display.cdf>
[6] This content is available online at <http://cnx.org/content/m10058/2.17/>.

time analog signals have continuous independent and dependent variables. This module will describe some useful continuous time analog signals.

1.1.3.2 Important Continuous Time Signals

1.1.3.2.1 Sinusoids

One of the most important elemental signal that you will deal with is the real-valued sinusoid. In its continuous-time form, we write the general expression as

$$A\cos(\omega t + \phi) \tag{1.4}$$

where A is the amplitude, ω is the frequency, and ϕ is the phase. Thus, the period of the sinusoid is

$$T = \frac{2\pi}{\omega} \tag{1.5}$$

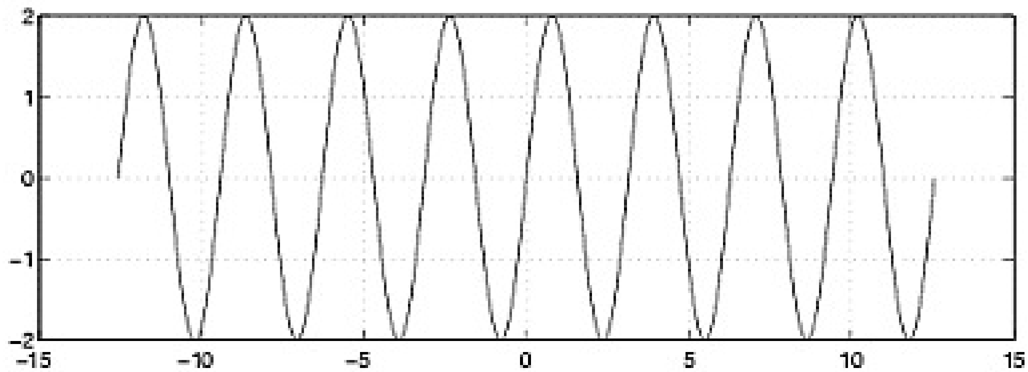


Figure 1.13: Sinusoid with $A = 2$, $w = 2$, and $\phi = 0$.

1.1.3.2.2 Complex Exponentials

As important as the general sinusoid, the **complex exponential** function will become a critical part of your study of signals and systems. Its general continuous form is written as

$$Ae^{st} \tag{1.6}$$

where $s = \sigma + j\omega$ is a complex number in terms of σ, the attenuation constant, and ω the angular frequency.

1.1.3.2.3 Unit Impulses

The unit impulse function, also known as the Dirac delta function, is a signal that has infinite height and infinitesimal width. However, because of the way it is defined, it integrates to one. While this signal is useful for the understanding of many concepts, a formal understanding of its definition more involved. The unit impulse is commonly denoted $\delta(t)$.

More detail is provided in the section on the continuous time impulse function. For now, it suffices to say that this signal is crucially important in the study of continuous signals, as it allows the sifting property to be used in signal representation and signal decomposition.

1.1.3.2.4 Unit Step

Another very basic signal is the **unit-step function** that is defined as

$$u(t) = \begin{cases} 0 & \text{if } t < 0 \\ 1 & \text{if } t \geq 0 \end{cases} \quad (1.7)$$

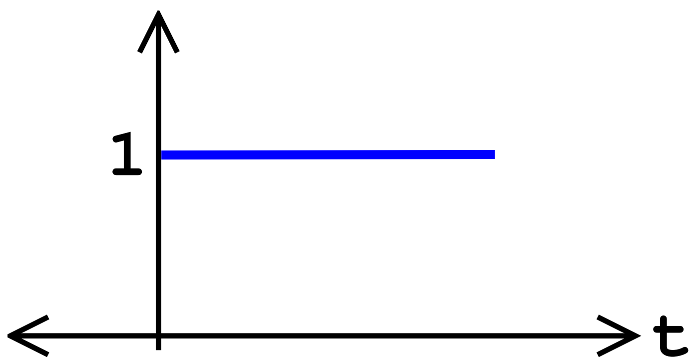


Figure 1.14: Continuous-Time Unit-Step Function

The step function is a useful tool for testing and for defining other signals. For example, when different shifted versions of the step function are multiplied by other signals, one can select a certain portion of the signal and zero out the rest.

1.1.3.3 Common Continuous Time Signals Summary

Some of the most important and most frequently encountered signals have been discussed in this module. There are, of course, many other signals of significant consequence not discussed here. As you will see later, many of the other more complicated signals will be studied in terms of those listed here. Especially take note of the complex exponentials and unit impulse functions, which will be the key focus of several topics included in this course.

1.1.4 Continuous Time Impulse Function[7]

1.1.4.1 Introduction

In engineering, we often deal with the idea of an action occurring at a point. Whether it be a force at a point in space or some other signal at a point in time, it becomes worth while to develop some way of quantitatively defining this. This leads us to the idea of a unit impulse, probably the second most important function, next to the complex exponential, in this systems and signals course.

1.1.4.2 Dirac Delta Function

The **Dirac delta function**, often referred to as the unit impulse or delta function, is the function that defines the idea of a unit impulse in continuous-time. Informally, this function is one that is infinitesimally

[7] This content is available online at <http://cnx.org/content/m10059/2.27/>.

narrow, infinitely tall, yet integrates to one. Perhaps the simplest way to visualize this is as a rectangular pulse from $a-\frac{\epsilon}{2}$ to $a+\frac{\epsilon}{2}$ with a height of $\frac{1}{\epsilon}$. As we take the limit of this setup as ϵ approaches 0, we see that the width tends to zero and the height tends to infinity as the total area remains constant at one. The impulse function is often written as $\delta(t)$.

$$\int_{-\infty}^{\infty} \delta(t)\, dt = 1 \tag{1.8}$$

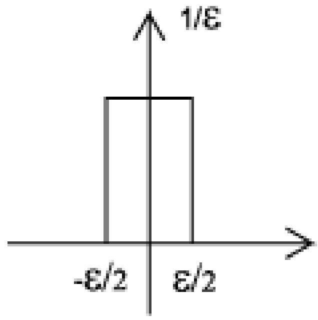


Figure 1.15: This is one way to visualize the Dirac Delta Function.

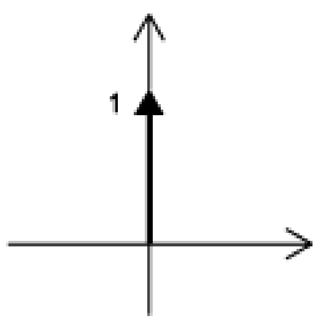


Figure 1.16: Since it is quite difficult to draw something that is infinitely tall, we represent the Dirac with an arrow centered at the point it is applied. If we wish to scale it, we may write the value it is scaled by next to the point of the arrow. This is a unit impulse (no scaling).

Below is a brief list a few important properties of the unit impulse without going into detail of their proofs.

Unit Impulse Properties

- $\delta(\alpha t) = \frac{1}{|\alpha|}\delta(t)$
- $\delta(t) = \delta(-t)$
- $\delta(t) = \frac{d}{dt}u(t)$, where $u(t)$ is the unit step.
- $f(t)\delta(t) = f(0)\delta(t)$

The last of these is especially important as it gives rise to the sifting property of the dirac delta function, which selects the value of a function at a specific time and is especially important in studying the relationship of an operation called convolution to time domain analysis of linear time invariant systems. The sifting property is shown and derived below.

$$\int_{-\infty}^{\infty} f(t)\,\delta(t)\,dt = \int_{-\infty}^{\infty} f(0)\,\delta(t)\,dt = f(0)\int_{-\infty}^{\infty} \delta(t)\,dt = f(0) \tag{1.9}$$

1.1.4.3 Unit Impulse Limiting Demonstration

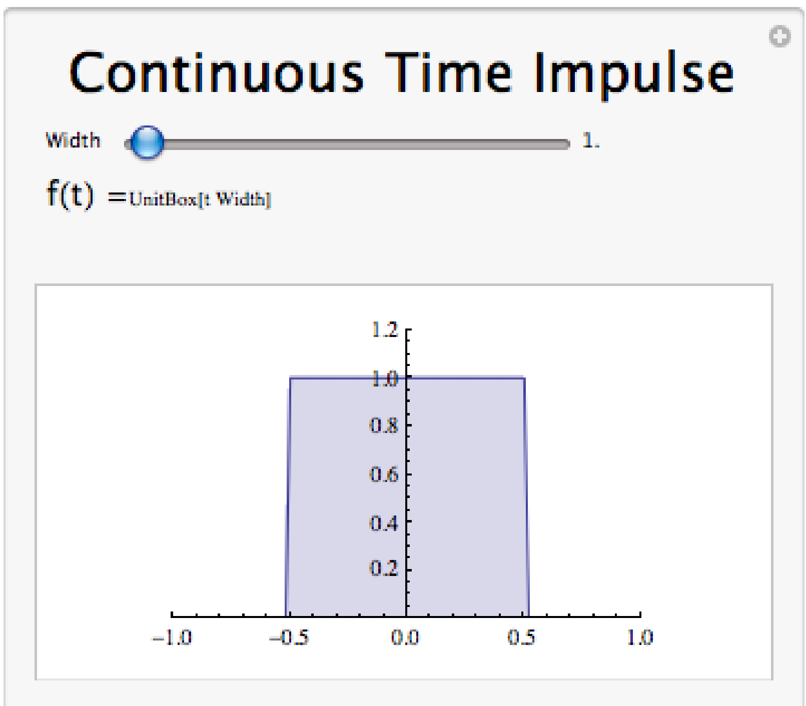


Figure 1.17: Click on the above thumbnail image (when online) to download an interactive Mathematica Player demonstrating the Continuous Time Impulse Function.

1.1.4.4 Continuous Time Unit Impulse Summary

The continuous time unit impulse function, also known as the Dirac delta function, is of great importance to the study of signals and systems. Informally, it is a function with infinite height ant infinitesimal width that integrates to one, which can be viewed as the limiting behavior of a unit area rectangle as it narrows while preserving area. It has several important properties that will appear again when studying systems.

1.1.5 Continuous Time Complex Exponential[8]

1.1.5.1 Introduction

Complex exponentials are some of the most important functions in our study of signals and systems. Their importance stems from their status as eigenfunctions of linear time invariant systems. Before proceeding, you should be familiar with complex numbers.

1.1.5.2 The Continuous Time Complex Exponential

1.1.5.2.1 Complex Exponentials

The complex exponential function will become a critical part of your study of signals and systems. Its general continuous form is written as

$$Ae^{st} \tag{1.10}$$

where $s = \sigma + i\omega$ is a complex number in terms of σ, the attenuation constant, and ω the angular frequency.

1.1.5.2.2 Euler's Formula

The mathematician Euler proved an important identity relating complex exponentials to trigonometric functions. Specifically, he discovered the eponymously named identity, Euler's formula, which states that

$$e^{jx} = cos\left(x\right) + jsin\left(x\right) \tag{1.11}$$

which can be proven as follows.

In order to prove Euler's formula, we start by evaluating the Taylor series for e^z about $z = 0$, which converges for all complex z, at $z = jx$. The result is

$$\begin{array}{ll} e^{jx} & = \sum_{k=0}^{\infty} \frac{(jx)^k}{k!} \\ & = \sum_{k=0}^{\infty} (-1)^k \frac{x^{2k}}{(2k)!} + j \sum_{k=0}^{\infty} (-1)^k \frac{x^{2k+1}}{(2k+1)!} \\ & = cos\left(x\right) + jsin\left(x\right) \end{array} \tag{1.12}$$

because the second expression contains the Taylor series for $cos\left(x\right)$ and $sin\left(x\right)$ about $t = 0$, which converge for all real x. Thus, the desired result is proven.

Choosing $x = \omega t$ this gives the result

$$e^{j\omega t} = cos\left(\omega t\right) + jsin\left(\omega t\right) \tag{1.13}$$

which breaks a continuous time complex exponential into its real part and imaginary part. Using this formula, we can also derive the following relationships.

$$cos\left(\omega t\right) = \frac{1}{2}e^{j\omega t} + \frac{1}{2}e^{-j\omega t} \tag{1.14}$$

$$sin\left(\omega t\right) = \frac{1}{2j}e^{j\omega t} - \frac{1}{2j}e^{-j\omega t} \tag{1.15}$$

[8] This content is available online at <http://cnx.org/content/m10060/2.25/>.

1.1.5.2.3 Continuous Time Phasors

It has been shown how the complex exponential with purely imaginary frequency can be broken up into its real part and its imaginary part. Now consider a general complex frequency $s = \sigma + \omega j$ where σ is the attenuation factor and ω is the frequency. Also consider a phase difference θ. It follows that

$$e^{(\sigma+j\omega)t+j\theta} = e^{\sigma t}\left(cos\left(\omega t+\theta\right)+jsin\left(\omega t+\theta\right)\right). \tag{1.16}$$

Thus, the real and imaginary parts of e^{st} appear below.

$$\text{Re}\{e^{(\sigma+j\omega)t+j\theta}\} = e^{\sigma t}cos\left(\omega t+\theta\right) \tag{1.17}$$

$$\text{Im}\{e^{(\sigma+j\omega)t+j\theta}\} = e^{\sigma t}sin\left(\omega t+\theta\right) \tag{1.18}$$

Using the real or imaginary parts of complex exponential to represent sinusoids with a phase delay multiplied by real exponential is often useful and is called attenuated phasor notation.

We can see that both the real part and the imaginary part have a sinusoid times a real exponential. We also know that sinusoids oscillate between one and negative one. From this it becomes apparent that the real and imaginary parts of the complex exponential will each oscillate within an envelope defined by the real exponential part.

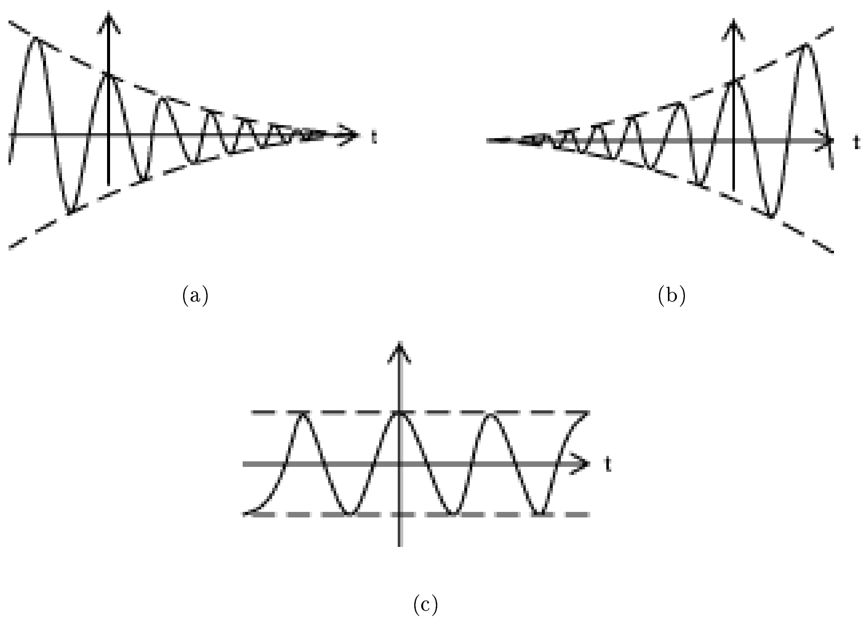


Figure 1.18: The shapes possible for the real part of a complex exponential. Notice that the oscillations are the result of a cosine, as there is a local maximum at $t = 0$. (a) If σ is negative, we have the case of a decaying exponential window. (b) If σ is positive, we have the case of a growing exponential window. (c) If σ is zero, we have the case of a constant window.

1.1.5.3 Complex Exponential Demonstration

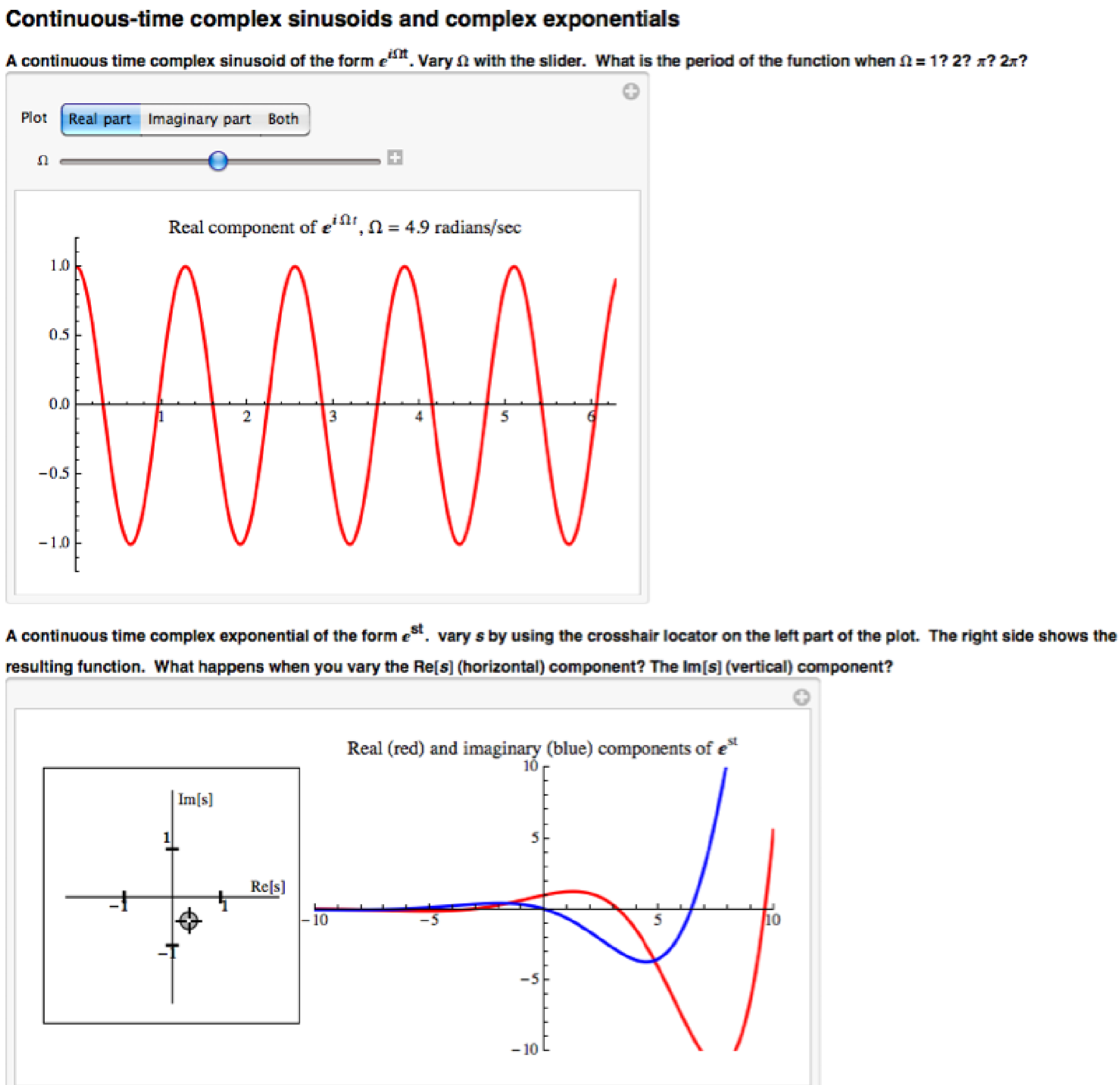


Figure 1.19: Interact (when online) with a Mathematica CDF demonstrating the Continuous Time Complex Exponential. To Download, right-click and save target as .cdf.

1.1.5.4 Continuous Time Complex Exponential Summary

Continuous time complex exponentials are signals of great importance to the study of signals and systems. They can be related to sinusoids through Euler's formula, which identifies the real and imaginary parts of purely imaginary complex exponentials. Eulers formula reveals that, in general, the real and imaginary parts of complex exponentials are sinusoids multiplied by real exponentials. Thus, attenuated phasor notation is often useful in studying these signals.

1.1.6 Discrete-Time Signals[9]

So far, we have treated what are known as **analog** signals and systems. Mathematically, analog signals are functions having continuous quantities as their independent variables, such as space and time. Discrete-time signals[10] are functions defined on the integers; they are sequences. One of the fundamental results of signal theory[11] will detail conditions under which an analog signal can be converted into a discrete-time one and retrieved **without error**. This result is important because discrete-time signals can be manipulated by systems instantiated as computer programs. Subsequent modules describe how virtually all analog signal processing can be performed with software.

As important as such results are, discrete-time signals are more general, encompassing signals derived from analog ones **and** signals that aren't. For example, the characters forming a text file form a sequence, which is also a discrete-time signal. We must deal with such symbolic valued[12] signals and systems as well.

As with analog signals, we seek ways of decomposing real-valued discrete-time signals into simpler components. With this approach leading to a better understanding of signal structure, we can exploit that structure to represent information (create ways of representing information with signals) and to extract information (retrieve the information thus represented). For symbolic-valued signals, the approach is different: We develop a common representation of all symbolic-valued signals so that we can embody the information they contain in a unified way. From an information representation perspective, the most important issue becomes, for both real-valued and symbolic-valued signals, efficiency; What is the most parsimonious and compact way to represent information so that it can be extracted later.

1.1.6.1 Real- and Complex-valued Signals

A discrete-time signal is represented symbolically as $s(n)$, where $n = \{\dots, -1, 0, 1, \dots\}$. We usually draw discrete-time signals as stem plots to emphasize the fact they are functions defined only on the integers. We can delay a discrete-time signal by an integer just as with analog ones. A delayed unit sample has the expression $\delta(n-m)$, and equals one when $n = m$.

Discrete-Time Cosine Signal

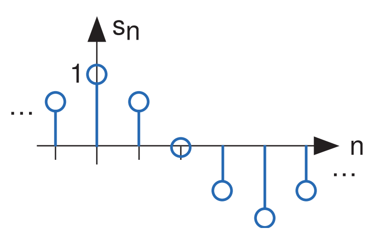


Figure 1.20: The discrete-time cosine signal is plotted as a stem plot. Can you find the formula for this signal?

[9]This content is available online at <http://cnx.org/content/m0009/2.24/>.
[10]"Discrete-Time Signals and Systems" <http://cnx.org/content/m10342/latest/>
[11]"The Sampling Theorem" <http://cnx.org/content/m0050/latest/>
[12]"Discrete-Time Signals and Systems" <http://cnx.org/content/m10342/latest/#para11>

1.1.6.2 Complex Exponentials

The most important signal is, of course, the **complex exponential sequence**.

$$s(n) = e^{i2\pi fn} \quad (1.19)$$

1.1.6.3 Sinusoids

Discrete-time sinusoids have the obvious form $s(n) = A\cos(2\pi fn + \phi)$. As opposed to analog complex exponentials and sinusoids that can have their frequencies be any real value, frequencies of their discrete-time counterparts yield unique waveforms **only** when f lies in the interval $\left(-\frac{1}{2}, \frac{1}{2}\right]$. This property can be easily understood by noting that adding an integer to the frequency of the discrete-time complex exponential has no effect on the signal's value.

$$\begin{aligned} e^{i2\pi(f+m)n} &= e^{i2\pi fn} e^{i2\pi mn} \\ &= e^{i2\pi fn} \end{aligned} \quad (1.20)$$

This derivation follows because the complex exponential evaluated at an integer multiple of 2π equals one.

1.1.6.4 Unit Sample

The second-most important discrete-time signal is the **unit sample**, which is defined to be

$$\delta(n) = \begin{cases} 1 & \text{if } n = 0 \\ 0 & \text{otherwise} \end{cases} \quad (1.21)$$

Unit Sample

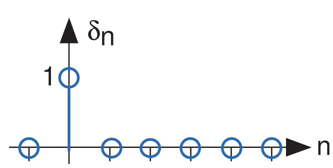


Figure 1.21: The unit sample.

Examination of a discrete-time signal's plot, like that of the cosine signal shown in Figure 1.20 (Discrete-Time Cosine Signal), reveals that all signals consist of a sequence of delayed and scaled unit samples. Because the value of a sequence at each integer m is denoted by $s(m)$ and the unit sample delayed to occur at m is written $\delta(n-m)$, we can decompose **any** signal as a sum of unit samples delayed to the appropriate location and scaled by the signal value.

$$s(n) = \sum_{m=-\infty}^{\infty} s(m)\,\delta(n-m) \quad (1.22)$$

This kind of decomposition is unique to discrete-time signals, and will prove useful subsequently.

Discrete-time systems can act on discrete-time signals in ways similar to those found in analog signals and systems. Because of the role of software in discrete-time systems, many more different systems can be envisioned and "constructed" with programs than can be with analog signals. In fact, a special class of analog signals can be converted into discrete-time signals, processed with software, and converted back into an analog signal, all without the incursion of error. For such signals, systems can be easily produced in software, with equivalent analog realizations difficult, if not impossible, to design.

1.1.6.5 Symbolic-valued Signals

Another interesting aspect of discrete-time signals is that their values do not need to be real numbers. We do have real-valued discrete-time signals like the sinusoid, but we also have signals that denote the sequence of characters typed on the keyboard. Such characters certainly aren't real numbers, and as a collection of possible signal values, they have little mathematical structure other than that they are members of a set. More formally, each element of the symbolic-valued signal $s(n)$ takes on one of the values $\{a_1, \dots, a_K\}$ which comprise the **alphabet** A. This technical terminology does not mean we restrict symbols to being members of the English or Greek alphabet. They could represent keyboard characters, bytes (8-bit quantities), integers that convey daily temperature. Whether controlled by software or not, discrete-time systems are ultimately constructed from digital circuits, which consist **entirely** of analog circuit elements. Furthermore, the transmission and reception of discrete-time signals, like e-mail, is accomplished with analog signals and systems. Understanding how discrete-time and analog signals and systems intertwine is perhaps the main goal of this course.

[Media Object][13]

1.2 Systems

1.2.1 System Classifications and Properties[14]

1.2.1.1 Introduction

In this module some of the basic classifications of systems will be briefly introduced and the most important properties of these systems are explained. As can be seen, the properties of a system provide an easy way to distinguish one system from another. Understanding these basic differences between systems, and their properties, will be a fundamental concept used in all signal and system courses. Once a set of systems can be identified as sharing particular properties, one no longer has to reprove a certain characteristic of a system each time, but it can simply be known due to the the system classification.

1.2.1.2 Classification of Systems

1.2.1.2.1 Continuous vs. Discrete

One of the most important distinctions to understand is the difference between discrete time and continuous time systems. A system in which the input signal and output signal both have continuous domains is said to be a continuous system. One in which the input signal and output signal both have discrete domains is said to be a discrete system. Of course, it is possible to conceive of signals that belong to neither category, such as systems in which sampling of a continuous time signal or reconstruction from a discrete time signal take place.

[13]This media object is a LabVIEW VI. Please view or download it at <SignalApprox.llb>

[14]This content is available online at <http://cnx.org/content/m10084/2.24/>.

1.2.1.2.2 Linear vs. Nonlinear

A linear system is any system that obeys the properties of scaling (first order homogeneity) and superposition (additivity) further described below. A nonlinear system is any system that does not have at least one of these properties.

To show that a system H obeys the scaling property is to show that

$$H\left(kf\left(t\right)\right)=kH\left(f\left(t\right)\right) \tag{1.23}$$

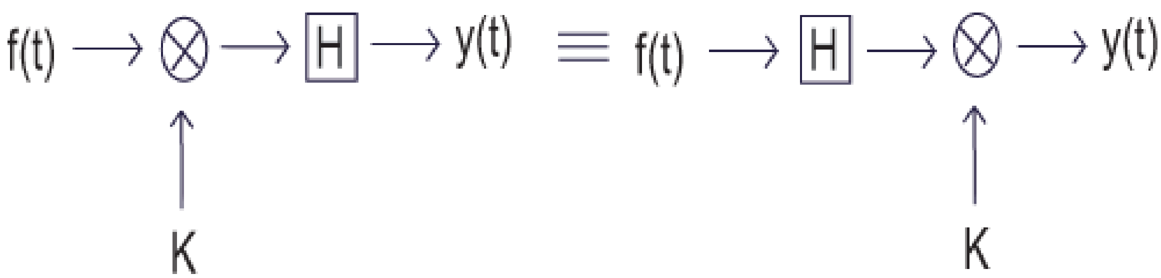


Figure 1.22: A block diagram demonstrating the scaling property of linearity

To demonstrate that a system H obeys the superposition property of linearity is to show that

$$H\left(f_1\left(t\right)+f_2\left(t\right)\right)=H\left(f_1\left(t\right)\right)+H\left(f_2\left(t\right)\right) \tag{1.24}$$

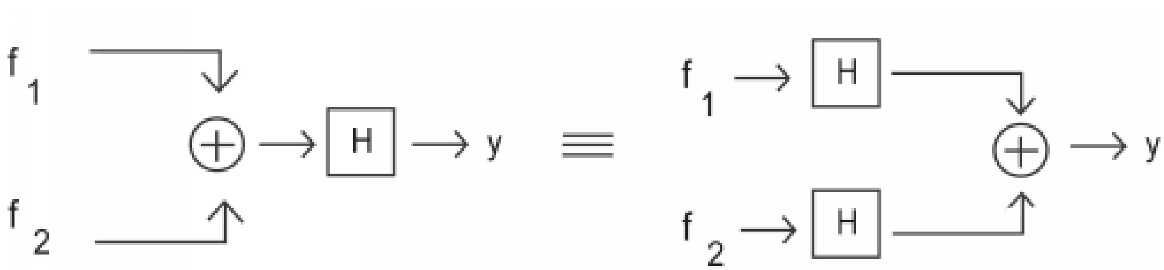


Figure 1.23: A block diagram demonstrating the superposition property of linearity

It is possible to check a system for linearity in a single (though larger) step. To do this, simply combine the first two steps to get

$$H\left(k_1f_1\left(t\right)+k_2f_2\left(t\right)\right)=k_1H\left(f_1\left(t\right)\right)+k_2H\left(f_2\left(t\right)\right) \tag{1.25}$$

1.2.1.2.3 Time Invariant vs. Time Varying

A system is said to be time invariant if it commutes with the parameter shift operator defined by $S_T(f(t)) = f(t-T)$ for all T, which is to say

$$HS_T = S_T H \tag{1.26}$$

for all real T. Intuitively, that means that for any input function that produces some output function, any time shift of that input function will produce an output function identical in every way except that it is shifted by the same amount. Any system that does not have this property is said to be time varying.

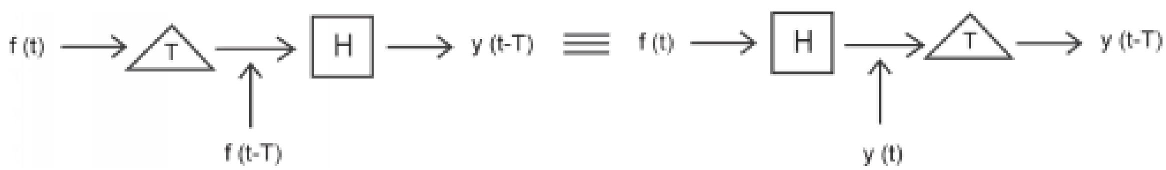


Figure 1.24: This block diagram shows what the condition for time invariance. The output is the same whether the delay is put on the input or the output.

1.2.1.2.4 Causal vs. Noncausal

A causal system is one in which the output depends only on current or past inputs, but not future inputs. Similarly, an anticausal system is one in which the output depends only on current or future inputs, but not past inputs. Finally, a noncausal system is one in which the output depends on both past and future inputs. All "realtime" systems must be causal, since they can not have future inputs available to them.

One may think the idea of future inputs does not seem to make much physical sense; however, we have only been dealing with time as our dependent variable so far, which is not always the case. Imagine rather that we wanted to do image processing. Then the dependent variable might represent pixel positions to the left and right (the "future") of the current position on the image, and we would not necessarily have a causal system.

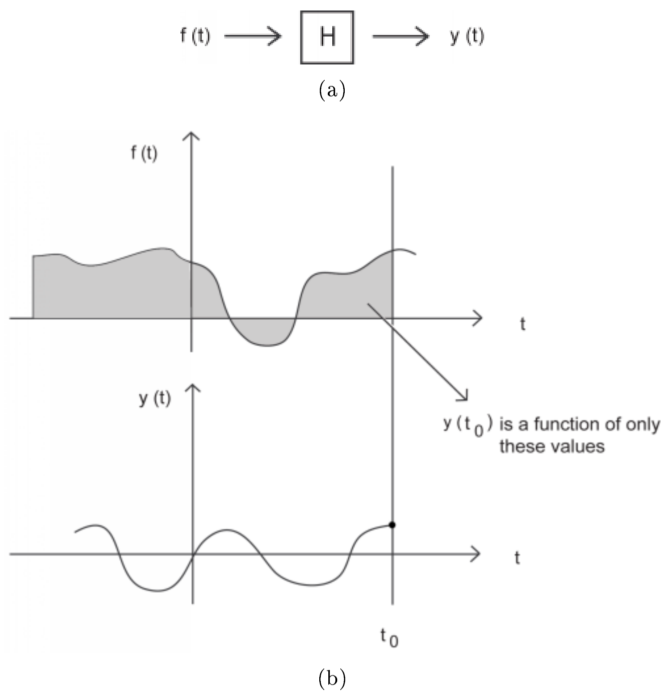


Figure 1.25: (a) For a typical system to be causal... (b) ...the output at time t_0, $y(t_0)$, can only depend on the portion of the input signal before t_0.

1.2.1.2.5 Stable vs. Unstable

There are several definitions of stability, but the one that will be used most frequently in this course will be bounded input, bounded output (BIBO) stability. In this context, a stable system is one in which the output is bounded if the input is also bounded. Similarly, an unstable system is one in which at least one bounded input produces an unbounded output.

Representing this mathematically, a stable system must have the following property, where $x(t)$ is the input and $y(t)$ is the output. The output must satisfy the condition

$$|y(t)| \leq M_y < \infty \tag{1.27}$$

whenever we have an input to the system that satisfies

$$|x(t)| \leq M_x < \infty \tag{1.28}$$

M_x and M_y both represent a set of finite positive numbers and these relationships hold for all of t. Otherwise, the system is unstable.

1.2.1.3 System Classifications Summary

This module describes just some of the many ways in which systems can be classified. Systems can be continuous time, discrete time, or neither. They can be linear or nonlinear, time invariant or time varying, and stable or unstable. We can also divide them based on their causality properties. There are other ways to classify systems, such as use of memory, that are not discussed here but will be described in subsequent modules.

1.2.2 Linear Time Invariant Systems[15]

1.2.2.1 Introduction

Linearity and time invariance are two system properties that greatly simplify the study of systems that exhibit them. In our study of signals and systems, we will be especially interested in systems that demonstrate both of these properties, which together allow the use of some of the most powerful tools of signal processing.

1.2.2.2 Linear Time Invariant Systems

1.2.2.2.1 Linear Systems

If a system is linear, this means that when an input to a given system is scaled by a value, the output of the system is scaled by the same amount.

Linear Scaling

(a) (b)

Figure 1.26

In Figure 1.26(a) above, an input x to the linear system L gives the output y. If x is scaled by a value α and passed through this same system, as in Figure 1.26(b), the output will also be scaled by α.

A linear system also obeys the principle of superposition. This means that if two inputs are added together and passed through a linear system, the output will be the sum of the individual inputs' outputs.

(a) (b)

Figure 1.27

[15] This content is available online at <http://cnx.org/content/m2102/2.26/>.

Superposition Principle

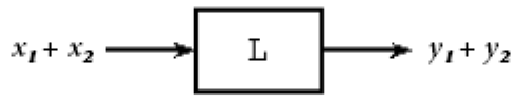


Figure 1.28: If Figure 1.27 is true, then the principle of superposition says that Figure 1.28 (Superposition Principle) is true as well. This holds for linear systems.

That is, if Figure 1.27 is true, then Figure 1.28 (Superposition Principle) is also true for a linear system. The scaling property mentioned above still holds in conjunction with the superposition principle. Therefore, if the inputs x and y are scaled by factors α and β, respectively, then the sum of these scaled inputs will give the sum of the individual scaled outputs:

(a)

(b)

Figure 1.29

Superposition Principle with Linear Scaling

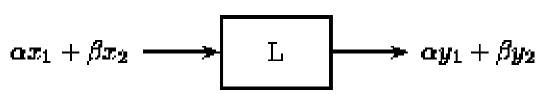


Figure 1.30: Given Figure 1.29 for a linear system, Figure 1.30 (Superposition Principle with Linear Scaling) holds as well.

Example 1.4
Consider the system H_1 in which

$$H_1 (f (t)) = t f (t) \tag{1.29}$$

for all signals f. Given any two signals f, g and scalars a, b

$$H_1 (a f (t) + b g (t)) = t (a f (t) + b g (t)) = a t f (t) + b t g (t) = a H_1 (f (t)) + b H_1 (g (t)) \tag{1.30}$$

for all real t. Thus, H_1 is a linear system.

Example 1.5
Consider the system H_2 in which

$$H_2 (f (t)) = (f (t))^2 \tag{1.31}$$

for all signals f. Because

$$H_2 (2t) = 4t^2 \neq 2t^2 = 2H_2 (t) \tag{1.32}$$

for nonzero t, H_2 is not a linear system.

1.2.2.2.2 Time Invariant Systems

A time-invariant system has the property that a certain input will always give the same output (up to timing), without regard to when the input was applied to the system.

Time-Invariant Systems

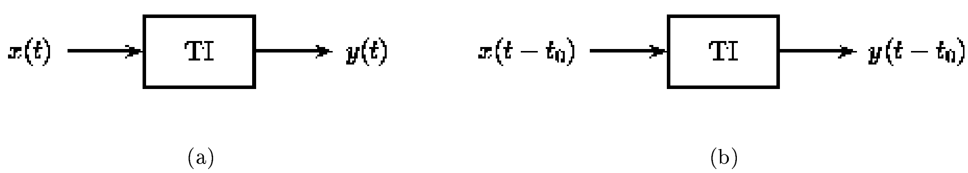


Figure 1.31: Figure 1.31(a) shows an input at time t while Figure 1.31(b) shows the same input t_0 seconds later. In a time-invariant system both outputs would be identical except that the one in Figure 1.31(b) would be delayed by t_0.

In this figure, $x (t)$ and $x (t - t_0)$ are passed through the system TI. Because the system TI is time-invariant, the inputs $x (t)$ and $x (t - t_0)$ produce the same output. The only difference is that the output due to $x (t - t_0)$ is shifted by a time t_0.

Whether a system is time-invariant or time-varying can be seen in the differential equation (or difference equation) describing it. **Time-invariant systems are modeled with constant coefficient equations**. A constant coefficient differential (or difference) equation means that the parameters of the system are **not** changing over time and an input now will give the same result as the same input later.

Example 1.6
Consider the system H_1 in which

$$H_1 (f (t)) = t f (t) \tag{1.33}$$

for all signals f. Because

$$S_T (H_1 (f (t))) = S_T (t f (t)) = (t - T) f (t - T) \neq t f (t - T) = H_1 (f (t - T)) = H_1 (S_T (f (t))) \tag{1.34}$$

for nonzero T, H_1 is not a time invariant system.

Example 1.7
Consider the system H_2 in which

$$H_2 (f (t)) = (f (t))^2 \tag{1.35}$$

for all signals f. For all real T and signals f,

$$S_T (H_2 (f (t))) = S_T \left(f(t)^2\right) = (f (t - T))^2 = H_2 (f (t - T)) = H_2 (S_T (f (t))) \tag{1.36}$$

for all real t. Thus, H_2 is a time invariant system.

1.2.2.2.3 Linear Time Invariant Systems

Certain systems are both linear and time-invariant, and are thus referred to as LTI systems.

Linear Time-Invariant Systems

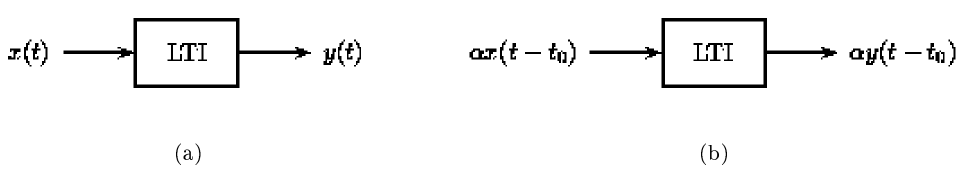
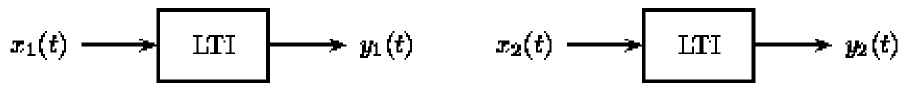


Figure 1.32: This is a combination of the two cases above. Since the input to Figure 1.32(b) is a scaled, time-shifted version of the input in Figure 1.32(a), so is the output.

As LTI systems are a subset of linear systems, they obey the principle of superposition. In the figure below, we see the effect of applying time-invariance to the superposition definition in the linear systems section above.



(a) (b)

Figure 1.33

Superposition in Linear Time-Invariant Systems

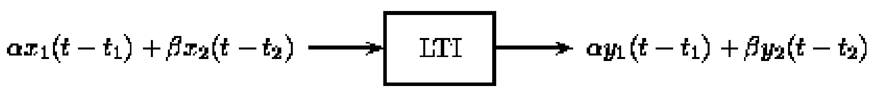
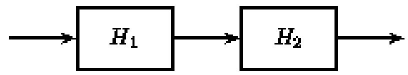


Figure 1.34: The principle of superposition applied to LTI systems

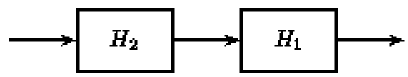
1.2.2.2.3.1 LTI Systems in Series

If two or more LTI systems are in series with each other, their order can be interchanged without affecting the overall output of the system. Systems in series are also called cascaded systems.

Cascaded LTI Systems



(a)



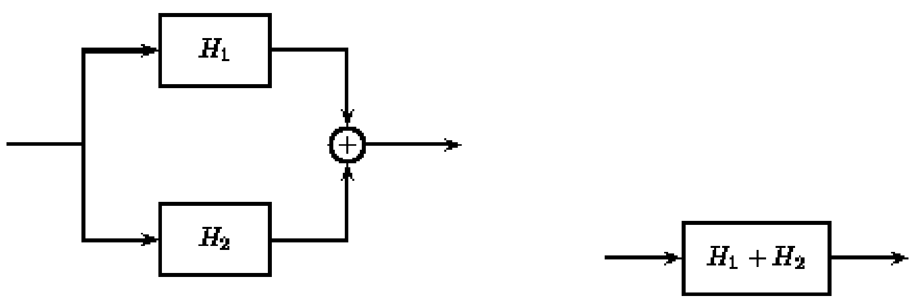
(b)

Figure 1.35: The order of cascaded LTI systems can be interchanged without changing the overall effect.

1.2.2.2.3.2 LTI Systems in Parallel

If two or more LTI systems are in parallel with one another, an equivalent system is one that is defined as the sum of these individual systems.

Parallel LTI Systems



(a) (b)

Figure 1.36: Parallel systems can be condensed into the sum of systems.

Example 1.8
Consider the system H_3 in which

$$H_3 (f (t)) = 2f (t) \tag{1.37}$$

for all signals f. Given any two signals f, g and scalars a, b

$$H_3 (af (t) + bg (t)) = 2 (af (t) + bg (t)) = a2f (t) + b2g (t) = aH_3 (f (t)) + bH_3 (g (t)) \tag{1.38}$$

for all real t. Thus, H_3 is a linear system. For all real T and signals f,

$$S_T (H_3 (f (t))) = S_T (2f (t)) = 2f (t - T) = H_3 (f (t - T)) = H_3 (S_T (f (t))) \tag{1.39}$$

for all real t. Thus, H_3 is a time invariant system. Therefore, H_3 is a linear time invariant system.

Example 1.9
As has been previously shown, each of the following systems are not linear or not time invariant.

$$H_1 (f (t)) = tf (t) \tag{1.40}$$

$$H_2 (f (t)) = (f (t))^2 \tag{1.41}$$

Thus, they are not linear time invariant systems.

1.2.2.3 Linear Time Invariant Demonstration

Figure 1.37: Interact(when online) with the Mathematica CDF above demonstrating Linear Time Invariant systems. To download, right click and save file as .cdf.

1.2.2.4 LTI Systems Summary

Two very important and useful properties of systems have just been described in detail. The first of these, linearity, allows us the knowledge that a sum of input signals produces an output signal that is the summed original output signals and that a scaled input signal produces an output signal scaled from the original output signal. The second of these, time invariance, ensures that time shifts commute with application of the system. In other words, the output signal for a time shifted input is the same as the output signal for the original input signal, except for an identical shift in time. Systems that demonstrate both linearity and time invariance, which are given the acronym LTI systems, are particularly simple to study as these properties allow us to leverage some of the most powerful tools in signal processing.

1.3 Time Domain Analysis of Continuous Time Systems

1.3.1 Continuous Time Convolution[16]

1.3.1.1 Introduction

Convolution, one of the most important concepts in electrical engineering, can be used to determine the output a system produces for a given input signal. It can be shown that a linear time invariant system is completely characterized by its impulse response. The sifting property of the continuous time impulse function tells us that the input signal to a system can be represented as an integral of scaled and shifted impulses and, therefore, as the limit of a sum of scaled and shifted approximate unit impulses. Thus, by linearity, it would seem reasonable to compute of the output signal as the limit of a sum of scaled and shifted unit impulse responses and, therefore, as the integral of a scaled and shifted impulse response. That is exactly what the operation of convolution accomplishes. Hence, convolution can be used to determine a linear time invariant system's output from knowledge of the input and the impulse response.

1.3.1.2 Convolution and Circular Convolution

1.3.1.2.1 Convolution

1.3.1.2.1.1 Operation Definition

Continuous time convolution is an operation on two continuous time signals defined by the integral

$$(f * g)(t) = \int_{-\infty}^{\infty} f(\tau)\, g(t-\tau)\, d\tau \qquad (1.42)$$

for all signals f, g defined on $\mathbb{R}$. It is important to note that the operation of convolution is commutative, meaning that

$$f * g = g * f \qquad (1.43)$$

for all signals f, g defined on $\mathbb{R}$. Thus, the convolution operation could have been just as easily stated using the equivalent definition

$$(f * g)(t) = \int_{-\infty}^{\infty} f(t-\tau)\, g(\tau)\, d\tau \qquad (1.44)$$

for all signals f, g defined on $\mathbb{R}$. Convolution has several other important properties not listed here but explained and derived in a later module.

1.3.1.2.1.2 Definition Motivation

The above operation definition has been chosen to be particularly useful in the study of linear time invariant systems. In order to see this, consider a linear time invariant system H with unit impulse response h. Given a system input signal x we would like to compute the system output signal $H(x)$. First, we note that the input can be expressed as the convolution

$$x(t) = \int_{-\infty}^{\infty} x(\tau)\, \delta(t-\tau)\, d\tau \qquad (1.45)$$

by the sifting property of the unit impulse function. Writing this integral as the limit of a summation,

$$x(t) = \lim_{\Delta \to 0} \sum_n x(n\Delta)\, \delta_\Delta(t - n\Delta)\, \Delta \qquad (1.46)$$

[16] This content is available online at <http://cnx.org/content/m10085/2.35/>.

where

$$\delta_\Delta(t) = \{ \begin{array}{cc} 1/\Delta & 0 \leq t < \Delta \\ 0 & \text{otherwise} \end{array} \tag{1.47}$$

approximates the properties of $\delta(t)$. By linearity

$$Hx(t) = \lim_{\Delta \to 0} \sum_n x(n\Delta) H\delta_\Delta(t - n\Delta)\Delta \tag{1.48}$$

which evaluated as an integral gives

$$Hx(t) = \int_{-\infty}^{\infty} x(\tau) H\delta(t-\tau) d\tau. \tag{1.49}$$

Since $H\delta(t-\tau)$ is the shifted unit impulse response $h(t-\tau)$, this gives the result

$$Hx(t) = \int_{-\infty}^{\infty} x(\tau) h(t-\tau) d\tau = (x * h)(t). \tag{1.50}$$

Hence, convolution has been defined such that the output of a linear time invariant system is given by the convolution of the system input with the system unit impulse response.

1.3.1.2.1.3 Graphical Intuition

It is often helpful to be able to visualize the computation of a convolution in terms of graphical processes. Consider the convolution of two functions f, g given by

$$(f * g)(t) = \int_{-\infty}^{\infty} f(\tau) g(t-\tau) d\tau = \int_{-\infty}^{\infty} f(t-\tau) g(\tau) d\tau. \tag{1.51}$$

The first step in graphically understanding the operation of convolution is to plot each of the functions. Next, one of the functions must be selected, and its plot reflected across the $\tau = 0$ axis. For each real t, that same function must be shifted left by t. The product of the two resulting plots is then constructed. Finally, the area under the resulting curve is computed.

Example 1.10

Recall that the impulse response for the capacitor voltage in a series RC circuit is given by

$$h(t) = \frac{1}{RC} e^{-t/RC} u(t), \tag{1.52}$$

and consider the response to the input voltage

$$x(t) = u(t). \tag{1.53}$$

We know that the output for this input voltage is given by the convolution of the impulse response with the input signal

$$y(t) = x(t) * h(t). \tag{1.54}$$

We would like to compute this operation by beginning in a way that minimizes the algebraic complexity of the expression. Thus, since $x(t) = u(t)$ is the simpler of the two signals, it is desirable to select it for time reversal and shifting. Thus, we would like to compute

$$y(t) = \int_{-\infty}^{\infty} \frac{1}{RC} e^{-\tau/RC} u(\tau) u(t-\tau) d\tau. \tag{1.55}$$

The step functions can be used to further simplify this integral by narrowing the region of integration to the nonzero region of the integrand. Therefore,

$$y(t) = \int_0^{max\{0,t\}} \frac{1}{RC} e^{-\tau/RC} d\tau. \tag{1.56}$$

Hence, the output is

$$y(t) = \{ \begin{array}{cc} 0 & t \leq 0 \\ 1 - e^{-t/RC} & t > 0 \end{array} \tag{1.57}$$

which can also be written as

$$y(t) = \left(1 - e^{-t/RC}\right) u(t). \tag{1.58}$$

1.3.1.2.2 Circular Convolution

Continuous time circular convolution is an operation on two finite length or periodic continuous time signals defined by the integral

$$(f * g)(t) = \int_0^T \hat{f}(\tau) \hat{g}(t - \tau) d\tau \tag{1.59}$$

for all signals f, g defined on $\mathbb{R}[0, T]$ where $\hat{f}, \hat{g}$ are periodic extensions of f and g. It is important to note that the operation of circular convolution is commutative, meaning that

$$f * g = g * f \tag{1.60}$$

for all signals f, g defined on $\mathbb{R}[0, T]$. Thus, the circular convolution operation could have been just as easily stated using the equivalent definition

$$(f * g)(t) = \int_0^T \hat{f}(t - \tau) \hat{g}(\tau) d\tau \tag{1.61}$$

for all signals f, g defined on $\mathbb{R}[0, T]$ where $\hat{f}, \hat{g}$ are periodic extensions of f and g. Circular convolution has several other important properties not listed here but explained and derived in a later module.

Alternatively, continuous time circular convolution can be expressed as the sum of two integrals given by

$$(f * g)(t) = \int_0^t f(\tau) g(t - \tau) d\tau + \int_t^T f(\tau) g(t - \tau + T) d\tau \tag{1.62}$$

for all signals f, g defined on $\mathbb{R}[0, T]$.

Meaningful examples of computing continuous time circular convolutions in the time domain would involve complicated algebraic manipulations dealing with the wrap around behavior, which would ultimately be more confusing than helpful. Thus, none will be provided in this section. However, continuous time circular convolutions are more easily computed using frequency domain tools as will be shown in the continuous time Fourier series section.

1.3.1.2.2.1 Definition Motivation

The above operation definition has been chosen to be particularly useful in the study of linear time invariant systems. In order to see this, consider a linear time invariant system H with unit impulse response h. Given a finite or periodic system input signal x we would like to compute the system output signal $H(x)$. First, we note that the input can be expressed as the circular convolution

$$x(t) = \int_0^T \hat{x}(\tau)\,\hat{\delta}(t-\tau)\,d\tau \tag{1.63}$$

by the sifting property of the unit impulse function. Writing this integral as the limit of a summation,

$$x(t) = \lim_{\Delta\to 0} \sum_n \hat{x}(n\Delta)\,\hat{\delta}_\Delta(t-n\Delta)\,\Delta \tag{1.64}$$

where

$$\delta_\Delta(t) = \begin{cases} 1/\Delta & 0 \le t < \Delta \\ 0 & \text{otherwise} \end{cases} \tag{1.65}$$

approximates the properties of $\delta(t)$. By linearity

$$Hx(t) = \lim_{\Delta\to 0} \sum_n \hat{x}(n\Delta)\,H\hat{\delta}_\Delta(t-n\Delta)\,\Delta \tag{1.66}$$

which evaluated as an integral gives

$$Hx(t) = \int_0^T \hat{x}(\tau)\,H\,\hat{\delta}(t-\tau)\,d\tau. \tag{1.67}$$

Since $H\delta(t-\tau)$ is the shifted unit impulse response $h(t-\tau)$, this gives the result

$$Hx(t) = \int_0^T \hat{x}(\tau)\,\hat{h}(t-\tau)\,d\tau = (x*h)(t). \tag{1.68}$$

Hence, circular convolution has been defined such that the output of a linear time invariant system is given by the convolution of the system input with the system unit impulse response.

1.3.1.2.2.2 Graphical Intuition

It is often helpful to be able to visualize the computation of a circular convolution in terms of graphical processes. Consider the circular convolution of two finite length functions f, g given by

$$(f*g)(t) = \int_0^T \hat{f}(\tau)\,\hat{g}(t-\tau)\,d\tau = \int_0^T \hat{f}(t-\tau)\,\hat{g}(\tau)\,d\tau. \tag{1.69}$$

The first step in graphically understanding the operation of convolution is to plot each of the periodic extensions of the functions. Next, one of the functions must be selected, and its plot reflected across the $\tau = 0$ axis. For each $t \in \mathbb{R}[0,T]$, that same function must be shifted left by t. The product of the two resulting plots is then constructed. Finally, the area under the resulting curve on $\mathbb{R}[0,T]$ is computed.

1.3.1.3 Convolution Demonstration

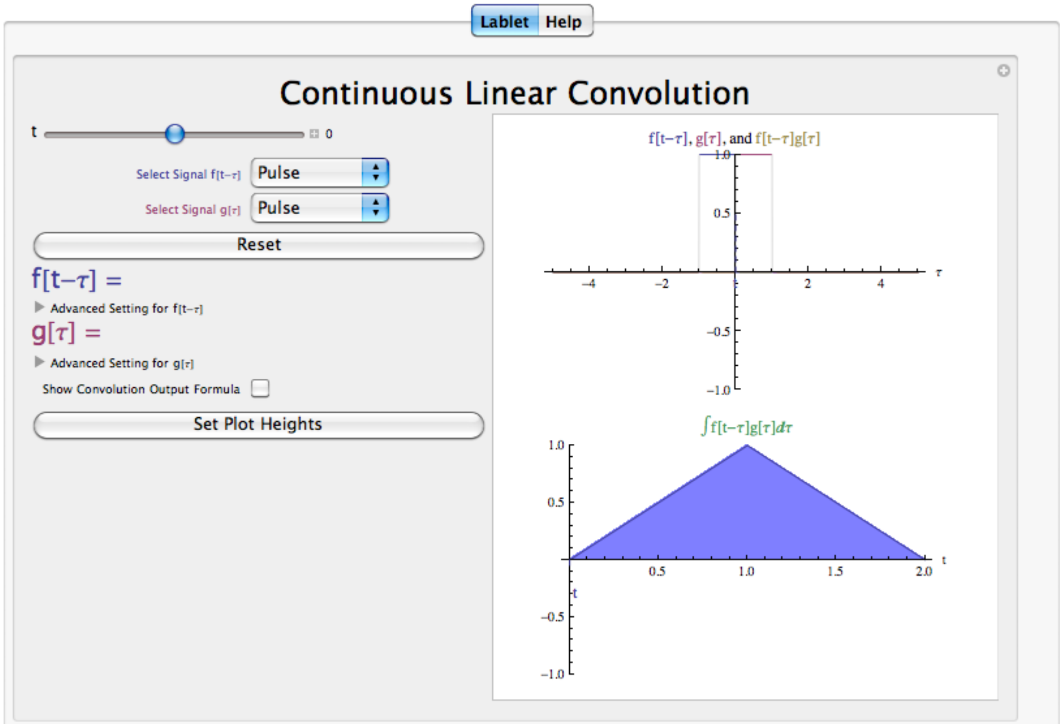


Figure 1.38: Interact (when online) with a Mathematica CDF demonstrating Convolution. To Download, right-click and save target as .cdf.

1.3.1.4 Convolution Summary

Convolution, one of the most important concepts in electrical engineering, can be used to determine the output signal of a linear time invariant system for a given input signal with knowledge of the system's unit impulse response. The operation of continuous time convolution is defined such that it performs this function for infinite length continuous time signals and systems. The operation of continuous time circular convolution is defined such that it performs this function for finite length and periodic continuous time signals. In each case, the output of the system is the convolution or circular convolution of the input signal with the unit impulse response.

1.3.2 Properties of Continuous Time Convolution[17]

1.3.2.1 Introduction

We have already shown the important role that continuous time convolution plays in signal processing. This section provides discussion and proof of some of the important properties of continuous time convolution. Analogous properties can be shown for continuous time circular convolution with trivial modification of the proofs provided except where explicitly noted otherwise.

[17] This content is available online at <http://cnx.org/content/m10088/2.21/>.

1.3.2.2 Continuous Time Convolution Properties

1.3.2.2.1 Associativity

The operation of convolution is associative. That is, for all continuous time signals x_1, x_2, x_3 the following relationship holds.

$$x_1 * (x_2 * x_3) = (x_1 * x_2) * x_3 \tag{1.70}$$

In order to show this, note that

$$\begin{aligned} (x_1 * (x_2 * x_3))(t) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x_1(\tau_1)\, x_2(\tau_2)\, x_3((t-\tau_1)-\tau_2)\, d\tau_2 d\tau_1 \\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x_1(\tau_1)\, x_2((\tau_1+\tau_2)-\tau_1)\, x_3(t-(\tau_1+\tau_2))\, d\tau_2 d\tau_1 \\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x_1(\tau_1)\, x_2(\tau_3-\tau_1)\, x_3(t-\tau_3)\, d\tau_1 d\tau_3 \\ &= ((x_1 * x_2) * x_3)(t) \end{aligned} \tag{1.71}$$

proving the relationship as desired through the substitution $\tau_3 = \tau_1 + \tau_2$.

1.3.2.2.2 Commutativity

The operation of convolution is commutative. That is, for all continuous time signals x_1, x_2 the following relationship holds.

$$x_1 * x_2 = x_2 * x_1 \tag{1.72}$$

In order to show this, note that

$$\begin{aligned} (x_1 * x_2)(t) &= \int_{-\infty}^{\infty} x_1(\tau_1)\, x_2(t-\tau_1)\, d\tau_1 \\ &= \int_{-\infty}^{\infty} x_1(t-\tau_2)\, x_2(\tau_2)\, d\tau_2 \\ &= (x_2 * x_1)(t) \end{aligned} \tag{1.73}$$

proving the relationship as desired through the substitution $\tau_2 = t - \tau_1$.

1.3.2.2.3 Distributivity

The operation of convolution is distributive over the operation of addition. That is, for all continuous time signals x_1, x_2, x_3 the following relationship holds.

$$x_1 * (x_2 + x_3) = x_1 * x_2 + x_1 * x_3 \tag{1.74}$$

In order to show this, note that

$$\begin{aligned} (x_1 * (x_2 + x_3))(t) &= \int_{-\infty}^{\infty} x_1(\tau)\,(x_2(t-\tau) + x_3(t-\tau))\, d\tau \\ &= \int_{-\infty}^{\infty} x_1(\tau)\, x_2(t-\tau)\, d\tau + \int_{-\infty}^{\infty} x_1(\tau)\, x_3(t-\tau)\, d\tau \\ &= (x_1 * x_2 + x_1 * x_3)(t) \end{aligned} \tag{1.75}$$

proving the relationship as desired.

1.3.2.2.4 Multilinearity

The operation of convolution is linear in each of the two function variables. Additivity in each variable results from distributivity of convolution over addition. Homogenity of order one in each variable results from the fact that for all continuous time signals x_1, x_2 and scalars a the following relationship holds.

$$a(x_1 * x_2) = (ax_1) * x_2 = x_1 * (ax_2) \tag{1.76}$$

In order to show this, note that

$$\begin{aligned} (a(x_1 * x_2))(t) &= a \int_{-\infty}^{\infty} x_1(\tau) x_2(t-\tau)\, d\tau \\ &= \int_{-\infty}^{\infty} (ax_1(\tau)) x_2(t-\tau)\, d\tau \\ &= ((ax_1) * x_2)(t) \\ &= \int_{-\infty}^{\infty} x_1(\tau)(ax_2(t-\tau))\, d\tau \\ &= (x_1 * (ax_2))(t) \end{aligned} \tag{1.77}$$

proving the relationship as desired.

1.3.2.2.5 Conjugation

The operation of convolution has the following property for all continuous time signals x_1, x_2.

$$\overline{x_1 * x_2} = \overline{x_1} * \overline{x_2} \tag{1.78}$$

In order to show this, note that

$$\begin{aligned} (\overline{x_1 * x_2})(t) &= \overline{\int_{-\infty}^{\infty} x_1(\tau) x_2(t-\tau)\, d\tau} \\ &= \int_{-\infty}^{\infty} \overline{x_1(\tau) x_2(t-\tau)} d\tau \\ &= \int_{-\infty}^{\infty} \overline{x_1}(\tau) \overline{x_2}(t-\tau)\, d\tau \\ &= (\overline{x_1} * \overline{x_2})(t) \end{aligned} \tag{1.79}$$

proving the relationship as desired.

1.3.2.2.6 Time Shift

The operation of convolution has the following property for all continuous time signals x_1, x_2 where S_T is the time shift operator.

$$S_T(x_1 * x_2) = (S_T x_1) * x_2 = x_1 * (S_T x_2) \tag{1.80}$$

In order to show this, note that

$$\begin{aligned} S_T(x_1 * x_2)(t) &= \int_{-\infty}^{\infty} x_2(\tau) x_1((t-T)-\tau)\, d\tau \\ &= \int_{-\infty}^{\infty} x_2(\tau) S_T x_1(t-\tau)\, d\tau \\ &= ((S_T x_1) * x_2)(t) \\ &= \int_{-\infty}^{\infty} x_1(\tau) x_2((t-T)-\tau)\, d\tau \\ &= \int_{-\infty}^{\infty} x_1(\tau) S_T x_2(t-\tau)\, d\tau \\ &= x_1 * (S_T x_2)(t) \end{aligned} \tag{1.81}$$

proving the relationship as desired.

1.3.2.2.7 Differentiation

The operation of convolution has the following property for all continuous time signals x_1, x_2.

$$\frac{d}{dt}(x_1 * x_2)(t) = \left(\frac{dx_1}{dt} * x_2\right)(t) = \left(x_1 * \frac{dx_2}{dt}\right)(t) \tag{1.82}$$

In order to show this, note that

$$\begin{aligned} \frac{d}{dt}(x_1 * x_2)(t) &= \int_{-\infty}^{\infty} x_2(\tau) \frac{d}{dt} x_1(t-\tau)\, d\tau \\ &= \left(\frac{dx_1}{dt} * x_2\right)(t) \\ &= \int_{-\infty}^{\infty} x_1(\tau) \frac{d}{dt} x_2(t-\tau)\, d\tau \\ &= \left(x_1 * \frac{dx_2}{dt}\right)(t) \end{aligned} \tag{1.83}$$

proving the relationship as desired.

1.3.2.2.8 Impulse Convolution

The operation of convolution has the following property for all continuous time signals x where δ is the Dirac delta funciton.

$$x * \delta = x \tag{1.84}$$

In order to show this, note that

$$\begin{aligned} (x * \delta)(t) &= \int_{-\infty}^{\infty} x(\tau)\, \delta(t-\tau)\, d\tau \\ &= x(t) \int_{-\infty}^{\infty} \delta(t-\tau)\, d\tau \\ &= x(t) \end{aligned} \tag{1.85}$$

proving the relationship as desired.

1.3.2.2.9 Width

The operation of convolution has the following property for all continuous time signals x_1, x_2 where $Duration(x)$ gives the duration of a signal x.

$$Duration(x_1 * x_2) = Duration(x_1) + Duration(x_2) \tag{1.86}$$

. In order to show this informally, note that $(x_1 * x_2)(t)$ is nonzero for all t for which there is a τ such that $x_1(\tau)\, x_2(t-\tau)$ is nonzero. When viewing one function as reversed and sliding past the other, it is easy to see that such a τ exists for all t on an interval of length $Duration(x_1) + Duration(x_2)$. Note that this is not always true of circular convolution of finite length and periodic signals as there is then a maximum possible duration within a period.

1.3.2.3 Convolution Properties Summary

As can be seen the operation of continuous time convolution has several important properties that have been listed and proven in this module. With slight modifications to proofs, most of these also extend to continuous time circular convolution as well and the cases in which exceptions occur have been noted above. These identities will be useful to keep in mind as the reader continues to study signals and systems.

1.4 Frequency Domain

1.4.1 Introduction to the Frequency Domain[18]

In developing ways of analyzing linear circuits, we invented the impedance method because it made solving circuits easier. Along the way, we developed the notion of a circuit's frequency response or transfer function. This notion, which also applies to all linear, time-invariant systems, describes how the circuit responds to a sinusoidal input when we express it in terms of a complex exponential. We also learned the Superposition Principle for linear systems: The system's output to an input consisting of a sum of two signals is the sum of the system's outputs to each individual component.

The study of the frequency domain combines these two notions–a system's sinusoidal response is easy to find and a linear system's output to a sum of inputs is the sum of the individual outputs–to develop the crucial idea of a signal's **spectrum**. We begin by finding that those signals that can be represented as a sum of sinusoids is very large. In fact, **all signals can be expressed as a superposition of sinusoids**.

As this story unfolds, we'll see that information systems rely heavily on spectral ideas. For example, radio, television, and cellular telephones transmit over different portions of the spectrum. In fact, spectrum is so important that communications systems are regulated as to which portions of the spectrum they can use by the Federal Communications Commission in the United States and by International Treaty for the world (see Frequency Allocations[19]). Calculating the spectrum is easy: The **Fourier transform** defines how we can find a signal's spectrum.

1.4.2 Complex Fourier Series[20]

In an earlier module[21], we showed that a square wave could be expressed as a superposition of pulses. As useful as this decomposition was in this example, it does not generalize well to other periodic signals: How can a superposition of pulses equal a smooth signal like a sinusoid? Because of the importance of sinusoids to linear systems, you might wonder whether they could be added together to represent a large number of periodic signals. You would be right and in good company as well. Euler[22] and Gauss[23] in particular worried about this problem, and Jean Baptiste Fourier[24] got the credit even though tough mathematical issues were not settled until later. They worked on what is now known as the **Fourier series**: representing **any** periodic signal as a superposition of sinusoids.

But the Fourier series goes well beyond being another signal decomposition method. Rather, the Fourier series begins our journey to appreciate how a signal can be described in either the time-domain or the frequency-domain with **no** compromise. Let $s(t)$ be a **periodic** signal with period T. We want to show that periodic signals, even those that have constant-valued segments like a square wave, can be expressed as sum of **harmonically** related sine waves: sinusoids having frequencies that are integer multiples of the **fundamental frequency**. Because the signal has period T, the fundamental frequency is $\frac{1}{T}$. The complex Fourier series expresses the signal as a superposition of complex exponentials having frequencies $\frac{k}{T}$, $k = \{\dots, -1, 0, 1, \dots\}$.

$$s(t) = \sum_{k=-\infty}^{\infty} c_k e^{i\frac{2\pi kt}{T}} \tag{1.87}$$

with $c_k = \frac{1}{2}(a_k - i b_k)$. The real and imaginary parts of the **Fourier coefficients** c_k are written in this unusual way for convenience in defining the classic Fourier series. The zeroth coefficient equals the signal's average value and is real- valued for real-valued signals: $c_0 = a_0$. The family of functions $\left\{e^{i\frac{2\pi kt}{T}}\right\}$ are called

[18]This content is available online at <http://cnx.org/content/m0038/2.10/>.
[19]"Frequency Allocations" <http://cnx.org/content/m0083/latest/>
[20]This content is available online at <http://cnx.org/content/m0042/2.33/>.
[21]"Signal Decomposition", Exercise 1 <http://cnx.org/content/m0008/latest/#swsuper>
[22]http://www-groups.dcs.st- and.ac.uk/~history/Mathematicians/Euler.html
[23]http://www-groups.dcs.st- and.ac.uk/~history/Mathematicians/Guass.html
[24]http://www-groups.dcs.st- and.ac.uk/~history/Mathematicians/Fourier.html

basis functions and form the foundation of the Fourier series. No matter what the periodic signal might be, these functions are always present and form the representation's building blocks. They depend on the signal period T, and are indexed by k.

KEY POINT: Assuming we know the period, knowing the Fourier coefficients is equivalent to knowing the signal. Thus, it makes no difference if we have a time-domain or a frequency- domain characterization of the signal.

Exercise 1.4.2.1 *(Solution on p. 139.)*

What is the complex Fourier series for a sinusoid?

To find the Fourier coefficients, we note the orthogonality property

$$\int_0^T e^{i\frac{2\pi kt}{T}} e^{(-i)\frac{2\pi lt}{T}}\,dt = \begin{cases} T & \text{if } k = l \\ 0 & \text{if } k \neq l \end{cases} \quad (1.88)$$

Assuming for the moment that the complex Fourier series "works," we can find a signal's complex Fourier coefficients, its **spectrum**, by exploiting the orthogonality properties of harmonically related complex exponentials. Simply multiply each side of (1.87) by $e^{-(i2\pi lt)}$ and integrate over the interval $[0, T]$.

$$\begin{aligned} c_k &= \frac{1}{T}\int_0^T s(t)\, e^{-\left(i\frac{2\pi kt}{T}\right)}\,dt \\ c_0 &= \frac{1}{T}\int_0^T s(t)\,dt \end{aligned} \quad (1.89)$$

Example 1.11

Finding the Fourier series coefficients for the square wave $\mathrm{sq}_T(t)$ is very simple. Mathematically, this signal can be expressed as

$$\mathrm{sq}_T(t) = \begin{cases} 1 & \text{if } 0 < t < \frac{T}{2} \\ -1 & \text{if } \frac{T}{2} < t < T \end{cases}$$

The expression for the Fourier coefficients has the form

$$c_k = \frac{1}{T}\int_0^{\frac{T}{2}} e^{-\left(i\frac{2\pi kt}{T}\right)}\,dt - \frac{1}{T}\int_{\frac{T}{2}}^{T} e^{-\left(i\frac{2\pi kt}{T}\right)}\,dt \quad (1.90)$$

NOTE: When integrating an expression containing i, treat it just like any other constant.

The two integrals are very similar, one equaling the negative of the other. The final expression becomes

$$\begin{aligned} c_k &= \frac{-2}{i2\pi k}\left((-1)^k - 1\right) \\ &= \begin{cases} \frac{2}{i\pi k} & \text{if } k\,\text{odd} \\ 0 & \text{if } k\,\text{even} \end{cases} \end{aligned} \quad (1.91)$$

$$\mathrm{sq}(t) = \sum_{k\in\{\dots,-3,-1,1,3,\dots\}} \frac{2}{i\pi k} e^{(i)\frac{2\pi kt}{T}} \quad (1.92)$$

Consequently, the square wave equals a sum of complex exponentials, but only those having frequencies equal to odd multiples of the fundamental frequency $\frac{1}{T}$. The coefficients decay slowly as the frequency index k increases. This index corresponds to the k-th harmonic of the signal's period.

A signal's Fourier series spectrum c_k has interesting properties.

Property 1.1:
If $s(t)$ is real, $c_k = \overline{c_{-k}}$ (real-valued periodic signals have conjugate-symmetric spectra).

This result follows from the integral that calculates the c_k from the signal. Furthermore, this result means that $\Re(c_k) = \Re(c_{-k})$: The real part of the Fourier coefficients for real-valued signals is even. Similarly, $\Im(c_k) = -\Im(c_{-k})$: The imaginary parts of the Fourier coefficients have odd symmetry. Consequently, if you are given the Fourier coefficients for positive indices and zero and are told the signal is real-valued, you can find the negative-indexed coefficients, hence the entire spectrum. This kind of symmetry, $c_k = \overline{c_{-k}}$, is known as **conjugate symmetry**.

Property 1.2:
If $s(-t) = s(t)$, which says the signal has even symmetry about the origin, $c_{-k} = c_k$.

Given the previous property for real-valued signals, the Fourier coefficients of even signals are real-valued. A real-valued Fourier expansion amounts to an expansion in terms of only cosines, which is the simplest example of an even signal.

Property 1.3:
If $s(-t) = -s(t)$, which says the signal has odd symmetry, $c_{-k} = -c_k$.

Therefore, the Fourier coefficients are purely imaginary. The square wave is a great example of an odd-symmetric signal.

Property 1.4:
The spectral coefficients for a periodic signal delayed by τ, $s(t-\tau)$, are $c_k e^{-\frac{i2\pi k\tau}{T}}$, where c_k denotes the spectrum of $s(t)$. Delaying a signal by τ seconds results in a spectrum having a **linear phase shift** of $-\frac{2\pi k\tau}{T}$ in comparison to the spectrum of the undelayed signal. Note that the spectral magnitude is unaffected. Showing this property is easy.

Proof:

$$\begin{aligned} \frac{1}{T}\int_0^T s(t-\tau)\, e^{(-i)\frac{2\pi kt}{T}}\, dt &= \frac{1}{T}\int_{-\tau}^{T-\tau} s(t)\, e^{(-i)\frac{2\pi k(t+\tau)}{T}}\, dt \\ &= \frac{1}{T} e^{(-i)\frac{2\pi k\tau}{T}} \int_{-\tau}^{T-\tau} s(t)\, e^{(-i)\frac{2\pi kt}{T}}\, dt \end{aligned} \tag{1.93}$$

Note that the range of integration extends over a period of the integrand. Consequently, it should not matter how we integrate over a period, which means that $\int_{-\tau}^{T-\tau} (\cdot)\, dt = \int_0^T (\cdot)\, dt$, and we have our result.

The complex Fourier series obeys **Parseval's Theorem**, one of the most important results in signal analysis. This general mathematical result says you can calculate a signal's power in either the time domain or the frequency domain.

Theorem 1.1: Parseval's Theorem
Average power calculated in the time domain equals the power calculated in the frequency domain.

$$\frac{1}{T}\int_0^T s^2(t)\, dt = \sum_{k=-\infty}^{\infty} (|c_k|)^2 \tag{1.94}$$

This result is a (simpler) re-expression of how to calculate a signal's power than with the real-valued Fourier series expression for power.

Let's calculate the Fourier coefficients of the periodic pulse signal shown here (Figure 1.39).

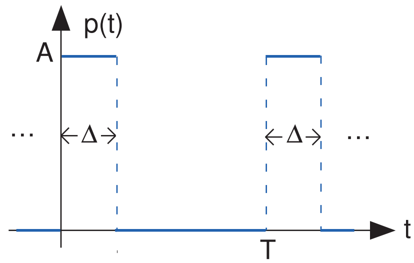


Figure 1.39: Periodic pulse signal.

The pulse width is Δ, the period T, and the amplitude A. The complex Fourier spectrum of this signal is given by

$$c_k = \frac{1}{T}\int_0^{\Delta} A e^{-\frac{i2\pi kt}{T}}\, dt = -\left(\frac{A}{i2\pi k}\left(e^{-\frac{i2\pi k\Delta}{T}} - 1\right)\right)$$

At this point, simplifying this expression requires knowing an interesting property.

$$1 - e^{-(i\theta)} = e^{-\frac{i\theta}{2}}\left(e^{\frac{i\theta}{2}} - e^{-\frac{i\theta}{2}}\right) = e^{-\frac{i\theta}{2}} 2i\sin\left(\frac{\theta}{2}\right)$$

Armed with this result, we can simply express the Fourier series coefficients for our pulse sequence.

$$c_k = A e^{-\frac{i\pi k\Delta}{T}} \frac{\sin\left(\frac{\pi k\Delta}{T}\right)}{\pi k} \tag{1.95}$$

Because this signal is real-valued, we find that the coefficients do indeed have conjugate symmetry: $c_k = \overline{c_{-k}}$. The periodic pulse signal has neither even nor odd symmetry; consequently, no additional symmetry exists in the spectrum. Because the spectrum is complex valued, to plot it we need to calculate its magnitude and phase.

$$|c_k| = A\left|\frac{\sin\left(\frac{\pi k\Delta}{T}\right)}{\pi k}\right| \tag{1.96}$$

$$\angle\left(c_k\right) = -\frac{\pi k\Delta}{T} + \pi \mathrm{neg}\left(\frac{\sin\left(\frac{\pi k\Delta}{T}\right)}{\pi k}\right)\mathrm{sign}\left(k\right)$$

The function $\mathrm{neg}\left(\cdot\right)$ equals -1 if its argument is negative and zero otherwise. The somewhat complicated expression for the phase results because the sine term can be negative; magnitudes must be positive, leaving the occasional negative values to be accounted for as a phase shift of π.

Periodic Pulse Sequence

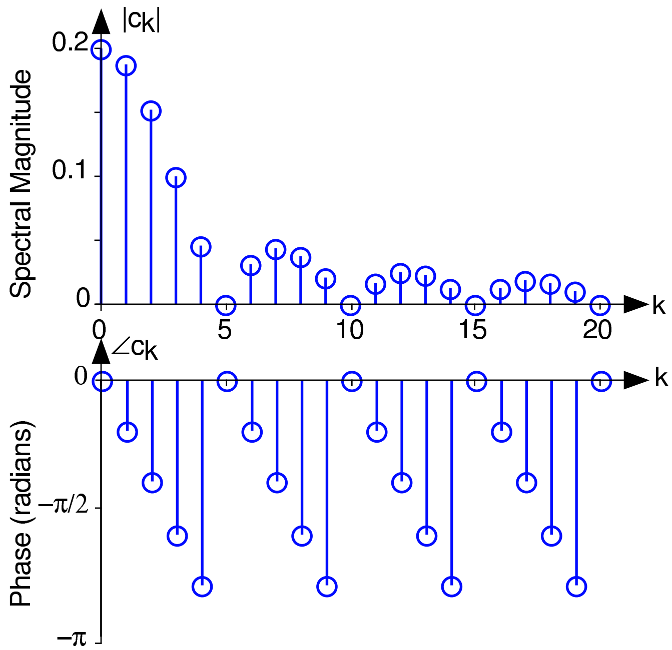


Figure 1.40: The magnitude and phase of the periodic pulse sequence's spectrum is shown for positive-frequency indices. Here $\frac{\Delta}{T} = 0.2$ and $A = 1$.

Also note the presence of a linear phase term (the first term in $\angle(c_k)$ is proportional to frequency $\frac{k}{T}$). Comparing this term with that predicted from delaying a signal, a delay of $\frac{\Delta}{2}$ is present in our signal. Advancing the signal by this amount centers the pulse about the origin, leaving an even signal, which in turn means that its spectrum is real-valued. Thus, our calculated spectrum is consistent with the properties of the Fourier spectrum.

Exercise 1.4.2.2 ***(Solution on p. 139.)***

What is the value of c_0? Recalling that this spectral coefficient corresponds to the signal's average value, does your answer make sense?

The phase plot shown in Figure 1.40 (Periodic Pulse Sequence) requires some explanation as it does not seem to agree with what (1.96) suggests. There, the phase has a linear component, with a jump of π every time the sinusoidal term changes sign. We must realize that any integer multiple of 2π can be added to a phase at each frequency **without** affecting the value of the complex spectrum. We see that at frequency index 4 the phase is nearly $-\pi$. The phase at index 5 is undefined because the magnitude is zero in this example. At index 6, the formula suggests that the phase of the linear term should be less than $-\pi$ (more negative). In addition, we expect a shift of $-\pi$ in the phase between indices 4 and 6. Thus, the phase value predicted by the formula is a little less than $-(2\pi)$. Because we can add 2π without affecting the value of the spectrum at index 6, the result is a slightly negative number as shown. Thus, the formula and the plot do agree. In phase calculations like those made in MATLAB, values are usually confined to the range $[-\pi, \pi)$ by adding some (possibly negative) multiple of 2π to each phase value.

1.4.3 Classic Fourier Series[25]

The classic Fourier series as derived originally expressed a periodic signal (period T) in terms of harmonically related sines and cosines.

$$s(t) = a_0 + \sum_{k=1}^{\infty} a_k \cos\left(\frac{2\pi kt}{T}\right) + \sum_{k=1}^{\infty} b_k \sin\left(\frac{2\pi kt}{T}\right) \tag{1.97}$$

The complex Fourier series and the sine-cosine series are identical, each representing a signal's spectrum. The **Fourier coefficients**, a_k and b_k, express the real and imaginary parts respectively of the spectrum while the coefficients c_k of the complex Fourier series express the spectrum as a magnitude and phase. Equating the classic Fourier series (1.97) to the complex Fourier series (1.87), an extra factor of two and complex conjugate become necessary to relate the Fourier coefficients in each.

$$c_k = \frac{1}{2}(a_k - ib_k)$$

Exercise 1.4.3.1 ***(Solution on p. 139.)***

Derive this relationship between the coefficients of the two Fourier series.

Just as with the complex Fourier series, we can find the Fourier coefficients using the **orthogonality** properties of sinusoids. Note that the cosine and sine of harmonically related frequencies, even the **same** frequency, are orthogonal.

$$\forall k, l, k \in \mathbb{Z} l \in \mathbb{Z} : \left(\int_0^T \sin\left(\frac{2\pi kt}{T}\right)\cos\left(\frac{2\pi lt}{T}\right) dt = 0\right) \tag{1.98}$$

$$\int_0^T \sin\left(\frac{2\pi kt}{T}\right)\sin\left(\frac{2\pi lt}{T}\right) dt = \begin{cases} \frac{T}{2} \text{ if } (k = l) \wedge (k \neq 0) \wedge (l \neq 0) \\ 0 \text{ if } (k \neq l) \vee (k = 0 = l) \end{cases}$$

$$\int_0^T \cos\left(\frac{2\pi kt}{T}\right)\cos\left(\frac{2\pi lt}{T}\right) dt = \begin{cases} \frac{T}{2} \text{ if } (k = l) \wedge (k \neq 0) \wedge (l \neq 0) \\ T \text{ if } k = 0 = l \\ 0 \text{ if } k \neq l \end{cases}$$

These orthogonality relations follow from the following important trigonometric identities.

$$\begin{aligned} \sin(\alpha)\sin(\beta) &= \tfrac{1}{2}(\cos(\alpha-\beta) - \cos(\alpha+\beta)) \\ \cos(\alpha)\cos(\beta) &= \tfrac{1}{2}(\cos(\alpha+\beta) + \cos(\alpha-\beta)) \\ \sin(\alpha)\cos(\beta) &= \tfrac{1}{2}(\sin(\alpha+\beta) + \sin(\alpha-\beta)) \end{aligned} \tag{1.99}$$

These identities allow you to substitute a sum of sines and/or cosines for a product of them. Each term in the sum can be integrated by noticing one of two important properties of sinusoids.

- The integral of a sinusoid over an **integer** number of periods equals zero.
- The integral of the **square** of a unit-amplitude sinusoid over a period T equals $\frac{T}{2}$.

To use these, let's, for example, multiply the Fourier series for a signal by the cosine of the l^{th} harmonic $\cos\left(\frac{2\pi lt}{T}\right)$ and integrate. The idea is that, because integration is linear, the integration will sift out all but the term involving a_l.

$$\int_0^T s(t)\cos\left(\frac{2\pi lt}{T}\right) dt = \int_0^T a_0 \cos\left(\frac{2\pi lt}{T}\right) dt + \sum_{k=1}^{\infty} a_k \int_0^T \cos\left(\frac{2\pi kt}{T}\right)\cos\left(\frac{2\pi lt}{T}\right) dt + \sum_{k=1}^{\infty} b_k \int_0^T \sin\left(\frac{2\pi kt}{T}\right)\cos\left(\frac{2\pi lt}{T}\right) dt \tag{1.100}$$

[25]This content is available online at <http://cnx.org/content/m0039/2.24/>.

The first and third terms are zero; in the second, the only non-zero term in the sum results when the indices k and l are equal (but not zero), in which case we obtain $\frac{a_l T}{2}$. If $k = 0 = l$, we obtain $a_0 T$. Consequently,

$$\forall l, l \neq 0 : \left(a_l = \frac{2}{T} \int_0^T s(t) \cos\left(\frac{2\pi l t}{T}\right) dt\right)$$

All of the Fourier coefficients can be found similarly.

$$\begin{array}{l} a_0 = \frac{1}{T} \int_0^T s(t)\, dt \\ \forall k, k \neq 0 : \left(a_k = \frac{2}{T} \int_0^T s(t) \cos\left(\frac{2\pi k t}{T}\right) dt\right) \\ b_k = \frac{2}{T} \int_0^T s(t) \sin\left(\frac{2\pi k t}{T}\right) dt \end{array} \quad (1.101)$$

Exercise 1.4.3.2 *(Solution on p. 139.)*

The expression for a_0 is referred to as the **average value** of $s(t)$. Why?

Exercise 1.4.3.3 *(Solution on p. 139.)*

What is the Fourier series for a unit-amplitude square wave?

Example 1.12

Let's find the Fourier series representation for the half-wave rectified sinusoid.

$$s(t) = \begin{cases} \sin\left(\frac{2\pi t}{T}\right) & \text{if } 0 \leq t < \frac{T}{2} \\ 0 & \text{if } \frac{T}{2} \leq t < T \end{cases} \quad (1.102)$$

Begin with the sine terms in the series; to find b_k we must calculate the integral

$$b_k = \frac{2}{T} \int_0^{\frac{T}{2}} \sin\left(\frac{2\pi t}{T}\right) \sin\left(\frac{2\pi k t}{T}\right) dt \quad (1.103)$$

Using our trigonometric identities turns our integral of a product of sinusoids into a sum of integrals of individual sinusoids, which are much easier to evaluate.

$$\begin{aligned} \int_0^{\frac{T}{2}} \sin\left(\frac{2\pi t}{T}\right) \sin\left(\frac{2\pi k t}{T}\right) dt &= \frac{1}{2} \int_0^{\frac{T}{2}} \cos\left(\frac{2\pi (k-1) t}{T}\right) - \cos\left(\frac{2\pi (k+1) t}{T}\right) dt \\ &= \begin{cases} \frac{1}{2} & \text{if } k = 1 \\ 0 & \text{otherwise} \end{cases} \end{aligned} \quad (1.104)$$

Thus,

$$b_1 = \frac{1}{2}$$

$$b_2 = b_3 = \dots = 0$$

On to the cosine terms. The average value, which corresponds to a_0, equals $\frac{1}{\pi}$. The remainder of the cosine coefficients are easy to find, but yield the complicated result

$$a_k = \begin{cases} -\left(\frac{2}{\pi} \frac{1}{k^2 - 1}\right) & \text{if } k \in \{2, 4, \dots\} \\ 0 & \text{if k odd} \end{cases} \quad (1.105)$$

Thus, the Fourier series for the half-wave rectified sinusoid has non-zero terms for the average, the fundamental, and the even harmonics.

1.4.4 A Signal's Spectrum[26]

A periodic signal, such as the half-wave rectified sinusoid, consists of a sum of elemental sinusoids. A plot of the Fourier coefficients as a function of the frequency index, such as shown in Figure 1.41 (Fourier Series spectrum of a half-wave rectified sine wave), displays the signal's **spectrum**. The word "spectrum" implies that the independent variable, here k, corresponds somehow to frequency. Each coefficient is directly related to a sinusoid having a frequency of $\frac{k}{T}$. Thus, if we half-wave rectified a 1 kHz sinusoid, $k=1$ corresponds to 1 kHz, $k=2$ to 2 kHz, etc.

Fourier Series spectrum of a half-wave rectified sine wave

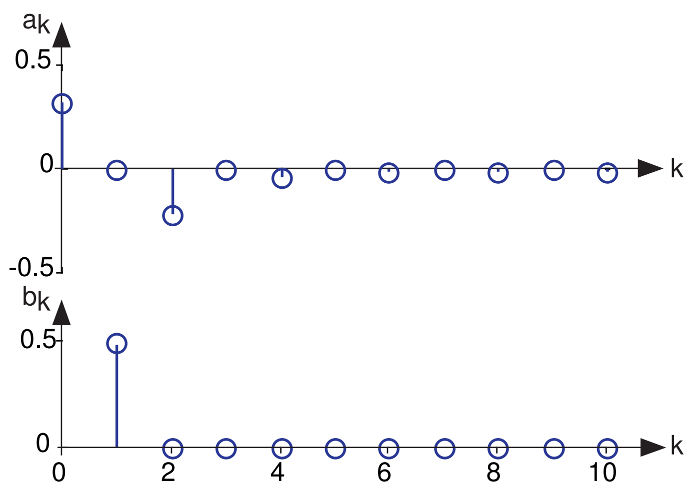


Figure 1.41: The Fourier series spectrum of a half-wave rectified sinusoid is shown. The index indicates the multiple of the fundamental frequency at which the signal has energy.

A subtle, but very important, aspect of the Fourier spectrum is its **uniqueness**: You can unambiguously find the spectrum from the signal (decomposition (1.101)) and the signal from the spectrum (composition). Thus, any aspect of the signal can be found from the spectrum and vice versa. **A signal's frequency domain expression is its spectrum**. A periodic signal can be defined either in the time domain (as a function) or in the frequency domain (as a spectrum).

A fundamental aspect of solving electrical engineering problems is whether the time or frequency domain provides the most understanding of a signal's properties and the simplest way of manipulating it. The uniqueness property says that either domain can provide the right answer. As a simple example, suppose we want to know the (periodic) signal's maximum value. Clearly the time domain provides the answer directly. To use a frequency domain approach would require us to find the spectrum, form the signal from the spectrum and calculate the maximum; we're back in the time domain!

Another feature of a signal is its average **power**. A signal's instantaneous power is defined to be its square. The average power is the average of the instantaneous power over some time interval. For a periodic signal, the natural time interval is clearly its period; for nonperiodic signals, a better choice would be entire time or time from onset. For a periodic signal, the average power is the square of its root-mean-squared

[26]This content is available online at <http://cnx.org/content/m0040/2.21/>.

(rms) value. We define the **rms** value of a periodic signal to be

$$\mathrm{rms}(s) = \sqrt{\frac{1}{T}\int_0^T s^2(t)\,dt} \quad (1.106)$$

and thus its average power is

$$\begin{aligned} \mathrm{power}(s) &= \mathrm{rms}^2(s) \\ &= \frac{1}{T}\int_0^T s^2(t)\,dt \end{aligned} \quad (1.107)$$

Exercise 1.4.4.1 ***(Solution on p. 139.)***

What is the rms value of the half-wave rectified sinusoid?

To find the average power in the frequency domain, we need to substitute the spectral representation of the signal into this expression.

$$\mathrm{power}(s) = \frac{1}{T}\int_0^T \left(a_0 + \sum_{k=1}^{\infty} a_k \cos\left(\frac{2\pi kt}{T}\right) + \sum_{k=1}^{\infty} b_k \sin\left(\frac{2\pi kt}{T}\right)\right)^2 dt$$

The square inside the integral will contain all possible pairwise products. However, the orthogonality properties (1.98) say that most of these crossterms integrate to zero. The survivors leave a rather simple expression for the power we seek.

$$\mathrm{power}(s) = {a_0}^2 + \frac{1}{2}\sum_{k=1}^{\infty} {a_k}^2 + {b_k}^2 \quad (1.108)$$

Power Spectrum of a Half-Wave Rectified Sinusoid

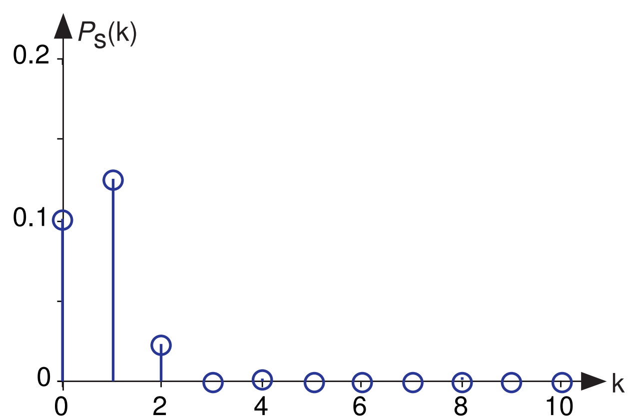


Figure 1.42: Power spectrum of a half-wave rectified sinusoid.

It could well be that computing this sum is easier than integrating the signal's square. Furthermore, the contribution of each term in the Fourier series toward representing the signal can be measured by its contribution to the signal's average power. Thus, the power contained in a signal at its kth harmonic is $\frac{{a_k}^2+{b_k}^2}{2}$. The **power spectrum**, $P_s(k)$, such as shown in Figure 1.42 (Power Spectrum of a Half-Wave Rectified Sinusoid), plots each harmonic's contribution to the total power.

Exercise 1.4.4.2 *(Solution on p. 139.)*

In high-end audio, deviation of a sine wave from the ideal is measured by the **total harmonic distortion**, which equals the total power in the harmonics higher than the first compared to power in the fundamental. Find an expression for the total harmonic distortion for any periodic signal. Is this calculation most easily performed in the time or frequency domain?

1.4.5 Fourier Series Approximation of Signals[27]

It is interesting to consider the sequence of signals that we obtain as we incorporate more terms into the Fourier series approximation of the half-wave rectified sine wave (Example 1.12). Define $s_K(t)$ to be the signal containing $K+1$ Fourier terms.

$$s_K(t) = a_0 + \sum_{k=1}^{K} a_k \cos\left(\frac{2\pi kt}{T}\right) + \sum_{k=1}^{K} b_k \sin\left(\frac{2\pi kt}{T}\right) \tag{1.109}$$

Figure 1.43 (Fourier Series spectrum of a half-wave rectified sine wave) shows how this sequence of signals portrays the signal more accurately as more terms are added.

[27] This content is available online at <http://cnx.org/content/m10687/2.11/>.

Fourier Series spectrum of a half-wave rectified sine wave

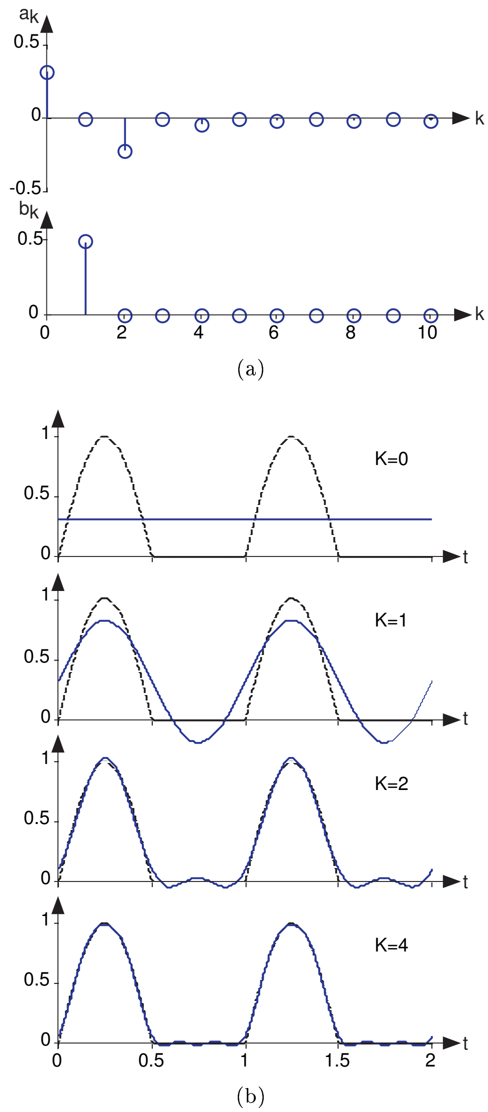


Figure 1.43: The Fourier series spectrum of a half-wave rectified sinusoid is shown in the upper portion. The index indicates the multiple of the fundamental frequency at which the signal has energy. The cumulative effect of adding terms to the Fourier series for the half-wave rectified sine wave is shown in the bottom portion. The dashed line is the actual signal, with the solid line showing the finite series approximation to the indicated number of terms, $K+1$.

We need to assess quantitatively the accuracy of the Fourier series approximation so that we can judge how rapidly the series approaches the signal. When we use a $K+1$-term series, the error—the difference between the signal and the $K+1$-term series—corresponds to the unused terms from the series.

$$\epsilon_K(t) = \sum_{k=K+1}^{\infty} a_k \cos\left(\frac{2\pi kt}{T}\right) + \sum_{k=K+1}^{\infty} b_k \sin\left(\frac{2\pi kt}{T}\right) \quad (1.110)$$

To find the rms error, we must square this expression and integrate it over a period. Again, the integral of most cross-terms is zero, leaving

$$\mathrm{rms}(\epsilon_K) = \sqrt{\frac{1}{2}\sum_{k=K+1}^{\infty} {a_k}^2 + {b_k}^2} \quad (1.111)$$

Figure 1.44 (Approximation error for a half-wave rectified sinusoid) shows how the error in the Fourier series for the half-wave rectified sinusoid decreases as more terms are incorporated. In particular, the use of four terms, as shown in the bottom plot of Figure 1.43 (Fourier Series spectrum of a half-wave rectified sine wave), has a rms error (relative to the rms value of the signal) of about 3%. The Fourier series in this case converges quickly to the signal.

Approximation error for a half-wave rectified sinusoid

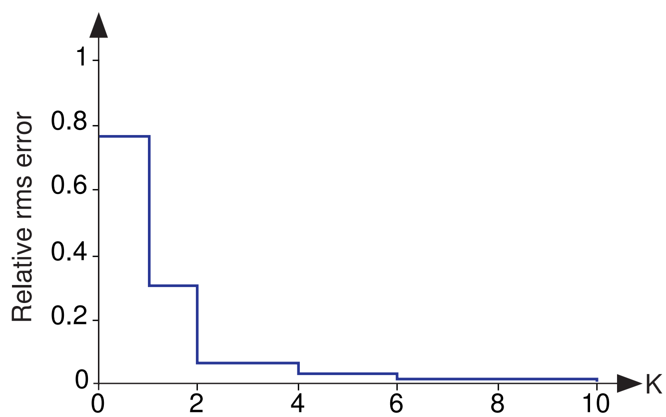


Figure 1.44: The rms error calculated according to (1.111) is shown as a function of the number of terms in the series for the half-wave rectified sinusoid. The error has been normalized by the rms value of the signal.

We can look at Figure 1.45 (Power spectrum and approximation error for a square wave) to see the power spectrum and the rms approximation error for the square wave.

Power spectrum and approximation error for a square wave

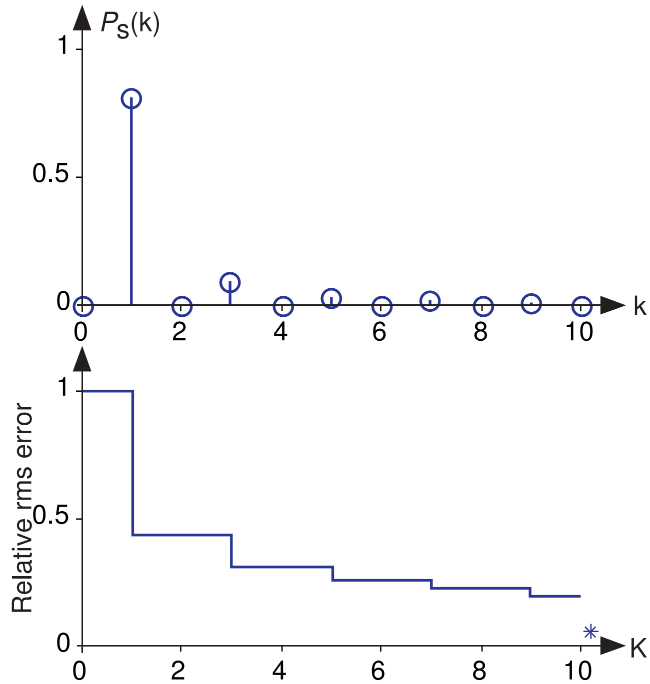


Figure 1.45: The upper plot shows the power spectrum of the square wave, and the lower plot the rms error of the finite-length Fourier series approximation to the square wave. The asterisk denotes the rms error when the number of terms K in the Fourier series equals 99.

Because the Fourier coefficients decay more slowly here than for the half-wave rectified sinusoid, the rms error is not decreasing quickly. Said another way, the square-wave's spectrum contains more power at higher frequencies than does the half-wave-rectified sinusoid. This difference between the two Fourier series results because the half-wave rectified sinusoid's Fourier coefficients are proportional to $\frac{1}{k^2}$ while those of the square wave are proportional to $\frac{1}{k}$. If fact, after 99 terms of the square wave's approximation, the error is bigger than 10 terms of the approximation for the half-wave rectified sinusoid. Mathematicians have shown that no signal has an rms approximation error that decays more slowly than it does for the square wave.

Exercise 1.4.5.1 *(Solution on p. 139.)*

Calculate the harmonic distortion for the square wave.

More than just decaying slowly, Fourier series approximation shown in Figure 1.46 (Fourier series approximation of a square wave) exhibits interesting behavior.

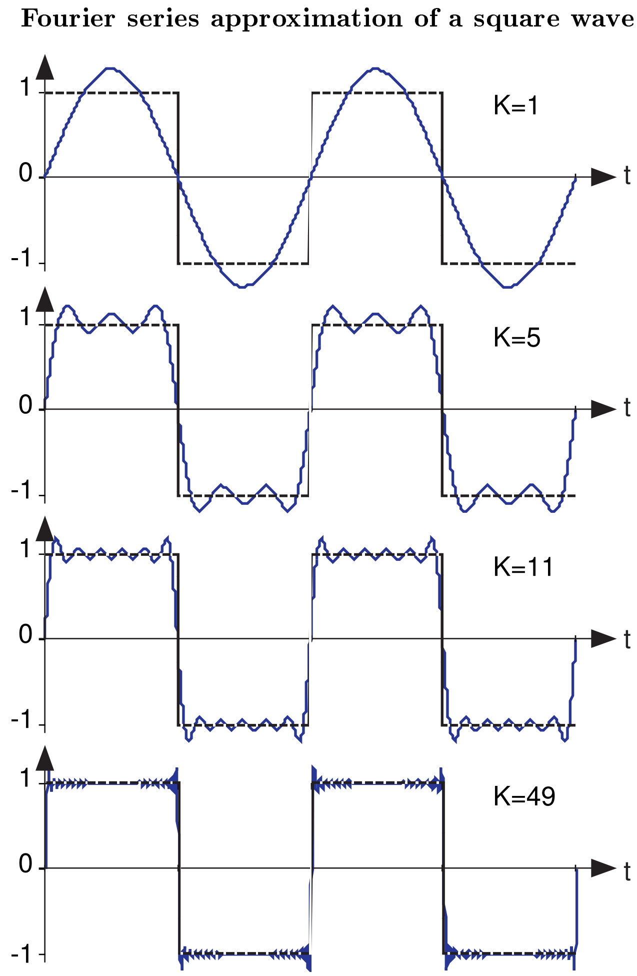


Figure 1.46: Fourier series approximation to $\mathrm{sq}(t)$. The number of terms in the Fourier sum is indicated in each plot, and the square wave is shown as a dashed line over two periods.

Although the square wave's Fourier series requires more terms for a given representation accuracy, when comparing plots it is not clear that the two are equal. Does the Fourier series really equal the square wave at **all** values of t? In particular, at each step-change in the square wave, the Fourier series exhibits a peak followed by rapid oscillations. As more terms are added to the series, the oscillations seem to become more rapid and smaller, but the peaks are not decreasing. For the Fourier series approximation for the half-wave rectified sinusoid (Figure 1.43: Fourier Series spectrum of a half-wave rectified sine wave), no such behavior occurs. What is happening?

Consider this mathematical question intuitively: Can a discontinuous function, like the square wave, be expressed as a sum, even an infinite one, of continuous signals? One should at least be suspicious, and in fact, it can't be thus expressed. This issue brought Fourier[28] much criticism from the French Academy of Science (Laplace, Lagrange, Monge and LaCroix comprised the review committee) for several years after its presentation on 1807. It was not resolved for almost a century, and its resolution is interesting and important to understand from a practical viewpoint.

[28]http://www-groups.dcs.st-and.ac.uk/~history/Mathematicians/Fourier.html

The extraneous peaks in the square wave's Fourier series **never** disappear; they are termed **Gibb's phenomenon** after the American physicist Josiah Willard Gibbs. They occur whenever the signal is discontinuous, and will always be present whenever the signal has jumps.

Let's return to the question of equality; how can the equal sign in the definition of the Fourier series be justified? The partial answer is that **pointwise**—each and every value of t—equality is **not** guaranteed. However, mathematicians later in the nineteenth century showed that the rms error of the Fourier series was always zero.

$$\operatorname*{limit}_{K \to \infty} \mathrm{rms}\left(\epsilon_K\right) = 0$$

What this means is that the error between a signal and its Fourier series approximation may not be zero, but that its rms value will be zero! It is through the eyes of the rms value that we redefine equality: The usual definition of equality is called **pointwise equality**: Two signals $s_1(t)$, $s_2(t)$ are said to be equal pointwise if $s_1(t) = s_2(t)$ for all values of t. A new definition of equality is **mean-square equality**: Two signals are said to be equal in the mean square if $\mathrm{rms}(s_1 - s_2) = 0$. For Fourier series, Gibb's phenomenon peaks have finite height and zero width. The error differs from zero only at isolated points—whenever the periodic signal contains discontinuities—and equals about 9% of the size of the discontinuity. The value of a function at a finite set of points does not affect its integral. This effect underlies the reason why defining the value of a discontinuous function, like we refrained from doing in defining the step function[29], at its discontinuity is meaningless. Whatever you pick for a value has no practical relevance for either the signal's spectrum or for how a system responds to the signal. The Fourier series value "at" the discontinuity is the average of the values on either side of the jump.

1.4.6 Encoding Information in the Frequency Domain[30]

To emphasize the fact that every periodic signal has both a time and frequency domain representation, we can exploit both to **encode information** into a signal. Refer to the Fundamental Model of Communication[31]. We have an information source, and want to construct a transmitter that produces a signal $x(t)$. For the source, let's assume we have information to encode every T seconds. For example, we want to represent typed letters produced by an extremely good typist (a key is struck every T seconds). Let's consider the complex Fourier series formula in the light of trying to encode information.

$$x(t) = \sum_{k=-K}^{K} c_k e^{i\frac{2\pi kt}{T}} \tag{1.112}$$

We use a finite sum here merely for simplicity (fewer parameters to determine). An important aspect of the spectrum is that each frequency component c_k can be manipulated separately: Instead of finding the Fourier spectrum from a time-domain specification, let's construct it in the frequency domain by selecting the c_k according to some rule that relates coefficient values to the alphabet. In defining this rule, we want to always create a real-valued signal $x(t)$. Because of the Fourier spectrum's properties (Property 1.1, p. 41), the spectrum must have conjugate symmetry. This requirement means that we can only assign positive-indexed coefficients (positive frequencies), with negative-indexed ones equaling the complex conjugate of the corresponding positive-indexed ones.

Assume we have N letters to encode: $\{a_1, \ldots, a_N\}$. One simple encoding rule could be to make a single Fourier coefficient be non-zero and all others zero for each letter. For example, if a_n occurs, we make $c_n = 1$ and $c_k = 0$, $k \neq n$. In this way, the n^{th} harmonic of the frequency $\frac{1}{T}$ is used to represent a letter. Note that the **bandwidth**—the range of frequencies required for the encoding—equals $\frac{N}{T}$. Another possibility is to consider the binary representation of the letter's index. For example, if the letter a_{13} occurs, converting

[29] "Elemental Signals": Section Unit Step <http://cnx.org/content/m0004/latest/#stepdef>
[30] This content is available online at <http://cnx.org/content/m0043/2.17/>.
[31] "Structure of Communication Systems", Figure 1: Fundamental model of communication <http://cnx.org/content/m0002/latest/#commsys>

13 to its base 2 representation, we have $13 = 1101_2$. We can use the pattern of zeros and ones to represent directly which Fourier coefficients we "turn on" (set equal to one) and which we "turn off."

Exercise 1.4.6.1 *(Solution on p. 139.)*

Compare the bandwidth required for the direct encoding scheme (one nonzero Fourier coefficient for each letter) to the binary number scheme. Compare the bandwidths for a 128-letter alphabet. Since both schemes represent information without loss – we can determine the typed letter uniquely from the signal's spectrum – both are viable. Which makes more efficient use of bandwidth and thus might be preferred?

Exercise 1.4.6.2 *(Solution on p. 140.)*

Can you think of an information-encoding scheme that makes even more efficient use of the spectrum? In particular, can we use only one Fourier coefficient to represent N letters uniquely?

We can create an encoding scheme in the frequency domain (p. 53) to represent an alphabet of letters. But, as this information-encoding scheme stands, we can represent one letter for all time. However, we note that the Fourier coefficients depend **only** on the signal's characteristics over a single period. We could change the signal's spectrum every T as each letter is typed. In this way, we turn spectral coefficients on and off as letters are typed, thereby encoding the entire typed document. For the receiver (see the Fundamental Model of Communication[32]) to retrieve the typed letter, it would simply use the Fourier formula for the complex Fourier spectrum[33] for each T-second interval to determine what each typed letter was. Figure 1.47 (Encoding Signals) shows such a signal in the time-domain.

Encoding Signals

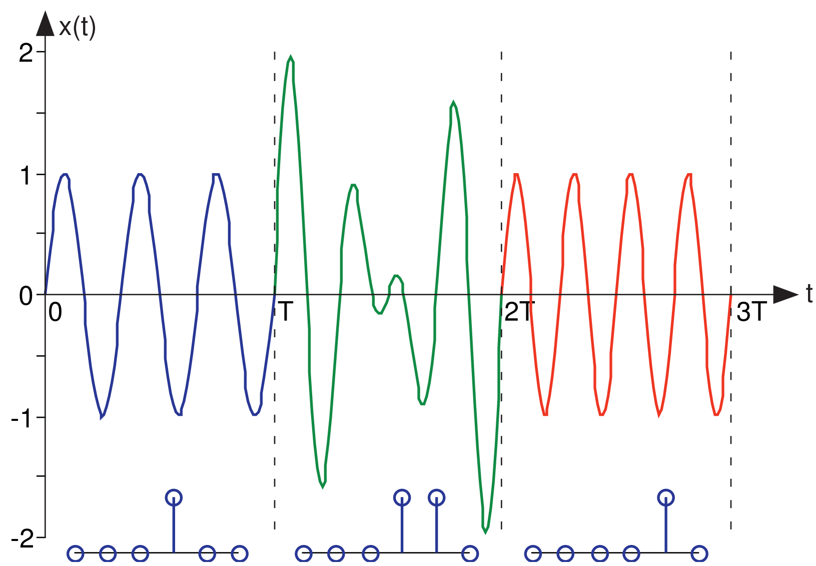


Figure 1.47: The encoding of signals via the Fourier spectrum is shown over three "periods." In this example, only the third and fourth harmonics are used, as shown by the spectral magnitudes corresponding to each T-second interval plotted below the waveforms. Can you determine the phase of the harmonics from the waveform?

[32]"Structure of Communication Systems", Figure 1: Fundamental model of communication <http://cnx.org/content/m0002/latest/#commsys>

[33]"Complex Fourier Series and Their Properties", (2) <http://cnx.org/content/m0065/latest/#complex>

In this Fourier-series encoding scheme, we have used the fact that spectral coefficients can be independently specified and that they can be uniquely recovered from the time-domain signal over one "period." Do note that the signal representing the entire document is no longer periodic. By understanding the Fourier series' properties (in particular that coefficients are determined only over a T-second interval, we can construct a communications system. This approach represents a simplification of how modern modems represent text that they transmit over telephone lines.

1.4.7 Filtering Periodic Signals[34]

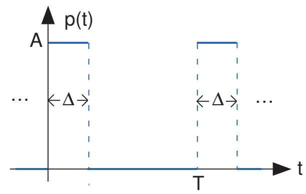
The Fourier series representation of a periodic signal makes it easy to determine how a linear, time-invariant filter reshapes such signals **in general**. The fundamental property of a linear system is that its input-output relation obeys superposition: $L(a_1 s_1(t) + a_2 s_2(t)) = a_1 L(s_1(t)) + a_2 L(s_2(t))$. Because the Fourier series represents a periodic signal as a linear combination of complex exponentials, we can exploit the superposition property. Furthermore, we found for linear circuits that their output to a complex exponential input is just the frequency response evaluated at the signal's frequency times the complex exponential. Said mathematically, if $x(t) = e^{i\frac{2\pi kt}{T}}$, then the output $y(t) = H\left(\frac{k}{T}\right)e^{i\frac{2\pi kt}{T}}$ because $f = \frac{k}{T}$. Thus, if $x(t)$ is periodic thereby having a Fourier series, a linear circuit's output to this signal will be the superposition of the output to each component.

$$y(t) = \sum_{k=-\infty}^{\infty} c_k H\left(\frac{k}{T}\right) e^{i\frac{2\pi kt}{T}} \tag{1.113}$$

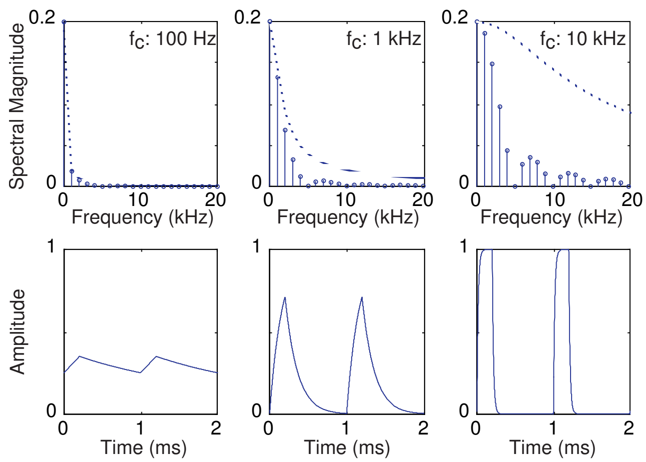
Thus, the output has a Fourier series, which means that it too is periodic. Its Fourier coefficients equal $c_k H\left(\frac{k}{T}\right)$. **To obtain the spectrum of the output, we simply multiply the input spectrum by the frequency response**. The circuit modifies the magnitude and phase of each Fourier coefficient. Note especially that while the Fourier coefficients do not depend on the signal's period, the circuit's transfer function does depend on frequency, which means that the circuit's output will differ as the period varies.

[34]This content is available online at <http://cnx.org/content/m0044/2.12/>.

Filtering a periodic signal



(a)



(b)

Figure 1.48: A periodic pulse signal, such as shown on the left part ($\frac{\Delta}{T} = 0.2$), serves as the input to an RC lowpass filter. The input's period was 1 ms (millisecond). The filter's cutoff frequency was set to the various values indicated in the top row, which display the output signal's spectrum and the filter's transfer function. The bottom row shows the output signal derived from the Fourier series coefficients shown in the top row. (a) Periodic pulse signal (b) Top plots show the pulse signal's spectrum for various cutoff frequencies. Bottom plots show the filter's output signals.

Example 1.13

The periodic pulse signal shown on the left above serves as the input to a RC-circuit that has the transfer function (calculated elsewhere[35])

$$H(f) = \frac{1}{1 + i2\pi fRC} \tag{1.114}$$

Figure 1.48 (Filtering a periodic signal) shows the output changes as we vary the filter's cutoff frequency. Note how the signal's spectrum extends well above its fundamental frequency. Having a cutoff frequency ten times higher than the fundamental does perceptibly change the output waveform, rounding the leading and trailing edges. As the cutoff frequency decreases (center, then left), the rounding becomes more prominent, with the leftmost waveform showing a small ripple.

[35]"Transfer Functions", Figure 2: Magnitude and phase of the transfer function <http://cnx.org/content/m0028/latest/#magphase>

Exercise 1.4.7.1 *(Solution on p. 140.)*

What is the average value of each output waveform? The correct answer may surprise you.

This example also illustrates the impact a lowpass filter can have on a waveform. The simple RC filter used here has a rather gradual frequency response, which means that higher harmonics are smoothly suppressed. Later, we will describe filters that have much more rapidly varying frequency responses, allowing a much more dramatic selection of the input's Fourier coefficients.

More importantly, we have calculated the output of a circuit to a periodic input **without** writing, much less solving, the differential equation governing the circuit's behavior. Furthermore, we made these calculations entirely in the frequency domain. Using Fourier series, we can calculate how **any** linear circuit will respond to a periodic input.

1.4.8 Derivation of the Fourier Transform[36]

Fourier series clearly open the frequency domain as an interesting and useful way of determining how circuits and systems respond to **periodic** input signals. Can we use similar techniques for nonperiodic signals? What is the response of the filter to a single pulse? Addressing these issues requires us to find the Fourier spectrum of all signals, both periodic and nonperiodic ones. We need a definition for **the** Fourier spectrum of a signal, periodic or not. This spectrum is calculated by what is known as the **Fourier transform**.

Let $s_T(t)$ be a periodic signal having period T. We want to consider what happens to this signal's spectrum as we let the period become longer and longer. We denote the spectrum for any assumed value of the period by $c_k(T)$. We calculate the spectrum according to the familiar formula

$$c_k(T) = \frac{1}{T}\int_{-\frac{T}{2}}^{\frac{T}{2}} s_T(t)\, e^{-\frac{i2\pi kt}{T}}\, dt \qquad (1.115)$$

where we have used a symmetric placement of the integration interval about the origin for subsequent derivational convenience. Let f be a **fixed** frequency equaling $\frac{k}{T}$; we vary the frequency index k proportionally as we increase the period. Define

$$S_T(f) \equiv T c_k(T) = \int_{-\frac{T}{2}}^{\frac{T}{2}} s_T(t)\, e^{-(i2\pi ft)}\, dt \qquad (1.116)$$

making the corresponding Fourier series

$$s_T(t) = \sum_{k=-\infty}^{\infty} S_T(f)\, e^{i2\pi ft}\frac{1}{T} \qquad (1.117)$$

As the period increases, the spectral lines become closer together, becoming a continuum. Therefore,

$$\operatorname*{limit}_{T\to\infty} s_T(t) \equiv s(t) = \int_{-\infty}^{\infty} S(f)\, e^{i2\pi ft}\, df \qquad (1.118)$$

with

$$S(f) = \int_{-\infty}^{\infty} s(t)\, e^{-(i2\pi ft)}\, dt \qquad (1.119)$$

$S(f)$ is the Fourier transform of $s(t)$ (the Fourier transform is symbolically denoted by the uppercase version of the signal's symbol) and is defined for **any** signal for which the integral ((1.119)) converges.

Example 1.14

Let's calculate the Fourier transform of the pulse signal[37], $p(t)$.

$$P(f) = \int_{-\infty}^{\infty} p(t)\, e^{-(i2\pi ft)}\, dt = \int_{0}^{\Delta} e^{-(i2\pi ft)}\, dt = \frac{1}{-(i2\pi f)}\left(e^{-(i2\pi f\Delta)} - 1\right)$$

[36] This content is available online at <http://cnx.org/content/m0046/2.22/>.
[37] "Elemental Signals": Section Pulse <http://cnx.org/content/m0004/latest/#pulsedef>

$$P(f) = e^{-(i\pi f\Delta)}\frac{\sin(\pi f\Delta)}{\pi f}$$

Note how closely this result resembles the expression for Fourier series coefficients of the periodic pulse signal (1.96).

Spectrum

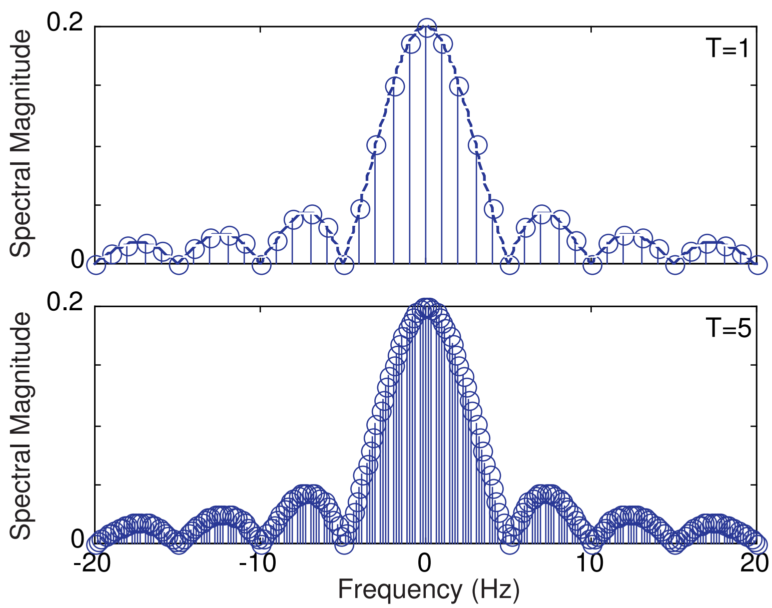


Figure 1.49: The upper plot shows the magnitude of the Fourier series spectrum for the case of $T = 1$ with the Fourier transform of $p(t)$ shown as a dashed line. For the bottom panel, we expanded the period to $T = 5$, keeping the pulse's duration fixed at 0.2, and computed its Fourier series coefficients.

Figure 1.49 (Spectrum) shows how increasing the period does indeed lead to a continuum of coefficients, and that the Fourier transform does correspond to what the continuum becomes. The quantity $\frac{\sin(t)}{t}$ has a special name, the **sinc** (pronounced "sink") function, and is denoted by $\mathrm{sinc}(t)$. Thus, the magnitude of the pulse's Fourier transform equals $|\Delta \mathrm{sinc}(\pi f\Delta)|$.

The Fourier transform relates a signal's time and frequency domain representations to each other. The direct Fourier transform (or simply the Fourier transform) calculates a signal's frequency domain representation from its time-domain variant ((1.120)). The inverse Fourier transform ((1.121)) finds the time-domain representation from the frequency domain. Rather than explicitly writing the required integral, we often symbolically express these transform calculations as $\mathcal{F}(s)$ and $\mathcal{F}^{-1}(S)$, respectively.

$$\begin{aligned}\mathcal{F}(s) &= S(f) \\ &= \int_{-\infty}^{\infty} s(t)\, e^{-(i2\pi f t)}\, dt\end{aligned} \quad (1.120)$$

$$\begin{aligned}\mathcal{F}^{-1}(S) &= s(t)\\ &= \int_{-\infty}^{\infty} S(f)\, e^{i2\pi f t} df\end{aligned} \tag{1.121}$$

We must have $s(t) = \mathcal{F}^{-1}(\mathcal{F}(s(t)))$ and $S(f) = \mathcal{F}\left(\mathcal{F}^{-1}(S(f))\right)$, and these results are indeed valid with minor exceptions.

NOTE: Recall that the Fourier series for a square wave gives a value for the signal at the discontinuities equal to the average value of the jump. This value may differ from how the signal is **defined** in the time domain, but being unequal at a point is indeed minor.

Showing that you "get back to where you started" is difficult from an analytic viewpoint, and we won't try here. Note that the direct and inverse transforms differ only in the sign of the exponent.

Exercise 1.4.8.1 ***(Solution on p. 140.)***

The differing exponent signs means that some curious results occur when we use the wrong sign. What is $\mathcal{F}(S(f))$? In other words, use the wrong exponent sign in evaluating the inverse Fourier transform.

Properties of the Fourier transform and some useful transform pairs are provided in the accompanying tables (Table 1.1: Short Table of Fourier Transform Pairs and Table 1.2: Fourier Transform Properties). Especially important among these properties is **Parseval's Theorem**, which states that power computed in either domain equals the power in the other.

$$\int_{-\infty}^{\infty} s^2(t)\, dt = \int_{-\infty}^{\infty} (|S(f)|)^2 df \tag{1.122}$$

Of practical importance is the conjugate symmetry property: When $s(t)$ is real-valued, the spectrum at negative frequencies equals the complex conjugate of the spectrum at the corresponding positive frequencies. Consequently, we need only plot the positive frequency portion of the spectrum (we can easily determine the remainder of the spectrum).

Exercise 1.4.8.2 ***(Solution on p. 140.)***

How many Fourier transform operations need to be applied to get the original signal back: $\mathcal{F}(\cdots(\mathcal{F}(s))) = s(t)$?

Note that the mathematical relationships between the time domain and frequency domain versions of the same signal are termed **transforms**. We are transforming (in the nontechnical meaning of the word) a signal from one representation to another. We express Fourier transform **pairs** as $s(t) \leftrightarrow S(f)$. A signal's time and frequency domain representations are uniquely related to each other. A signal thus "exists" in both the time and frequency domains, with the Fourier transform bridging between the two. We can define an information carrying signal in either the time or frequency domains; it behooves the wise engineer to use the simpler of the two.

A common misunderstanding is that while a signal exists in both the time and frequency domains, a single formula expressing a signal must contain **only** time or frequency: Both cannot be present simultaneously. This situation mirrors what happens with complex amplitudes in circuits: As we reveal how communications systems work and are designed, we will define signals entirely in the frequency domain without explicitly finding their time domain variants. This idea is shown in another module (Section 1.4.6) where we define Fourier series coefficients according to letter to be transmitted. Thus, a signal, though most familiarly defined in the time-domain, really can be defined equally as well (and sometimes more easily) in the frequency domain. For example, impedances depend on frequency and the time variable cannot appear.

We will learn (Section 1.4.9) that finding a linear, time-invariant system's output in the time domain can be most easily calculated by determining the input signal's spectrum, performing a simple calculation in the frequency domain, and inverse transforming the result. Furthermore, understanding communications and information processing systems requires a thorough understanding of signal structure and of how systems work in **both** the time and frequency domains.

The only difficulty in calculating the Fourier transform of any signal occurs when we have periodic signals (in either domain). Realizing that the Fourier series is a special case of the Fourier transform, we simply calculate the Fourier series coefficients instead, and plot them along with the spectra of nonperiodic signals on the same frequency axis.

Short Table of Fourier Transform Pairs

$s(t)$	$S(f)$
$e^{-(at)}u(t)$	$\frac{1}{i2\pi f+a}$
$e^{-(a\lvert t\rvert)}$	$\frac{2a}{4\pi^2 f^2+a^2}$
$p(t)=\begin{cases}1 & \text{if } \lvert t\rvert<\frac{\Delta}{2}\\ 0 & \text{if } \lvert t\rvert>\frac{\Delta}{2}\end{cases}$	$\frac{\sin(\pi f\Delta)}{\pi f}$
$\frac{\sin(2\pi Wt)}{\pi t}$	$S(f)=\begin{cases}1 & \text{if } \lvert f\rvert<W\\ 0 & \text{if } \lvert f\rvert>W\end{cases}$

Table 1.1

Fourier Transform Properties

	Time-Domain	**Frequency Domain**
Linearity	$a_1 s_1(t)+a_2 s_2(t)$	$a_1 S_1(f)+a_2 S_2(f)$
Conjugate Symmetry	$s(t)\in\mathbb{R}$	$S(f)=\overline{S(-f)}$
Even Symmetry	$s(t)=s(-t)$	$S(f)=S(-f)$
Odd Symmetry	$s(t)=-s(-t)$	$S(f)=-S(-f)$
Scale Change	$s(at)$	$\frac{1}{\lvert a\rvert}S\left(\frac{f}{a}\right)$
Time Delay	$s(t-\tau)$	$e^{-(i2\pi f\tau)}S(f)$
Complex Modulation	$e^{i2\pi f_0 t}s(t)$	$S(f-f_0)$
Amplitude Modulation by Cosine	$s(t)\cos(2\pi f_0 t)$	$\frac{S(f-f_0)+S(f+f_0)}{2}$
Amplitude Modulation by Sine	$s(t)\sin(2\pi f_0 t)$	$\frac{S(f-f_0)-S(f+f_0)}{2i}$
Differentiation	$\frac{d}{dt}s(t)$	$i2\pi f S(f)$
Integration	$\int_{-\infty}^{t}s(\alpha)\,d\alpha$	$\frac{1}{i2\pi f}S(f)$ if $S(0)=0$
Multiplication by t	$ts(t)$	$\frac{1}{-(i2\pi)}\frac{dS(f)}{df}$
Area	$\int_{-\infty}^{\infty}s(t)\,dt$	$S(0)$
Value at Origin	$s(0)$	$\int_{-\infty}^{\infty}S(f)\,df$
Parseval's Theorem	$\int_{-\infty}^{\infty}(\lvert s(t)\rvert)^2dt$	$\int_{-\infty}^{\infty}(\lvert S(f)\rvert)^2df$

Table 1.2

Example 1.15

In communications, a very important operation on a signal $s(t)$ is to **amplitude modulate** it. Using this operation more as an example rather than elaborating the communications aspects here, we want to compute the Fourier transform — the spectrum — of

$$(1+s(t))\cos(2\pi f_c t)$$

Thus,

$$(1 + s(t)) \cos(2\pi f_c t) = \cos(2\pi f_c t) + s(t) \cos(2\pi f_c t)$$

For the spectrum of $\cos(2\pi f_c t)$, we use the Fourier series. Its period is $\frac{1}{f_c}$, and its only nonzero Fourier coefficients are $c_{\pm 1} = \frac{1}{2}$. The second term is **not** periodic unless $s(t)$ has the same period as the sinusoid. Using Euler's relation, the spectrum of the second term can be derived as

$$s(t) \cos(2\pi f_c t) = \int_{-\infty}^{\infty} S(f) e^{i2\pi f t} df \cos(2\pi f_c t)$$

Using Euler's relation for the cosine,

$$(s(t) \cos(2\pi f_c t)) = \frac{1}{2} \int_{-\infty}^{\infty} S(f) e^{i2\pi (f+f_c) t} df + \frac{1}{2} \int_{-\infty}^{\infty} S(f) e^{i2\pi (f-f_c) t} df$$

$$(s(t) \cos(2\pi f_c t)) = \frac{1}{2} \int_{-\infty}^{\infty} S(f - f_c) e^{i2\pi f t} df + \frac{1}{2} \int_{-\infty}^{\infty} S(f + f_c) e^{i2\pi f t} df$$

$$(s(t) \cos(2\pi f_c t)) = \int_{-\infty}^{\infty} \frac{S(f - f_c) + S(f + f_c)}{2} e^{i2\pi f t} df$$

Exploiting the uniqueness property of the Fourier transform, we have

$$\mathcal{F}(s(t) \cos(2\pi f_c t)) = \frac{S(f - f_c) + S(f + f_c)}{2} \tag{1.123}$$

This component of the spectrum consists of the original signal's spectrum delayed and advanced **in frequency**. The spectrum of the amplitude modulated signal is shown in Figure 1.50.

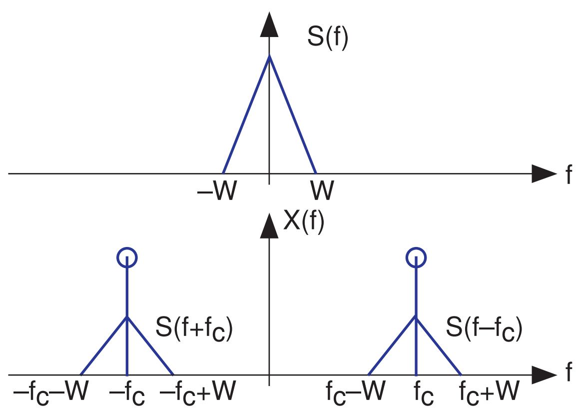


Figure 1.50: A signal which has a triangular shaped spectrum is shown in the top plot. Its highest frequency — the largest frequency containing power — is W Hz. Once amplitude modulated, the resulting spectrum has "lines" corresponding to the Fourier series components at $\pm(f_c)$ and the original triangular spectrum shifted to components at $\pm(f_c)$ and scaled by $\frac{1}{2}$.

Note how in this figure the signal $s(t)$ is defined in the frequency domain. To find its time domain representation, we simply use the inverse Fourier transform.

Exercise 1.4.8.3 ***(Solution on p. 140.)***

What is the signal $s(t)$ that corresponds to the spectrum shown in the upper panel of Figure 1.50?

Exercise 1.4.8.4 *(Solution on p. 140.)*

What is the power in $x(t)$, the amplitude-modulated signal? Try the calculation in both the time and frequency domains.

In this example, we call the signal $s(t)$ a **baseband signal** because its power is contained at low frequencies. Signals such as speech and the Dow Jones averages are baseband signals. The baseband signal's **bandwidth** equals W, the highest frequency at which it has power. Since $x(t)$'s spectrum is confined to a frequency band not close to the origin (we assume $f_c \gg W$), we have a **bandpass signal**. The bandwidth of a bandpass signal is **not** its highest frequency, but the range of positive frequencies where the signal has power. Thus, in this example, the bandwidth is $2W$Hz. Why a signal's bandwidth should depend on its spectral shape will become clear once we develop communications systems.

1.4.9 Linear Time Invariant Systems[38]

When we apply a periodic input to a linear, time-invariant system, the output is periodic and has Fourier series coefficients equal to the product of the system's frequency response and the input's Fourier coefficients (Filtering Periodic Signals (1.113)). The way we derived the spectrum of non-periodic signal from periodic ones makes it clear that the same kind of result works when the input is not periodic: **If $x(t)$ serves as the input to a linear, time-invariant system having frequency response $H(f)$, the spectrum of the output is $X(f)H(f)$.**

Example 1.16

Let's use this frequency-domain input-output relationship for linear, time-invariant systems to find a formula for the RC-circuit's response to a pulse input. We have expressions for the input's spectrum and the system's frequency response.

$$P(f) = e^{-(i\pi f\Delta)}\frac{\sin(\pi f\Delta)}{\pi f} \tag{1.124}$$

$$H(f) = \frac{1}{1 + i2\pi fRC} \tag{1.125}$$

Thus, the output's Fourier transform equals

$$Y(f) = e^{-(i\pi f\Delta)}\frac{\sin(\pi f\Delta)}{\pi f}\frac{1}{1 + i2\pi fRC} \tag{1.126}$$

You won't find this Fourier transform in our table, and the required integral is difficult to evaluate as the expression stands. This situation requires cleverness and an understanding of the Fourier transform's properties. In particular, recall Euler's relation for the sinusoidal term and note the fact that multiplication by a complex exponential in the frequency domain amounts to a time delay. Let's momentarily make the expression for $Y(f)$ more complicated.

$$\begin{aligned} e^{-(i\pi f\Delta)}\frac{\sin(\pi f\Delta)}{\pi f} &= e^{-(i\pi f\Delta)}\frac{e^{i\pi f\Delta} - e^{-(i\pi f\Delta)}}{i2\pi f} \\ &= \frac{1}{i2\pi f}\left(1 - e^{-(i2\pi f\Delta)}\right) \end{aligned} \tag{1.127}$$

Consequently,

$$Y(f) = \frac{1}{i2\pi f}\left(1 - e^{-(i\pi f\Delta)}\right)\frac{1}{1 + i2\pi fRC} \tag{1.128}$$

The table of Fourier transform properties (Table 1.2: Fourier Transform Properties) suggests thinking about this expression as a **product** of terms.

- Multiplication by $\frac{1}{i2\pi f}$ means integration.

[38]This content is available online at <http://cnx.org/content/m0048/2.19/>.

- Multiplication by the complex exponential $e^{-(i2\pi f\Delta)}$ means delay by Δ seconds in the time domain.
- The term $1-e^{-(i2\pi f\Delta)}$ means, in the time domain, subtract the time-delayed signal from its original.
- The inverse transform of the frequency response is $\frac{1}{RC}e^{-\frac{t}{RC}}u(t)$.

We can translate each of these frequency-domain products into time-domain operations **in any order we like** because the order in which multiplications occur doesn't affect the result. Let's start with the product of $\frac{1}{i2\pi f}$ (integration in the time domain) and the transfer function:

$$\frac{1}{i2\pi f}\frac{1}{1+i2\pi fRC} \leftrightarrow \left(1-e^{-\frac{t}{RC}}\right)u(t) \tag{1.129}$$

The middle term in the expression for $Y(f)$ consists of the difference of two terms: the constant 1 and the complex exponential $e^{-(i2\pi f\Delta)}$. Because of the Fourier transform's linearity, we simply subtract the results.

$$Y(f) \leftrightarrow \left(1-e^{-\frac{t}{RC}}\right)u(t) - \left(1-e^{-\frac{t-\Delta}{RC}}\right)u(t-\Delta) \tag{1.130}$$

Note that in delaying the signal how we carefully included the unit step. The second term in this result does not begin until $t=\Delta$. Thus, the waveforms shown in the Filtering Periodic Signals (Figure 1.48: Filtering a periodic signal) example mentioned above are exponentials. We say that the **time constant** of an exponentially decaying signal equals the time it takes to decrease by $\frac{1}{e}$ of its original value. Thus, the time-constant of the rising and falling portions of the output equal the product of the circuit's resistance and capacitance.

Exercise 1.4.9.1 ***(Solution on p. 140.)***

Derive the filter's output by considering the terms in (1.127) in the order given. Integrate last rather than first. You should get the same answer.

In this example, we used the table extensively to find the inverse Fourier transform, relying mostly on what multiplication by certain factors, like $\frac{1}{i2\pi f}$ and $e^{-(i2\pi f\Delta)}$, meant. We essentially treated multiplication by these factors as if they were transfer functions of some fictitious circuit. The transfer function $\frac{1}{i2\pi f}$ corresponded to a circuit that integrated, and $e^{-(i2\pi f\Delta)}$ to one that delayed. We even implicitly interpreted the circuit's transfer function as the input's spectrum! This approach to finding inverse transforms – breaking down a complicated expression into products and sums of simple components – is the engineer's way of breaking down the problem into several subproblems that are much easier to solve and then gluing the results together. Along the way we may make the system serve as the input, but in the rule $Y(f)=X(f)H(f)$, which term is the input and which is the transfer function is merely a notational matter (we labeled one factor with an X and the other with an H).

1.4.9.1 Transfer Functions

The notion of a transfer function applies well beyond linear circuits. Although we don't have all we need to demonstrate the result as yet, **all** linear, time-invariant systems have a frequency-domain input-output relation given by the product of the input's Fourier transform and the system's transfer function. Thus, linear circuits are a special case of linear, time-invariant systems. As we tackle more sophisticated problems in transmitting, manipulating, and receiving information, we will assume linear systems having certain properties (transfer functions) **without** worrying about what circuit has the desired property. At this point, you may be concerned that this approach is glib, and rightly so. Later we'll show that by involving software that we really don't need to be concerned about constructing a transfer function from circuit elements and op-amps.

1.4.9.2 Commutative Transfer Functions

Another interesting notion arises from the commutative property of multiplication (exploited in an example above (Example 1.16)): We can rather arbitrarily choose an order in which to apply each product. Consider a cascade of two linear, time-invariant systems. Because the Fourier transform of the first system's output is $X(f)\,H_1(f)$ and it serves as the second system's input, the cascade's output spectrum is $X(f)\,H_1(f)\,H_2(f)$. Because this product also equals $X(f)\,H_2(f)\,H_1(f)$, the **cascade having the linear systems in the opposite order yields the same result**. Furthermore, the cascade acts like a **single** linear system, having transfer function $H_1(f)\,H_2(f)$. This result applies to other configurations of linear, time-invariant systems as well; see this Frequency Domain Problem (Problem 1.13). Engineers exploit this property by determining what transfer function they want, then breaking it down into components arranged according to standard configurations. Using the fact that op-amp circuits can be connected in cascade with the transfer function equaling the product of its component's transfer function (see this analog signal processing problem[39]), we find a ready way of realizing designs. We now understand why op-amp implementations of transfer functions are so important.

1.4.10 Frequency Domain Problems[40]

Problem 1.1: Simple Fourier Series
Find the complex Fourier series representations of the following signals without explicitly calculating Fourier integrals. What is the signal's period in each case?

a) $s(t) = \sin(t)$
b) $s(t) = \sin^2(t)$
c) $s(t) = \cos(t) + 2\cos(2t)$
d) $s(t) = \cos(2t)\cos(t)$
e) $s(t) = \cos\left(10\pi t + \frac{\pi}{6}\right)(1 + \cos(2\pi t))$
f) $s(t)$ given by the depicted waveform (Figure 1.51).

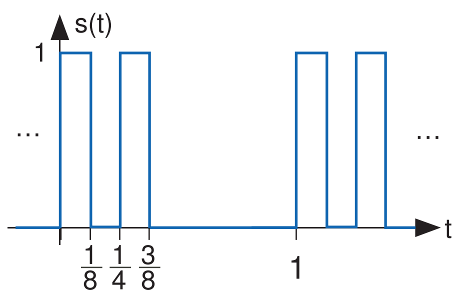


Figure 1.51

Problem 1.2: Fourier Series
Find the Fourier series representation for the following periodic signals (Figure 1.52). For the third signal, find the complex Fourier series for the triangle wave **without** performing the usual Fourier integrals. Hint: How is this signal related to one for which you already have the series?

[39]"Analog Signal Processing Problems", Problem 44 <http://cnx.org/content/m10349/latest/#i25>
[40]This content is available online at <http://cnx.org/content/m10350/2.43/>.

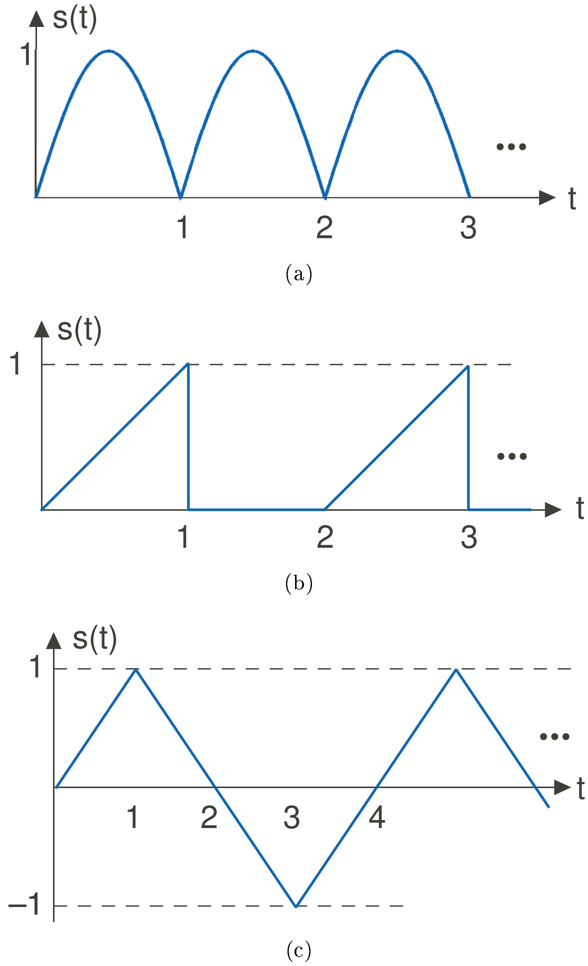


Figure 1.52

Problem 1.3: Phase Distortion

We can learn about phase distortion by returning to circuits and investigate the following circuit (Figure 1.53).

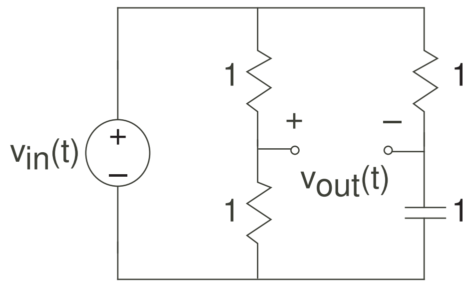


Figure 1.53

a) Find this filter's transfer function.
b) Find the magnitude and phase of this transfer function. How would you characterize this circuit?
c) Let $v_{\mathrm{in}}(t)$ be a square-wave of period T. What is the Fourier series for the output voltage?
d) Use Matlab to find the output's waveform for the cases $T = 0.01$ and $T = 2$. What value of T delineates the two kinds of results you found? The software in `fourier2.m` might be useful.
e) Instead of the depicted circuit, the square wave is passed through a system that delays its input, which applies a linear phase shift to the signal's spectrum. Let the delay τ be $\frac{T}{4}$. Use the transfer function of a delay to compute using Matlab the Fourier series of the output. Show that the square wave is indeed delayed.

Problem 1.4: Approximating Periodic Signals
Often, we want to approximate a reference signal by a somewhat simpler signal. To assess the quality of an approximation, the most frequently used error measure is the mean-squared error. For a periodic signal $s(t)$,

$$\epsilon^2 = \frac{1}{T}\int_0^T (s(t) - \tilde{s}(t))^2 dt$$

where $s(t)$ is the reference signal and $\tilde{s}(t)$ its approximation. One convenient way of finding approximations for periodic signals is to truncate their Fourier series.

$$\tilde{s}(t) = \sum_{k=-K}^{K} c_k e^{i\frac{2\pi k}{T}t}$$

The point of this problem is to analyze whether this approach is the best (*i.e.*, always minimizes the mean-squared error).

a) Find a frequency-domain expression for the approximation error when we use the truncated Fourier series as the approximation.
b) Instead of truncating the series, let's generalize the nature of the approximation to including any set of $2K + 1$ terms: We'll always include the c_0 and the negative indexed term corresponding to c_k. What selection of terms minimizes the mean-squared error? Find an expression for the mean-squared error resulting from your choice.
c) Find the Fourier series for the depicted signal (Figure 1.54). Use Matlab to find the truncated approximation and best approximation involving two terms. Plot the mean-squared error as a function of K for both approximations.

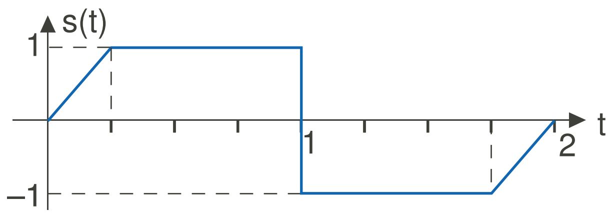


Figure 1.54

Problem 1.5: Long, Hot Days
The daily temperature is a consequence of several effects, one of them being the sun's heating. If this were the dominant effect, then daily temperatures would be proportional to the number of daylight hours. The plot (Figure 1.55) shows that the average daily high temperature does **not** behave that way.

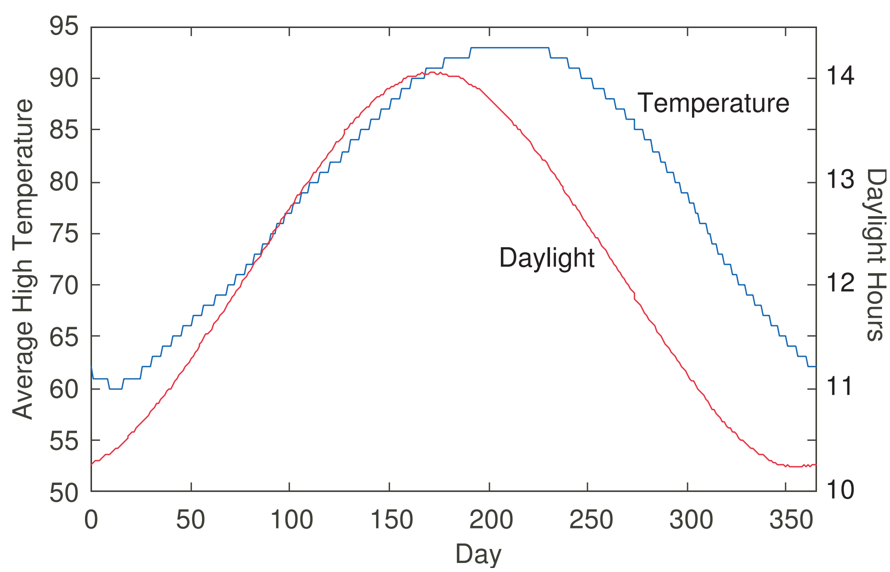


Figure 1.55

In this problem, we want to understand the temperature component of our environment using Fourier series and linear system theory. The file `temperature.mat` contains these data (daylight hours in the first row, corresponding average daily highs in the second) for Houston, Texas.

a) Let the length of day serve as the sole input to a system having an output equal to the average daily temperature. Examining the plots of input and output, would you say that the system is linear or not? How did you reach you conclusion?

b) Find the first five terms (c_0, ... , c_4) of the complex Fourier series for each signal. Use the following formula that approximates the integral required to find the Fourier coefficients.

$$c_k = \frac{1}{366}\sum_{n=0}^{366} s(n)\, e^{-\left(i\frac{2\pi nk}{366}\right)}$$

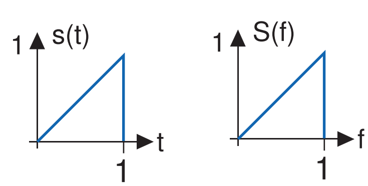
c) What is the harmonic distortion in the two signals? Exclude c_0 from this calculation.
d) Because the harmonic distortion is small, let's concentrate only on the first harmonic. What is the phase shift between input and output signals?
e) Find the transfer function of the simplest possible linear model that would describe the data. Characterize and interpret the structure of this model. In particular, give a physical explanation for the phase shift.
f) Predict what the output would be if the model had no phase shift. Would days be hotter? If so, by how much?

Problem 1.6: Fourier Transform Pairs
Find the Fourier or inverse Fourier transform of the following.

a) $\forall t : \left(x(t) = e^{-(a|t|)}\right)$
b) $x(t) = te^{-(at)}u(t)$
c) $X(f) = \begin{cases} 1 & \text{if } |f| < W \\ 0 & \text{if } |f| > W \end{cases}$
d) $x(t) = e^{-(at)}\cos(2\pi f_0 t)\, u(t)$

Problem 1.7: Duality in Fourier Transforms
"Duality" means that the Fourier transform and the inverse Fourier transform are very similar. Consequently, the waveform $s(t)$ in the time domain and the spectrum $s(f)$ have a Fourier transform and an inverse Fourier transform, respectively, that are very similar.

a) Calculate the Fourier transform of the signal shown below (Figure 1.56(a)).
b) Calculate the inverse Fourier transform of the spectrum shown below (Figure 1.56(b)).
c) How are these answers related? What is the general relationship between the Fourier transform of $s(t)$ and the inverse transform of $s(f)$?



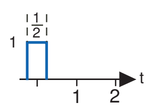
(a) (b)

Figure 1.56

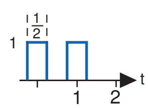
Problem 1.8: Spectra of Pulse Sequences

Pulse sequences occur often in digital communication and in other fields as well. What are their spectral properties?

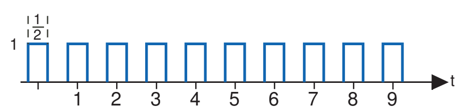
a) Calculate the Fourier transform of the single pulse shown below (Figure 1.57(a)).
b) Calculate the Fourier transform of the two-pulse sequence shown below (Figure 1.57(b)).
c) Calculate the Fourier transform for the **ten**-pulse sequence shown in below (Figure 1.57(c)). You should look for a general expression that holds for sequences of any length.
d) Using Matlab, plot the magnitudes of the three spectra. Describe how the spectra change as the number of repeated pulses increases.



(a)



(b)



(c)

Figure 1.57

Problem 1.9: Spectra of Digital Communication Signals

One way to represent bits with signals is shown in Figure 1.58. If the value of a bit is a "1", it is represented by a positive pulse of duration T. If it is a "0", it is represented by a negative pulse of the same duration. To represent a sequence of bits, the appropriately chosen pulses are placed one after the other.

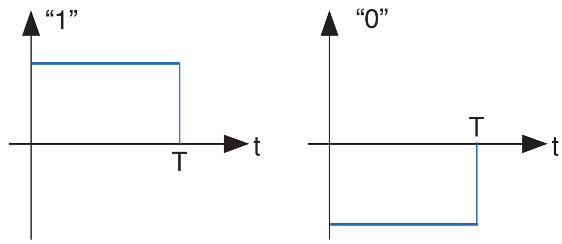


Figure 1.58

a) What is the spectrum of the waveform that represents the alternating bit sequence "...01010101..."?
b) This signal's bandwidth is defined to be the frequency range over which 90% of the power is contained. What is this signal's bandwidth?
c) Suppose the bit sequence becomes "...00110011...". Now what is the bandwidth?

Problem 1.10: Lowpass Filtering a Square Wave
Let a square wave (period T) serve as the input to a first-order lowpass system constructed as a RC filter. We want to derive an expression for the time-domain response of the filter to this input.

a) First, consider the response of the filter to a simple pulse, having unit amplitude and width $\frac{T}{2}$. Derive an expression for the filter's output to this pulse.
b) Noting that the square wave is a superposition of a sequence of these pulses, what is the filter's response to the square wave?
c) The nature of this response should change as the relation between the square wave's period and the filter's cutoff frequency change. How long must the period be so that the response does **not** achieve a relatively constant value between transitions in the square wave? What is the relation of the filter's cutoff frequency to the square wave's spectrum in this case?

Problem 1.11: Mathematics with Circuits
Simple circuits can implement simple mathematical operations, such as integration and differentiation. We want to develop an active circuit (it contains an op-amp) having an output that is proportional to the integral of its input. For example, you could use an integrator in a car to determine distance traveled from the speedometer.

a) What is the transfer function of an integrator?
b) Find an op-amp circuit so that its voltage output is proportional to the integral of its input for all signals.

Problem 1.12: Where is that sound coming from?
We determine where sound is coming from because we have two ears and a brain. Sound travels at a relatively slow speed and our brain uses the fact that sound will arrive at one ear before the other. As shown here (Figure 1.59), a sound coming from the right arrives at the left ear τ seconds after it arrives at the right ear.

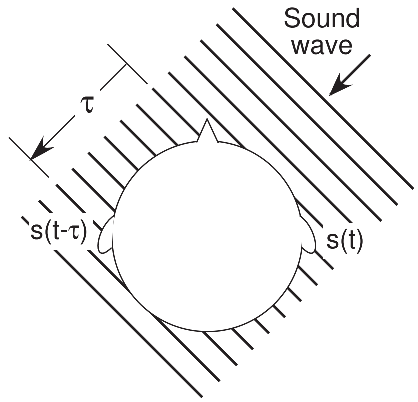
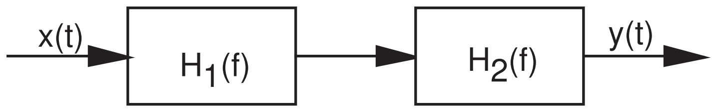


Figure 1.59

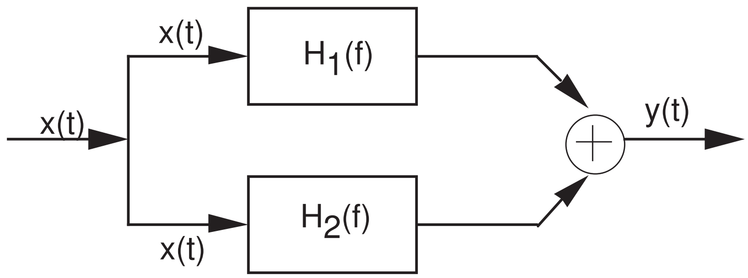
Once the brain finds this propagation delay, it can determine the sound direction. In an attempt to model what the brain might do, RU signal processors want to design an **optimal** system that delays each ear's signal by some amount then adds them together. Δ_l and Δ_r are the delays applied to the left and right signals respectively. The idea is to determine the delay values according to some criterion that is based on what is measured by the two ears.

a) What is the transfer function between the sound signal $s(t)$ and the processor output $y(t)$?
b) One way of determining the delay τ is to choose Δ_l and Δ_r to maximize the power in $y(t)$. How are these maximum-power processing delays related to τ?

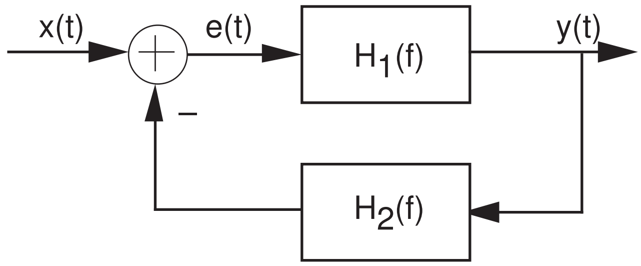
Problem 1.13: Arrangements of Systems
Architecting a system of modular components means arranging them in various configurations to achieve some overall input-output relation. For each of the following (Figure 1.60), determine the overall transfer function between $x(t)$ and $y(t)$.



(a) system a



(b) system b



(c) system c

Figure 1.60

The overall transfer function for the cascade (first depicted system) is particularly interesting. What does it say about the effect of the ordering of linear, time-invariant systems in a cascade?

Problem 1.14: Filtering
Let the signal $s(t) = \frac{\sin(\pi t)}{\pi t}$ be the input to a linear, time-invariant filter having the transfer function shown below (Figure 1.61). Find the expression for $y(t)$, the filter's output.

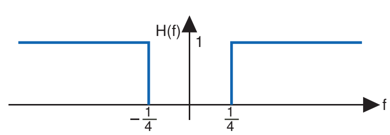


Figure 1.61

Problem 1.15: Circuits Filter!
A unit-amplitude pulse with duration of one second serves as the input to an RC-circuit having transfer function

$$H(f) = \frac{i2\pi f}{4 + i2\pi f}$$

a) How would you categorize this transfer function: lowpass, highpass, bandpass, other?
b) Find a circuit that corresponds to this transfer function.
c) Find an expression for the filter's output.

Problem 1.16: Reverberation
Reverberation corresponds to adding to a signal its delayed version.

a) Assuming τ represents the delay, what is the input-output relation for a reverberation system? Is the system linear and time-invariant? If so, find the transfer function; if not, what linearity or time-invariance criterion does reverberation violate.
b) A music group known as the ROwls is having trouble selling its recordings. The record company's engineer gets the idea of applying different delay to the low and high frequencies and adding the result to create a new musical effect. Thus, the ROwls' audio would be separated into two parts (one less than the frequency f_0, the other greater than f_0), these would be delayed by τ_l and τ_h respectively, and the resulting signals added. Draw a block diagram for this new audio processing system, showing its various components.
c) How does the magnitude of the system's transfer function depend on the two delays?

Problem 1.17: Echoes in Telephone Systems
A frequently encountered problem in telephones is echo. Here, because of acoustic coupling between the ear piece and microphone in the handset, what you hear is also sent to the person talking. That person thus not only hears you, but also hears her own speech delayed (because of propagation delay over the telephone network) and attenuated (the acoustic coupling gain is less than one). Furthermore, the same problem applies to you as well: The acoustic coupling occurs in her handset as well as yours.

a) Develop a block diagram that describes this situation.
b) Find the transfer function between your voice and what the listener hears.
c) Each telephone contains a system for reducing echoes using electrical means. What simple system could null the echoes?

Problem 1.18: Effective Drug Delivery
In most patients, it takes time for the concentration of an administered drug to achieve a constant level in the blood stream. Typically, if the drug concentration in the patient's intravenous line is $C_d u(t)$, the concentration in the patient's blood stream is $C_p \left(1 - e^{-(at)}\right) u(t)$.

a) Assuming the relationship between drug concentration in the patient's drug and the delivered concentration can be described as a linear, time-invariant system, what is the transfer function?
b) Sometimes, the drug delivery system goes awry and delivers drugs with little control. What would the patient's drug concentration be if the delivered concentration were a ramp? More precisely, if it were $C_d t u(t)$?
c) A clever doctor wants to have the flexibility to slow down or speed up the patient's drug concentration. In other words, the concentration is to be $C_p \left(1 - e^{-(bt)}\right) u(t)$, with b bigger or smaller than a. How should the delivered drug concentration signal be changed to achieve this concentration profile?

Problem 1.19: Catching Speeders with Radar
RU Electronics has been contracted to design a Doppler radar system. Radar transmitters emit a signal that bounces off any conducting object. Signal differences between what is sent and the radar return is processed and features of interest extracted. In **Doppler** systems, the object's speed along the direction of the radar beam is the feature the design must extract. The transmitted signal is a sinsusoid: $x(t) = A\cos(2\pi f_c t)$. The measured return signal equals $B\cos(2\pi((f_c + \Delta \mathrm{f})t + \varphi))$, where the Doppler offset frequency Δf equals $10v$, where v is the car's velocity coming toward the transmitter.

a) Design a system that uses the transmitted and return signals as inputs and produces Δf.
b) One problem with designs based on overly simplistic design goals is that they are sensitive to unmodeled assumptions. How would you change your design, if at all, so that whether the car is going away or toward the transmitter could be determined?
c) Suppose two objects traveling different speeds provide returns. How would you change your design, if at all, to accomodate multiple returns?

Problem 1.20: Demodulating an AM Signal
Let $m(t)$ denote the signal that has been amplitude modulated.

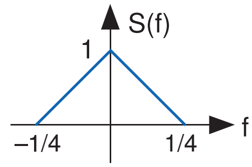
$$x(t) = A(1 + m(t))\sin(2\pi f_c t)$$

Radio stations try to restrict the amplitude of the signal $m(t)$ so that it is less than one in magnitude. The frequency f_c is very large compared to the frequency content of the signal. What we are concerned about here is not transmission, but reception.

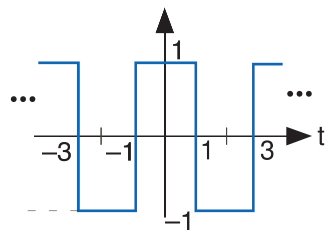
a) The so-called coherent demodulator simply multiplies the signal $x(t)$ by a sinusoid having the same frequency as the carrier and lowpass filters the result. Analyze this receiver and show that it works. Assume the lowpass filter is ideal.
b) One issue in coherent reception is the phase of the sinusoid used by the receiver relative to that used by the transmitter. Assuming that the sinusoid of the receiver has a phase ϕ, how does the output depend on ϕ? What is the worst possible value for this phase?
c) The incoherent receiver is more commonly used because of the phase sensitivity problem inherent in coherent reception. Here, the receiver full-wave rectifies the received signal and lowpass filters the result (again ideally). Analyze this receiver. Does its output differ from that of the coherent receiver in a significant way?

Problem 1.21: Unusual Amplitude Modulation
We want to send a band-limited signal having the depicted spectrum (Figure 1.62(a)) with amplitude modulation in the usual way. I.B. Different suggests using the square-wave carrier shown below (Figure 1.62(b)). Well, it is different, but his friends wonder if any technique can demodulate it.

a) Find an expression for $X(f)$, the Fourier transform of the modulated signal.
b) Sketch the magnitude of $X(f)$, being careful to label important magnitudes and frequencies.
c) What demodulation technique obviously works?
d) I.B. challenges three of his friends to demodulate $x(t)$ some other way. One friend suggests modulating $x(t)$ with $\cos\left(\frac{\pi t}{2}\right)$, another wants to try modulating with $\cos(\pi t)$ and the third thinks $\cos\left(\frac{3\pi t}{2}\right)$ will work. Sketch the magnitude of the Fourier transform of the signal each student's approach produces. Which student comes closest to recovering the original signal? Why?



(a)



(b)

Figure 1.62

Problem 1.22: Sammy Falls Asleep...
While sitting in ELEC 241 class, he falls asleep during a critical time when an AM receiver is being described. The received signal has the form $r(t) = A(1 + m(t))\cos(2\pi f_c t + \phi)$ where the phase ϕ is unknown. The message signal is $m(t)$; it has a bandwidth of W Hz and a magnitude less than 1 ($|m(t)| < 1$). The phase ϕ is unknown. The instructor drew a diagram (Figure 1.63) for a receiver on the board; Sammy slept through the description of what the unknown systems where.

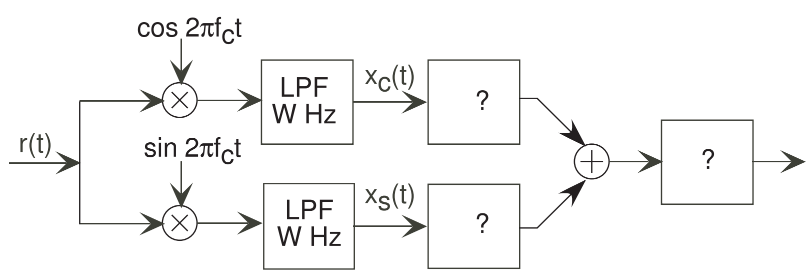


Figure 1.63

a) What are the signals $x_c(t)$ and $x_s(t)$?
b) What would you put in for the unknown systems that would guarantee that the final output contained the message regardless of the phase?

   HINT: Think of a trigonometric identity that would prove useful.

c) Sammy may have been asleep, but he can think of a far simpler receiver. What is it?

Problem 1.23: Jamming
Sid Richardson college decides to set up its own AM radio station KSRR. The resident electrical engineer decides that she can choose **any** carrier frequency and message bandwidth for the station. A rival college decides to **jam** its transmissions by transmitting a high-power signal that interferes with radios that try to receive KSRR. The jamming signal jam (t) is what is known as a **sawtooth** wave (depicted in Figure 1.64) having a period known to KSRR's engineer.

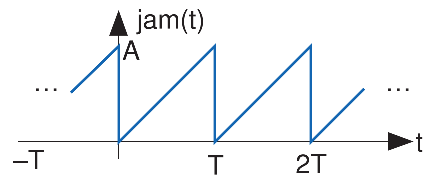


Figure 1.64

a) Find the spectrum of the jamming signal.
b) Can KSRR entirely circumvent the attempt to jam it by carefully choosing its carrier frequency and transmission bandwidth? If so, find the station's carrier frequency and transmission bandwidth in terms of T, the period of the jamming signal; if not, show why not.

Problem 1.24: AM Stereo
A stereophonic signal consists of a "left" signal $l(t)$ and a "right" signal $r(t)$ that conveys sounds coming from an orchestra's left and right sides, respectively. To transmit these two signals simultaneously, the transmitter first forms the sum signal $s_+(t) = l(t) + r(t)$ and the difference signal $s_-(t) = l(t) - r(t)$. Then, the transmitter amplitude-modulates the difference signal with a sinusoid having frequency $2W$, where W is the bandwidth of the left and right signals. The sum signal and the modulated difference signal are added, the sum amplitude-modulated to the radio station's carrier frequency f_c, and transmitted. Assume the spectra of the left and right signals are as shown (Figure 1.65).

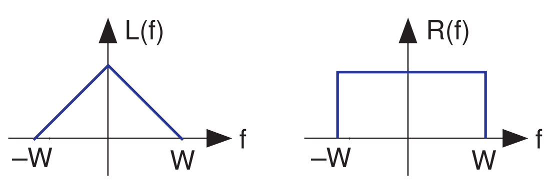


Figure 1.65

a) What is the expression for the transmitted signal? Sketch its spectrum.

b) Show the block diagram of a stereo AM receiver that can yield the left and right signals as separate outputs.
c) What signal would be produced by a conventional coherent AM receiver that expects to receive a standard AM signal conveying a message signal having bandwidth W?

Problem 1.25: Novel AM Stereo Method
A clever engineer has submitted a patent for a new method for transmitting two signals **simultaneously** in the **same** transmission bandwidth as commercial AM radio. As shown (Figure 1.66), her approach is to modulate the positive portion of the carrier with one signal and the negative portion with a second.

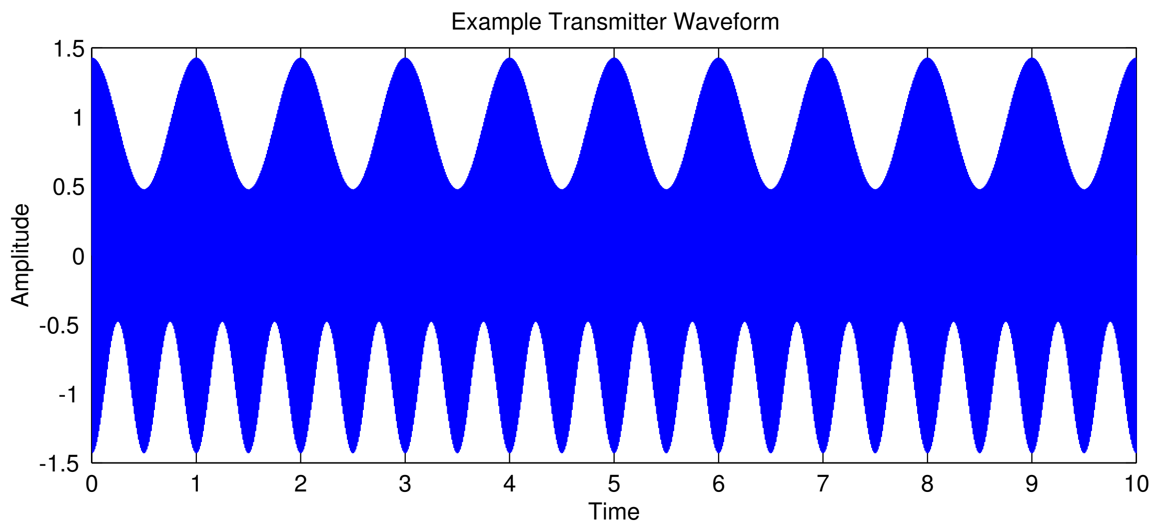


Figure 1.66

In detail the two message signals $m_1(t)$ and $m_2(t)$ are bandlimited to W Hz and have maximal amplitudes equal to 1. The carrier has a frequency f_c much greater than W. The transmitted signal $x(t)$ is given by

$$x(t) = \begin{cases} A(1 + am_1(t))\sin(2\pi f_c t) & \text{if } \sin(2\pi f_c t) \geq 0 \\ A(1 + am_2(t))\sin(2\pi f_c t) & \text{if } \sin(2\pi f_c t) < 0 \end{cases}$$

In all cases, $0 < a < 1$. The plot shows the transmitted signal when the messages are sinusoids: $m_1(t) = \sin(2\pi f_m t)$ and $m_2(t) = \sin(2\pi 2 f_m t)$ where $2f_m < W$. You, as the patent examiner, must determine whether the scheme meets its claims and is useful.

a) Provide a more concise expression for the transmitted signal $x(t)$ than given above.
b) What is the receiver for this scheme? It would yield both $m_1(t)$ and $m_2(t)$ from $x(t)$.
c) Find the spectrum of the positive portion of the transmitted signal.
d) Determine whether this scheme satisfies the design criteria, allowing you to grant the patent. Explain your reasoning.

Problem 1.26: A Radical Radio Idea
An ELEC 241 student has the bright idea of using a square wave instead of a sinusoid as an AM carrier. The transmitted signal would have the form

$$x(t) = A(1 + m(t))\,\mathrm{sq}_T(t)$$

where the message signal $m(t)$ would be amplitude-limited: $|m(t)| < 1$

a) Assuming the message signal is lowpass and has a bandwidth of W Hz, what values for the square wave's period T are feasible. In other words, do some combinations of W and T prevent reception?
b) Assuming reception is possible, can **standard** radios receive this innovative AM transmission? If so, show how a coherent receiver could demodulate it; if not, show how the coherent receiver's output would be corrupted. Assume that the message bandwidth $W = 5$ kHz.

Problem 1.27: Secret Communication
An amplitude-modulated secret message $m(t)$ has the following form.

$$r(t) = A(1 + m(t))\cos(2\pi(f_c + f_0)t)$$

The message signal has a bandwidth of W Hz and a magnitude less than 1 ($|m(t)| < 1$). The idea is to offset the carrier frequency by f_0 Hz from standard radio carrier frequencies. Thus, "off-the-shelf" coherent demodulators would assume the carrier frequency has f_c Hz. Here, $f_0 < W$.

a) Sketch the spectrum of the demodulated signal produced by a coherent demodulator tuned to f_c Hz.
b) Will this demodulated signal be a "scrambled" version of the original? If so, how so; if not, why not?
c) Can you develop a receiver that can demodulate the message without knowing the offset frequency f_c?

Problem 1.28: Signal Scrambling
An excited inventor announces the discovery of a way of using analog technology to render music unlistenable without knowing the secret recovery method. The idea is to modulate the bandlimited message $m(t)$ by a special periodic signal $s(t)$ that is zero during half of its period, which renders the message unlistenable and superficially, at least, unrecoverable (Figure 1.67).

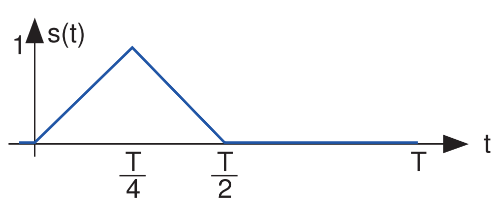


Figure 1.67

a) What is the Fourier series for the periodic signal?
b) What are the restrictions on the period T so that the message signal can be recovered from $m(t)\,s(t)$?
c) ELEC 241 students think they have "broken" the inventor's scheme and are going to announce it to the world. How would they recover the original message **without** having detailed knowledge of the modulating signal?

1.5 Continuous Time Fourier Transform (CTFT)

1.5.1 Continuous Time Fourier Transform (CTFT)[41]

1.5.1.1 Introduction

In this module, we will derive an expansion for any arbitrary continuous-time function, and in doing so, derive the **Continuous Time Fourier Transform** (CTFT).

Since complex exponentials (Section 1.1.5) are eigenfunctions of linear time-invariant (LTI) systems[42], calculating the output of an LTI system $\mathcal{H}$ given e^{st} as an input amounts to simple multiplication, where $H(s) \in \mathbb{C}$ is the eigenvalue corresponding to s. As shown in the figure, a simple exponential input would yield the output

$$y(t) = H(s) e^{st} \tag{1.131}$$

Image not finished

Using this and the fact that $\mathcal{H}$ is linear, calculating $y(t)$ for combinations of complex exponentials is also straightforward.

$$c_1 e^{s_1 t} + c_2 e^{s_2 t} \to c_1 H(s_1) e^{s_1 t} + c_2 H(s_2) e^{s_2 t}$$

$$\sum_n c_n e^{s_n t} \to \sum_n c_n H(s_n) e^{s_n t}$$

The action of H on an input such as those in the two equations above is easy to explain. $\mathcal{H}$ **independently scales** each exponential component $e^{s_n t}$ by a different complex number $H(s_n) \in \mathbb{C}$. As such, if we can write a function $f(t)$ as a combination of complex exponentials it allows us to easily calculate the output of a system.

Now, we will look to use the power of complex exponentials to see how we may represent arbitrary signals in terms of a set of simpler functions by superposition of a number of complex exponentials. Below we will present the **Continuous-Time Fourier Transform** (CTFT), commonly referred to as just the Fourier Transform (FT). Because the CTFT deals with nonperiodic signals, we must find a way to include all real frequencies in the general equations. For the CTFT we simply utilize integration over real numbers rather than summation over integers in order to express the aperiodic signals.

1.5.1.2 Fourier Transform Synthesis

Joseph Fourier[43] demonstrated that an arbitrary $s(t)$ can be written as a linear combination of harmonic complex sinusoids

$$s(t) = \sum_{n=-\infty}^{\infty} c_n e^{j\omega_0 n t} \tag{1.132}$$

where $\omega_0 = \frac{2\pi}{T}$ is the fundamental frequency. For almost all $s(t)$ of practical interest, there exists c_n to make (1.132) true. If $s(t)$ is finite energy ($s(t) \in L^2[0,T]$), then the equality in (1.132) holds in the sense of energy convergence; if $s(t)$ is continuous, then (1.132) holds pointwise. Also, if $s(t)$ meets some mild conditions (the Dirichlet conditions), then (1.132) holds pointwise everywhere except at points of discontinuity.

The c_n - called the Fourier coefficients - tell us "how much" of the sinusoid $e^{j\omega_0 n t}$ is in $s(t)$. The formula shows $s(t)$ as a sum of complex exponentials, each of which is easily processed by an LTI system (since it is an eigenfunction of **every** LTI system). Mathematically, it tells us that the set of complex exponentials $\left\{\forall n, n \in \mathbb{Z} : \left(e^{j\omega_0 n t}\right)\right\}$ form a basis for the space of T-periodic continuous time functions.

[41] This content is available online at <http://cnx.org/content/m10098/2.17/>.
[42] "Eigenfunctions of LTI Systems" <http://cnx.org/content/m10500/latest/>
[43] http://www-groups.dcs.st-and.ac.uk/~history/Mathematicians/Fourier.html

1.5.1.2.1 Equations

Now, in order to take this useful tool and apply it to arbitrary non-periodic signals, we will have to delve deeper into the use of the superposition principle. Let $s_T(t)$ be a periodic signal having period T. We want to consider what happens to this signal's spectrum as the period goes to infinity. We denote the spectrum for any assumed value of the period by $c_n(T)$. We calculate the spectrum according to the Fourier formula for a periodic signal, known as the Fourier Series (for more on this derivation, see the section on **Fourier Series**.)

$$c_n = \frac{1}{T}\int_0^T s(t)\, exp(-ß\omega_0 t)\; dt \tag{1.133}$$

where $\omega_0 = \frac{2\pi}{T}$ and where we have used a symmetric placement of the integration interval about the origin for subsequent derivational convenience. We vary the frequency index n proportionally as we increase the period. Define

$$S_T(f) \equiv Tc_n = \frac{1}{T}\int_0^T (S_T(f)\, exp(ß\omega_0 t)\; dt(1.134)$$

making the corresponding Fourier Series

$$s_T(t) = \sum_{-\infty}^{\infty} f(t)\, exp(ß\omega_0 t)\; \frac{1}{T} \tag{1.135}$$

As the period increases, the spectral lines become closer together, becoming a continuum. Therefore,

$$\lim_{T\to\infty} s_T(t) \equiv s(t) = \int_{-\infty}^{\infty} S(f)\, exp(ß\omega_0 t)\; df \tag{1.136}$$

with

$$S(f) = \int_{-\infty}^{\infty} s(t)\, exp(-ß\omega_0 t)\; dt \tag{1.137}$$

Continuous-Time Fourier Transform

$$\mathcal{F}(\Omega) = \int_{-\infty}^{\infty} f(t)\, e^{-(i\Omega t)} dt \tag{1.138}$$

Inverse CTFT

$$f(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \mathcal{F}(\Omega)\, e^{i\Omega t} d\Omega \tag{1.139}$$

WARNING: It is not uncommon to see the above formula written slightly different. One of the most common differences is the way that the exponential is written. The above equations use the radial frequency variable Ω in the exponential, where $\Omega = 2\pi f$, but it is also common to include the more explicit expression, $i2\pi ft$, in the exponential. Click here[44] for an overview of the notation used in Connexion's DSP modules.

Example 1.17

We know from Euler's formula that $cos(\omega t) + sin(\omega t) = \frac{1-j}{2}e^{j\omega t} + \frac{1+j}{2}e^{-j\omega t}$.

[44] "DSP notation" <http://cnx.org/content/m10161/latest/>

1.5.1.3 CTFT Definition Demonstration

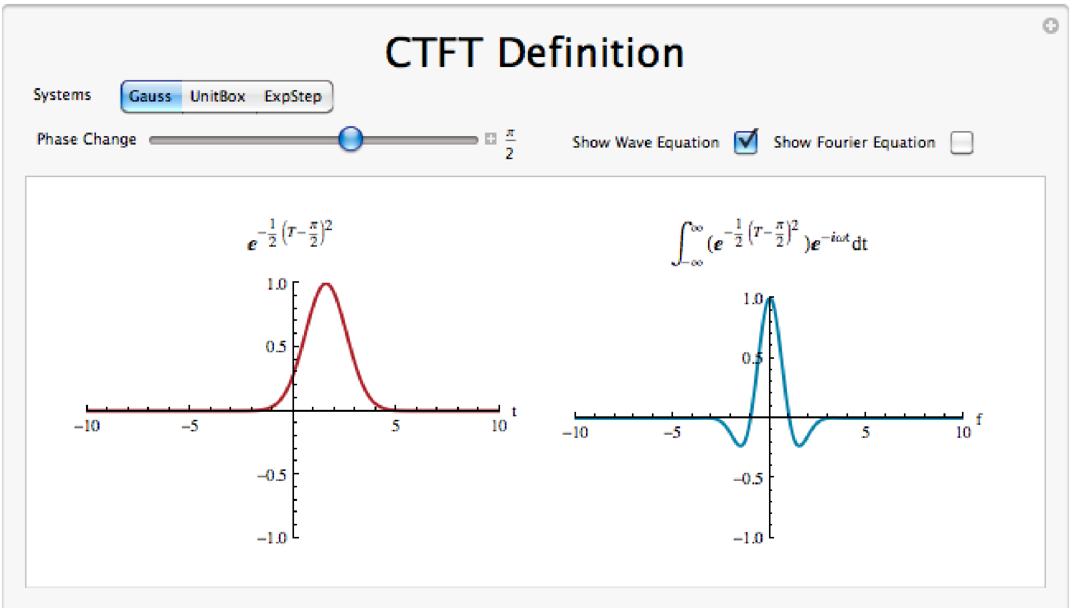


Figure 1.68: Interact (when online) with a Mathematica CDF demonstrating Continuous Time Fourier Transform. To Download, right-click and save as .cdf.

1.5.1.4 Example Problems

Exercise 1.5.1.1 *(Solution on p. 140.)*

Find the Fourier Transform (CTFT) of the function

$$f(t) = \begin{cases} e^{-(\alpha t)} & \text{if } t \geq 0 \\ 0 & \text{otherwise} \end{cases} \quad (1.140)$$

Exercise 1.5.1.2 *(Solution on p. 140.)*

Find the inverse Fourier transform of the ideal lowpass filter defined by

$$X(\Omega) = \begin{cases} 1 & \text{if } |\Omega| \leq M \\ 0 & \text{otherwise} \end{cases} \quad (1.141)$$

1.5.1.5 Fourier Transform Summary

Because complex exponentials are eigenfunctions of LTI systems, it is often useful to represent signals using a set of complex exponentials as a basis. The continuous time Fourier series synthesis formula expresses a continuous time, periodic function as the sum of continuous time, discrete frequency complex exponentials.

$$f(t) = \sum_{n=-\infty}^{\infty} c_n e^{j\omega_0 nt} \quad (1.142)$$

The continuous time Fourier series analysis formula gives the coefficients of the Fourier series expansion.

$$c_n = \frac{1}{T}\int_0^T f(t)\, e^{-(j\omega_0 nt)} dt \quad (1.143)$$

In both of these equations $\omega_0 = \frac{2\pi}{T}$ is the fundamental frequency.

1.5.2 Properties of the CTFT[45]

1.5.2.1 Introduction

This module will look at some of the basic properties of the Continuous-Time Fourier Transform (Section 1.5.1) (CTFT).

> NOTE: We will be discussing these properties for aperiodic, continuous-time signals but understand that very similar properties hold for discrete-time signals and periodic signals as well.

1.5.2.2 Discussion of Fourier Transform Properties

1.5.2.2.1 Linearity

The combined addition and scalar multiplication properties in the table above demonstrate the basic property of linearity. What you should see is that if one takes the Fourier transform of a linear combination of signals then it will be the same as the linear combination of the Fourier transforms of each of the individual signals. This is crucial when using a table (Section 1.8.7) of transforms to find the transform of a more complicated signal.

Example 1.18

We will begin with the following signal:

$$z(t) = af_1(t) + bf_2(t) \quad (1.144)$$

Now, after we take the Fourier transform, shown in the equation below, notice that the linear combination of the terms is unaffected by the transform.

$$Z(\omega) = aF_1(\omega) + bF_2(\omega) \quad (1.145)$$

1.5.2.2.2 Symmetry

Symmetry is a property that can make life quite easy when solving problems involving Fourier transforms. Basically what this property says is that since a rectangular function in time is a sinc function in frequency, then a sinc function in time will be a rectangular function in frequency. This is a direct result of the similarity between the forward CTFT and the inverse CTFT. The only difference is the scaling by 2π and a frequency reversal.

[45] This content is available online at <http://cnx.org/content/m10100/2.16/>.

1.5.2.2.3 Time Scaling

This property deals with the effect on the frequency-domain representation of a signal if the time variable is altered. The most important concept to understand for the time scaling property is that signals that are narrow in time will be broad in frequency and *vice versa*. The simplest example of this is a delta function, a unit pulse[46] with a **very** small duration, in time that becomes an infinite-length constant function in frequency.

The table above shows this idea for the general transformation from the time-domain to the frequency-domain of a signal. You should be able to easily notice that these equations show the relationship mentioned previously: if the time variable is increased then the frequency range will be decreased.

1.5.2.2.4 Time Shifting

Time shifting shows that a shift in time is equivalent to a linear phase shift in frequency. Since the frequency content depends only on the shape of a signal, which is unchanged in a time shift, then only the phase spectrum will be altered. This property is proven below:

Example 1.19

We will begin by letting $z(t) = f(t-\tau)$. Now let us take the Fourier transform with the previous expression substituted in for $z(t)$.

$$Z(\omega) = \int_{-\infty}^{\infty} f(t-\tau)\, e^{-(i\omega t)} dt \tag{1.146}$$

Now let us make a simple change of variables, where $\sigma = t - \tau$. Through the calculations below, you can see that only the variable in the exponential are altered thus only changing the phase in the frequency domain.

$$\begin{aligned} Z(\omega) &= \int_{-\infty}^{\infty} f(\sigma)\, e^{-(i\omega(\sigma+\tau)t)} d\tau \\ &= e^{-(i\omega\tau)} \int_{-\infty}^{\infty} f(\sigma)\, e^{-(i\omega\sigma)} d\sigma \\ &= e^{-(i\omega\tau)} F(\omega) \end{aligned} \tag{1.147}$$

1.5.2.2.5 Convolution

Convolution is one of the big reasons for converting signals to the frequency domain, since convolution in time becomes multiplication in frequency. This property is also another excellent example of symmetry between time and frequency. It also shows that there may be little to gain by changing to the frequency domain when multiplication in time is involved.

We will introduce the convolution integral here, but if you have not seen this before or need to refresh your memory, then look at the continuous-time convolution (Section 1.3.1) module for a more in depth explanation and derivation.

$$\begin{aligned} y(t) &= (f_1(t), f_2(t)) \\ &= \int_{-\infty}^{\infty} f_1(\tau)\, f_2(t-\tau)\, d\tau \end{aligned} \tag{1.148}$$

[46]"Elemental Signals": Section Pulse <http://cnx.org/content/m0004/latest/#pulsedef>

1.5.2.2.6 Time Differentiation

Since LTI (Section 1.2.1) systems can be represented in terms of differential equations, it is apparent with this property that converting to the frequency domain may allow us to convert these complicated differential equations to simpler equations involving multiplication and addition. This is often looked at in more detail during the study of the Laplace Transform[47].

1.5.2.2.7 Parseval's Relation

$$\int_{-\infty}^{\infty} (|f(t)|)^2 dt = \int_{-\infty}^{\infty} (|F(\omega)|)^2 df \tag{1.149}$$

Parseval's relation tells us that the energy of a signal is equal to the energy of its Fourier transform.

Figure 1.69

1.5.2.2.8 Modulation (Frequency Shift)

Modulation is absolutely imperative to communications applications. Being able to shift a signal to a different frequency, allows us to take advantage of different parts of the electromagnetic spectrum is what allows us to transmit television, radio and other applications through the same space without significant interference.

The proof of the frequency shift property is very similar to that of the time shift (Section 1.5.2.2.4: Time Shifting); however, here we would use the inverse Fourier transform in place of the Fourier transform. Since we went through the steps in the previous, time-shift proof, below we will just show the initial and final step to this proof:

$$z(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega - \phi) e^{i\omega t} d\omega \tag{1.150}$$

Now we would simply reduce this equation through another change of variables and simplify the terms. Then we will prove the property expressed in the table above:

$$z(t) = f(t) e^{i\phi t} \tag{1.151}$$

1.5.2.3 Properties Demonstration

An interactive example demonstration of the properties is included below:

[47] "The Laplace Transform" <http://cnx.org/content/m10110/latest/>

This media object is a LabVIEW VI. Please view or download it at
<CTFTSPlab.llb>

Figure 1.70: Interactive Signal Processing Laboratory Virtual Instrument created using NI's Labview.

1.5.2.4 Summary Table of CTFT Properties

Operation Name	**Signal (** $f(t)$ **)**	**Transform (** $F(\omega)$ **)**
Linearity (Section 1.5.2.2.1: Linearity)	$a(f_1, t) + b(f_2, t)$	$a(F_1, \omega) + b(F_2, \omega)$
Scalar Multiplication (Section 1.5.2.2.1: Linearity)	$\alpha f(t)$	$\alpha F(\omega)$
Symmetry (Section 1.5.2.2.2: Symmetry)	$F(t)$	$2\pi f(-\omega)$
Time Scaling (Section 1.5.2.2.3: Time Scaling)	$f(\alpha t)$	$\frac{1}{\lvert\alpha\rvert} F\left(\frac{\omega}{\alpha}\right)$
Time Shift (Section 1.5.2.2.4: Time Shifting)	$f(t-\tau)$	$F(\omega) e^{-(i\omega\tau)}$
Convolution in Time (Section 1.5.2.2.5: Convolution)	$(f_1(t), f_2(t))$	$F_1(t) F_2(t)$
Convolution in Frequency (Section 1.5.2.2.5: Convolution)	$f_1(t) f_2(t)$	$\frac{1}{2\pi}(F_1(t), F_2(t))$
Differentiation (Section 1.5.2.2.6: Time Differentiation)	$\frac{d^n}{dt^n} f(t)$	$(i\omega)^n F(\omega)$
Parseval's Theorem (Section 1.5.2.2.7: Parseval's Relation)	$\int_{-\infty}^{\infty} (\lvert f(t)\rvert)^2 dt$	$\int_{-\infty}^{\infty} (\lvert F(\omega)\rvert)^2 df$
Modulation (Frequency Shift) (Section 1.5.2.2.8: Modulation (Frequency Shift))	$f(t) e^{i\phi t}$	$F(\omega - \phi)$

Table 1.3: Table of Fourier Transform Properties

1.6 Sampling theory

1.6.1 Introduction[48]

Contents of Sampling chapter

- Introduction(Current module)
- Proof (Section 1.6.2)

[48]This content is available online at <http://cnx.org/content/m11419/1.29/>.

- Illustrations (Section 1.6.3)
- Matlab Example[49]
- Hold operation[50]
- System view (Section 1.6.4)
- Aliasing applet[51]
- Exercises[52]
- Table of formulas[53]

1.6.1.1 Why sample?

This section introduces sampling. Sampling is the necessary fundament for all digital signal processing and communication. Sampling can be defined as the process of measuring an analog signal at distinct points.

Digital representation of analog signals offers advantages in terms of

- robustness towards noise, meaning we can send more bits/s
- use of flexible processing equipment, in particular the computer
- more reliable processing equipment
- easier to adapt complex algorithms

1.6.1.2 Claude E. Shannon

Figure 1.71: Claude Elwood Shannon (1916-2001)

Claude Shannon[54] has been called the father of information theory, mainly due to his landmark papers on the "Mathematical theory of communication"[55] . Harry Nyquist[56] was the first to state the sampling theorem

[49]"Sampling and reconstruction with Matlab" <http://cnx.org/content/m11549/latest/>
[50]"Hold operation" <http://cnx.org/content/m11458/latest/>
[51]"Aliasing Applet" <http://cnx.org/content/m11448/latest/>
[52]"Exercises" <http://cnx.org/content/m11442/latest/>
[53]"Table of Formulas" <http://cnx.org/content/m11450/latest/>
[54]http://www.research.att.com/~njas/doc/ces5.html
[55]http://cm.bell-labs.com/cm/ms/what/shannonday/shannon1948.pdf
[56]http://www.wikipedia.org/wiki/Harry_Nyquist

in 1928, but it was not proven until Shannon proved it 21 years later in the paper "Communications in the presence of noise"[57] .

1.6.1.3 Notation

In this chapter we will be using the following notation

- Original analog signal $x(t)$
- Sampling frequency F_s
- Sampling interval T_s (Note that: $F_s = \frac{1}{T_s}$)
- Sampled signal $x_s(n)$. (Note that $x_s(n) = x(nT_s)$)
- Real angular frequency Ω
- Digital angular frequency ω. (Note that: $\omega = \Omega T_s$)

1.6.1.4 The Sampling Theorem

NOTE: When sampling an analog signal the sampling frequency must be greater than twice the highest frequency component of the analog signal to be able to reconstruct the original signal from the sampled version.

1.6.1.5

Finished? Have at look at: Proof (Section 1.6.2); Illustrations (Section 1.6.3); Matlab Example[58]; Aliasing applet[59]; Hold operation[60]; System view (Section 1.6.4); Exercises[61]

1.6.2 Proof[62]

NOTE: In order to recover the signal $x(t)$ from it's samples exactly, it is necessary to sample $x(t)$ at a rate greater than twice it's highest frequency component.

1.6.2.1 Introduction

As mentioned earlier (p. 86), sampling is the necessary fundament when we want to apply digital signal processing on analog signals.

Here we present the proof of the sampling theorem. The proof is divided in two. First we find an expression for the spectrum of the signal resulting from sampling the original signal $x(t)$. Next we show that the signal $x(t)$ can be recovered from the samples. Often it is easier using the frequency domain when carrying out a proof, and this is also the case here.

Key points in the proof

- We find an equation (1.159) for the spectrum of the sampled signal
- We find a simple method to reconstruct (1.165) the original signal
- The sampled signal has a periodic spectrum...
- ...and the period is $2 \times \pi F_s$

[57] http://www.stanford.edu/class/ee104/shannonpaper.pdf
[58] "Sampling and reconstruction with Matlab" <http://cnx.org/content/m11549/latest/>
[59] "Aliasing Applet" <http://cnx.org/content/m11448/latest/>
[60] "Hold operation" <http://cnx.org/content/m11458/latest/>
[61] "Exercises" <http://cnx.org/content/m11442/latest/>
[62] This content is available online at <http://cnx.org/content/m11423/1.27/>.

1.6.2.2 Proof part 1 - Spectral considerations

By sampling $x(t)$ every T_s second we obtain $x_s(n)$. The inverse fourier transform of this time discrete signal[63] is

$$x_s(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} X_s\left(e^{i\omega}\right) e^{i\omega n} d\omega \tag{1.152}$$

For convenience we express the equation in terms of the real angular frequency Ω using $\omega = \Omega T_s$. We then obtain

$$x_s(n) = \frac{T_s}{2\pi} \int_{\frac{-\pi}{T_s}}^{\frac{\pi}{T_s}} X_s\left(e^{i\Omega T_s}\right) e^{i\Omega T_s n} d\Omega \tag{1.153}$$

The inverse fourier transform of a continuous signal is

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(i\Omega) e^{i\Omega t} d\Omega \tag{1.154}$$

From this equation we find an expression for $x(nT_s)$

$$x(nT_s) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(i\Omega) e^{i\Omega n T_s} d\Omega \tag{1.155}$$

To account for the difference in region of integration we split the integration in (1.155) into subintervals of length $\frac{2\pi}{T_s}$ and then take the sum over the resulting integrals to obtain the complete area.

$$x(nT_s) = \frac{1}{2\pi} \sum_{k=-\infty}^{\infty} \int_{\frac{(2k-1)\pi}{T_s}}^{\frac{(2k+1)\pi}{T_s}} X(i\Omega) e^{i\Omega n T_s} d\Omega \tag{1.156}$$

Then we change the integration variable, setting $\Omega = \eta + \frac{2\times\pi k}{T_s}$

$$x(nT_s) = \frac{1}{2\pi} \sum_{k=-\infty}^{\infty} \int_{\frac{-\pi}{T_s}}^{\frac{\pi}{T_s}} X\left(i\left(\eta + \frac{2\times\pi k}{T_s}\right)\right) e^{i\left(\eta + \frac{2\times\pi k}{T_s}\right) n T_s} d\eta \tag{1.157}$$

We obtain the final form by observing that $e^{i2\times\pi k n} = 1$, reinserting $\eta = \Omega$ and multiplying by $\frac{T_s}{T_s}$

$$x(nT_s) = \frac{T_s}{2\pi} \int_{\frac{-\pi}{T_s}}^{\frac{\pi}{T_s}} \sum_{k=-\infty}^{\infty} \frac{1}{T_s} X\left(i\left(\Omega + \frac{2\times\pi k}{T_s}\right)\right) e^{i\Omega n T_s} d\Omega \tag{1.158}$$

To make $x_s(n) = x(nT_s)$ for all values of n, the integrands in (1.153) and (1.158) have to agreee, that is

$$X_s\left(e^{i\Omega T_s}\right) = \frac{1}{T_s} \sum_{k=-\infty}^{\infty} X\left(i\left(\Omega + \frac{2\pi k}{T_s}\right)\right) \tag{1.159}$$

This is a central result. We see that the digital spectrum consists of a sum of shifted versions of the original, analog spectrum. Observe the periodicity!

We can also express this relation in terms of the digital angular frequency $\omega = \Omega T_s$

$$X_s\left(e^{i\omega}\right) = \frac{1}{T_s} \sum_{k=-\infty}^{\infty} X\left(i \frac{\omega + 2\times\pi k}{T_s}\right) \tag{1.160}$$

This concludes the first part of the proof. Now we want to find a reconstruction formula, so that we can recover $x(t)$ from $x_s(n)$.

---

[63] "Discrete time signals" <http://cnx.org/content/m11476/latest/>

1.6.2.3 Proof part II - Signal reconstruction

For a bandlimited (Figure 1.73) signal the inverse fourier transform is

$$x(t) = \frac{1}{2\pi} \int_{\frac{-\pi}{T_s}}^{\frac{\pi}{T_s}} X(i\Omega) e^{i\Omega t} d\Omega \tag{1.161}$$

In the interval we are integrating we have: $X_s\left(e^{i\Omega T_s}\right) = \frac{X(i\Omega)}{T_s}$. Substituting this relation into (1.161) we get

$$x(t) = \frac{T_s}{2\pi} \int_{\frac{-\pi}{T_s}}^{\frac{\pi}{T_s}} X_s\left(e^{i\Omega T_s}\right) e^{i\Omega t} d\Omega \tag{1.162}$$

Using the DTFT[64] relation for $X_s\left(e^{i\Omega T_s}\right)$ we have

$$x(t) = \frac{T_s}{2\pi} \int_{\frac{-\pi}{T_s}}^{\frac{\pi}{T_s}} \sum_{n=-\infty}^{\infty} x_s(n) e^{-(i\Omega n T_s)} e^{i\Omega t} d\Omega \tag{1.163}$$

Interchanging integration and summation (under the assumption of convergence) leads to

$$x(t) = \frac{T_s}{2\pi} \sum_{n=-\infty}^{\infty} x_s(n) \int_{\frac{-\pi}{T_s}}^{\frac{\pi}{T_s}} e^{i\Omega(t-nT_s)} d\Omega \tag{1.164}$$

Finally we perform the integration and arrive at the important reconstruction formula

$$x(t) = \sum_{n=-\infty}^{\infty} x_s(n) \frac{\sin\left(\frac{\pi}{T_s}(t-nT_s)\right)}{\frac{\pi}{T_s}(t-nT_s)} \tag{1.165}$$

(Thanks to R.Loos for pointing out an error in the proof.)

[64]"Table of Formulas" <http://cnx.org/content/m11450/latest/>

1.6.2.4 Summary

NOTE: $X_s\left(e^{i\Omega T_s}\right)=\frac{1}{T_s}\sum_{k=-\infty}^{\infty}X\left(i\left(\Omega+\frac{2\pi k}{T_s}\right)\right)$

NOTE: $x(t)=\sum_{n=-\infty}^{\infty}x_s(n)\frac{\sin\left(\frac{\pi}{T_s}(t-nT_s)\right)}{\frac{\pi}{T_s}(t-nT_s)}$

1.6.2.5

Go to Introduction (Section 1.6.1); Illustrations (Section 1.6.3); Matlab Example[65]; Hold operation[66]; Aliasing applet[67]; System view (Section 1.6.4); Exercises[68] ?

1.6.3 Illustrations[69]

In this module we illustrate the processes involved in sampling and reconstruction. To see how all these processes work together as a whole, take a look at the system view (Section 1.6.4). In Sampling and reconstruction with Matlab[70] we provide a Matlab script for download. The matlab script shows the process of sampling and reconstruction **live**.

1.6.3.1 Basic examples

Example 1.20
To sample an analog signal with 3000 Hz as the highest frequency component requires sampling at 6000 Hz or above.

Example 1.21
The sampling theorem can also be applied in two dimensions, i.e. for image analysis. A 2D sampling theorem has a simple physical interpretation in image analysis: Choose the sampling interval such that it is less than or equal to half of the smallest interesting detail in the image.

1.6.3.2 The process of sampling

We start off with an analog signal. This can for example be the sound coming from your stereo at home or your friend talking.

The signal is then sampled uniformly. Uniform sampling implies that we sample every T_s seconds. In Figure 1.72 we see an analog signal. The analog signal has been sampled at times $t=nT_s$.

[65]"Sampling and reconstruction with Matlab" <http://cnx.org/content/m11549/latest/>
[66]"Hold operation" <http://cnx.org/content/m11458/latest/>
[67]"Aliasing Applet" <http://cnx.org/content/m11448/latest/>
[68]"Exercises" <http://cnx.org/content/m11442/latest/>
[69]This content is available online at <http://cnx.org/content/m11443/1.33/>.
[70]"Sampling and reconstruction with Matlab" <http://cnx.org/content/m11549/latest/>

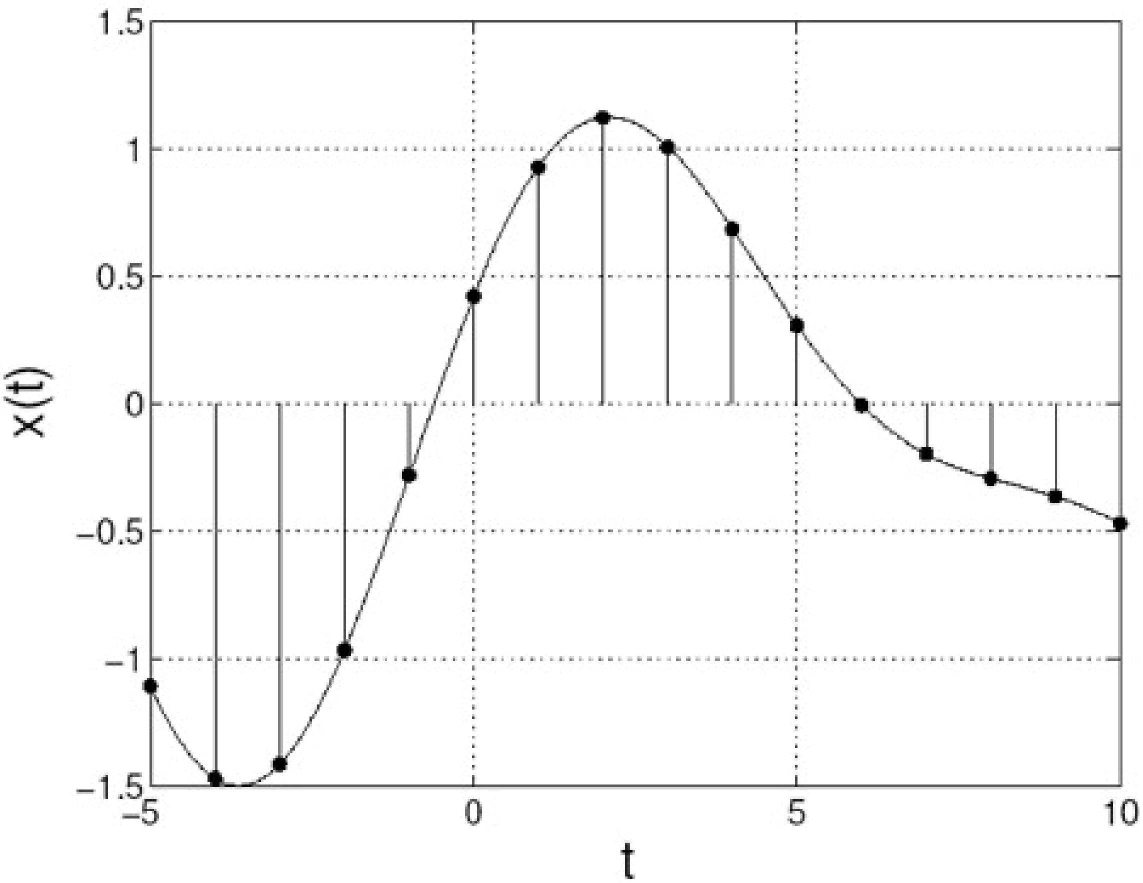


Figure 1.72: Analog signal, samples are marked with dots.

In signal processing it is often more convenient and easier to work in the frequency domain. So let's look at at the signal in frequency domain, Figure 1.73. For illustration purposes we take the frequency content of the signal as a triangle. (If you Fourier transform the signal in Figure 1.72 you will not get such a nice triangle.)

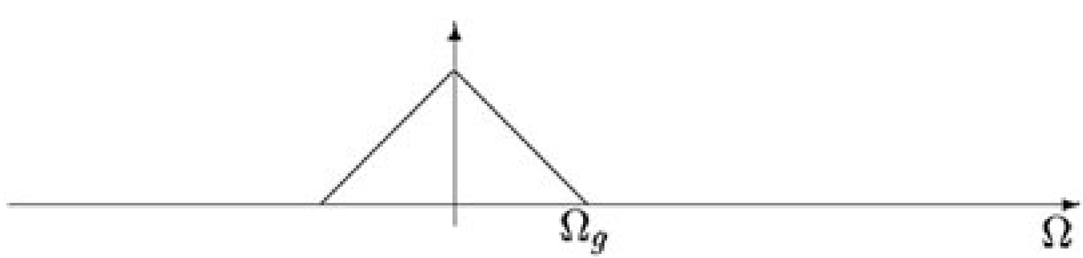


Figure 1.73: The spectrum $X(i\Omega)$.

Notice that the signal in Figure 1.73 is bandlimited. We can see that the signal is bandlimited because

$X(i\Omega)$ is zero outside the interval $[-\Omega_g, \Omega_g]$. Equivalentely we can state that the signal has no angular frequencies above Ω_g, corresponding to no frequencies above $F_g = \frac{\Omega_g}{2\pi}$.

Now let's take a look at the sampled signal in the frequency domain. While proving (Section 1.6.2) the sampling theorem we found the the spectrum of the sampled signal consists of a sum of shifted versions of the analog spectrum. Mathematically this is described by the following equation:

$$X_s\left(e^{i\Omega T_s}\right) = \frac{1}{T_s}\sum_{k=-\infty}^{\infty} X\left(i\left(\Omega + \frac{2\pi k}{T_s}\right)\right) \quad (1.166)$$

1.6.3.2.1 Sampling fast enough

In Figure 1.74 we show the result of sampling $x(t)$ according to the sampling theorem (Section 1.6.1.4: The Sampling Theorem). This means that when sampling the signal in Figure 1.72/Figure 1.73 we use $F_s \geq 2F_g$. Observe in Figure 1.74 that we have the same spectrum as in Figure 1.73 for $\Omega \in [-\Omega_g, \Omega_g]$, except for the scaling factor $\frac{1}{T_s}$. This is a consequence of the sampling frequency. As mentioned in the proof (Key points in the proof, p. 87) the spectrum of the sampled signal is periodic with period $2\pi F_s = \frac{2\pi}{T_s}$.

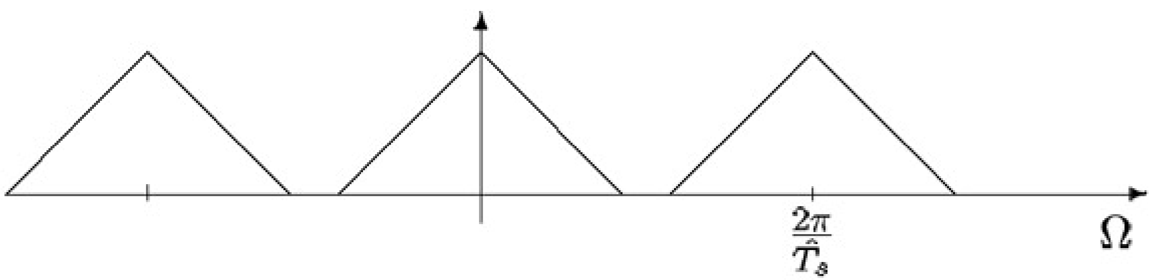


Figure 1.74: The spectrum X_s. Sampling frequency is OK.

So now we are, according to the sample theorem (Section 1.6.1.4: The Sampling Theorem), able to reconstruct the original signal **exactly**. How we can do this will be explored further down under reconstruction (Section 1.6.3.3: Reconstruction). But first we will take a look at what happens when we sample too slowly.

1.6.3.2.2 Sampling too slowly

If we sample $x(t)$ too slowly, that is $F_s < 2F_g$, we will get overlap between the repeated spectra, see Figure 1.75. According to (1.166) the resulting spectra is the sum of these. This overlap gives rise to the concept of aliasing.

> NOTE: If the sampling frequency is less than twice the highest frequency component, then frequencies in the original signal that are above half the sampling rate will be "aliased" and will appear in the resulting signal as lower frequencies.

The consequence of aliasing is that we cannot recover the original signal, so aliasing has to be avoided. Sampling too slowly will produce a sequence $x_s(n)$ that could have orginated from a number of signals. So there is **no** chance of recovering the original signal. To learn more about aliasing, take a look at this module[71]. (Includes an applet for demonstration!)

[71] "Aliasing Applet" <http://cnx.org/content/m11448/latest/>

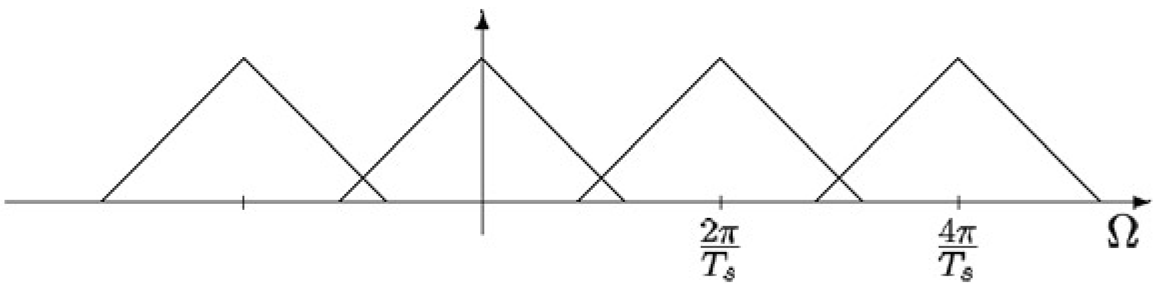


Figure 1.75: The spectrum X_s. Sampling frequency is too low.

To avoid aliasing we have to sample fast enough. But if we can't sample fast enough (possibly due to costs) we can include an Anti-Aliasing filter. This will not able us to get an exact reconstruction but can still be a good solution.

NOTE: Typically a low-pass filter that is applied before sampling to ensure that no components with frequencies greater than half the sample frequency remain.

Example 1.22
The stagecoach effect
In older western movies you can observe aliasing on a stagecoach when it starts to roll. At first the spokes appear to turn forward, but as the stagecoach increase its speed the spokes appear to turn backward. This comes from the fact that the sampling rate, here the number of frames per second, is too low. We can view each frame as a sample of an image that is changing continuously in time. (Applet illustrating the stagecoach effect[72])

1.6.3.3 Reconstruction

Given the signal in Figure 1.74 we want to recover the original signal, but the question is how?

When there is no overlapping in the spectrum, the spectral component given by $k = 0$ (see (1.166)),is equal to the spectrum of the analog signal. This offers an oppurtunity to use a simple reconstruction process. Remember what you have learned about filtering. What we want is to change signal in Figure 1.74 into that of Figure 1.73. To achieve this we have to remove all the extra components generated in the sampling process. To remove the extra components we apply an ideal analog low-pass filter as shown in Figure 1.76 As we see the ideal filter is rectangular in the frequency domain. A rectangle in the frequency domain corresponds to a sinc[73] function in time domain (and vice versa).

[72]http://flowers.ofthenight.org/wagonWheel/wagonWheel.html
[73]http://ccrma-www.stanford.edu/~jos/Interpolation/sinc_function.html

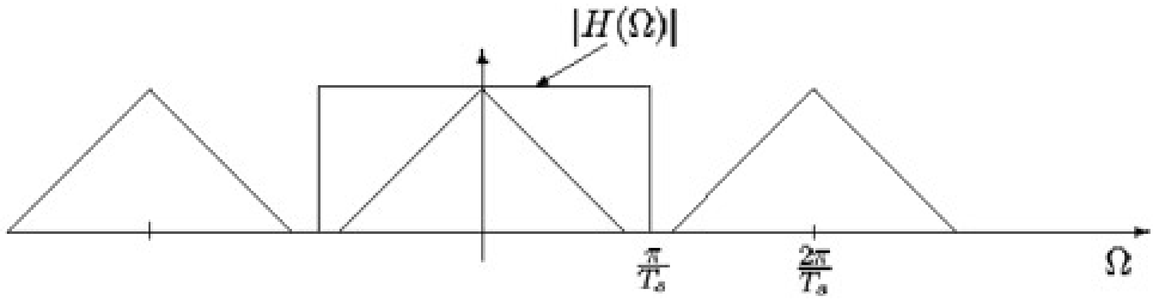


Figure 1.76: $H(i\Omega)$ The ideal reconstruction filter.

Then we have reconstructed the original spectrum, and as we know **if two signals are identical in the frequency domain, they are also identical in the time domain**. End of reconstruction.

1.6.3.4 Conclusions

The Shannon sampling theorem requires that the input signal prior to sampling is band-limited to at most half the sampling frequency. Under this condition the samples give an exact signal representation. It is truly remarkable that such a broad and useful class signals can be represented that easily!

We also looked into the problem of reconstructing the signals form its samples. Again the simplicity of the **principle** is striking: linear filtering by an ideal low-pass filter will do the job. However, the ideal filter is impossible to create, but that is another story...

1.6.3.5

Go to? Introduction (Section 1.6.1); Proof (Section 1.6.2); Illustrations (Section 1.6.3); Matlab Example[74]; Aliasing applet[75]; Hold operation[76]; System view (Section 1.6.4); Exercises[77]

1.6.4 Systems view of sampling and reconstruction[78]

1.6.4.1 Ideal reconstruction system

Figure 1.77 shows the ideal reconstruction system based on the results of the Sampling theorem proof (Section 1.6.2).

Figure 1.77 consists of a sampling device which produces a time-discrete sequence $x_s(n)$. The reconstruction filter, $h(t)$, is an ideal analog sinc[79] filter, with $h(t) = \mathrm{sinc}\left(\frac{t}{T_s}\right)$. We can't apply the time-discrete sequence $x_s(n)$ directly to the analog filter $h(t)$. To solve this problem we turn the sequence into an analog signal using delta functions[80]. Thus we write $x_s(t) = \sum_{n=-\infty}^{\infty} x_s(n)\,\delta(t-nT)$.

[74] "Sampling and reconstruction with Matlab" <http://cnx.org/content/m11549/latest/>
[75] "Aliasing Applet" <http://cnx.org/content/m11448/latest/>
[76] "Hold operation" <http://cnx.org/content/m11458/latest/>
[77] "Exercises" <http://cnx.org/content/m11442/latest/>
[78] This content is available online at <http://cnx.org/content/m11465/1.20/>.
[79] http://ccrma-www.stanford.edu/~jos/Interpolation/sinc_function.html
[80] "Table of Formulas" <http://cnx.org/content/m11450/latest/>

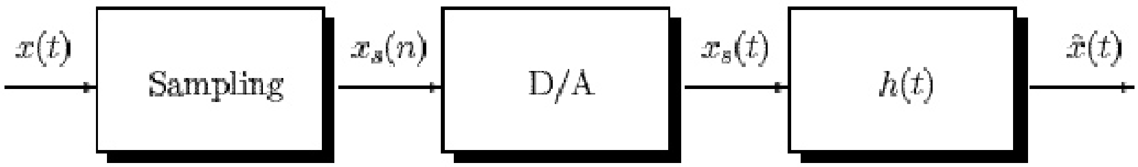


Figure 1.77: Ideal reconstruction system

But when will the system produce an output $\hat{x}(t) = x(t)$? According to the sampling theorem (Section 1.6.1.4: The Sampling Theorem) we have $\hat{x}(t) = x(t)$ when the sampling frequency, F_s, is at least twice the highest frequency component of $x(t)$.

1.6.4.2 Ideal system including anti-aliasing

To be sure that the reconstructed signal is free of aliasing it is customary to apply a lowpass filter, an anti-aliasing filter (p. 93), before sampling as shown in Figure 1.78.

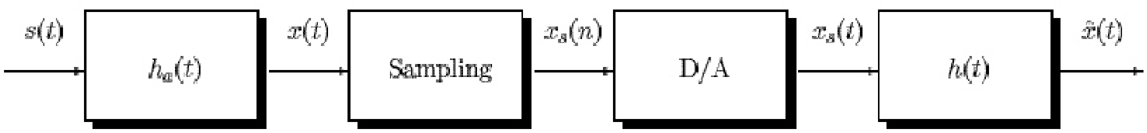


Figure 1.78: Ideal reconstruction system with anti-aliasing filter (p. 93)

Again we ask the question of when the system will produce an output $\hat{x}(t) = s(t)$? If the signal is entirely confined within the passband of the lowpass filter we will get perfect reconstruction if F_s is high enough.

But if the anti-aliasing filter removes the "higher" frequencies, (which in fact is the job of the anti-aliasing filter), we will **never** be able to **exactly** reconstruct the original signal, $s(t)$. If we sample fast enough we can reconstruct $x(t)$, which in most cases is satisfying.

The reconstructed signal, $\hat{x}(t)$, will not have aliased frequencies. This is essential for further use of the signal.

1.6.4.3 Reconstruction with hold operation

To make our reconstruction system realizable there are many things to look into. Among them are the fact that any practical reconstruction system must input finite length pulses into the reconstruction filter. This can be accomplished by the hold operation[81]. To alleviate the distortion caused by the hold opeator we apply the output from the hold device to a compensator. The compensation can be as accurate as we wish, this is cost and application consideration.

[81] "Hold operation" <http://cnx.org/content/m11458/latest/>

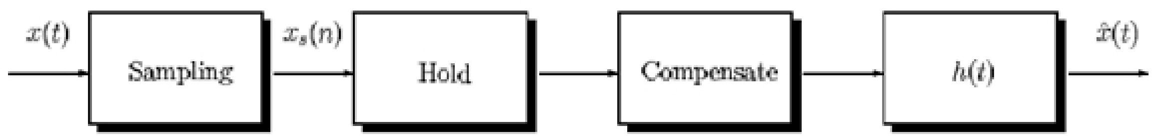


Figure 1.79: More practical reconstruction system with a hold component[82]

By the use of the hold component the reconstruction will not be exact, but as mentioned above we can get as close as we want.

1.6.4.4

Introduction (Section 1.6.1); Proof (Section 1.6.2); Illustrations (Section 1.6.3); Matlab example[83]; Hold operation[84]; Aliasing applet[85]; Exercises[86]

1.6.5 Sampling CT Signals: A Frequency Domain Perspective[87]

1.6.5.1 Understanding Sampling in the Frequency Domain

We want to relate $x_c(t)$ directly to $x[n]$. Compute the CTFT of

$$x_s(t) = \sum_{n=-\infty}^{\infty} x_c(nT)\,\delta(t-nT)$$

$$\begin{array}{rcl} X_s(\Omega) & = & \int_{-\infty}^{\infty} \sum_{n=-\infty}^{\infty} x_c(nT)\,\delta(t-nT)\,e^{(-i)\Omega t}dt \\ & = & \sum_{n=-\infty}^{\infty} x_c(nT) \int_{-\infty}^{\infty} \delta(t-nT)\,e^{(-i)\Omega t}dt \\ & = & \sum_{n=-\infty}^{\infty} x[n]\,e^{(-i)\Omega nT} \\ & = & \sum_{n=-\infty}^{\infty} x[n]\,e^{(-i)\omega n} \\ & = & X(\omega) \end{array} \quad (1.167)$$

where $\omega \equiv \Omega T$ and $X(\omega)$ is the DTFT of $x[n]$.

NOTE:

$$X_s(\Omega) = \frac{1}{T}\sum_{k=-\infty}^{\infty} X_c(\Omega - k\Omega_s)$$

$$\begin{array}{rcl} X(\omega) & = & \frac{1}{T}\sum_{k=-\infty}^{\infty} X_c(\Omega - k\Omega_s) \\ & = & \frac{1}{T}\sum_{k=-\infty}^{\infty} X_c\left(\frac{\omega-2\pi k}{T}\right) \end{array} \quad (1.168)$$

where this last part is 2π-periodic.

[82]"Hold operation" <http://cnx.org/content/m11458/latest/>
[83]"Sampling and reconstruction with Matlab" <http://cnx.org/content/m11549/latest/>
[84]"Hold operation" <http://cnx.org/content/m11458/latest/>
[85]"Aliasing Applet" <http://cnx.org/content/m11448/latest/>
[86]"Exercises" <http://cnx.org/content/m11442/latest/>
[87]This content is available online at <http://cnx.org/content/m10994/2.2/>.

1.6.5.1.1 Sampling

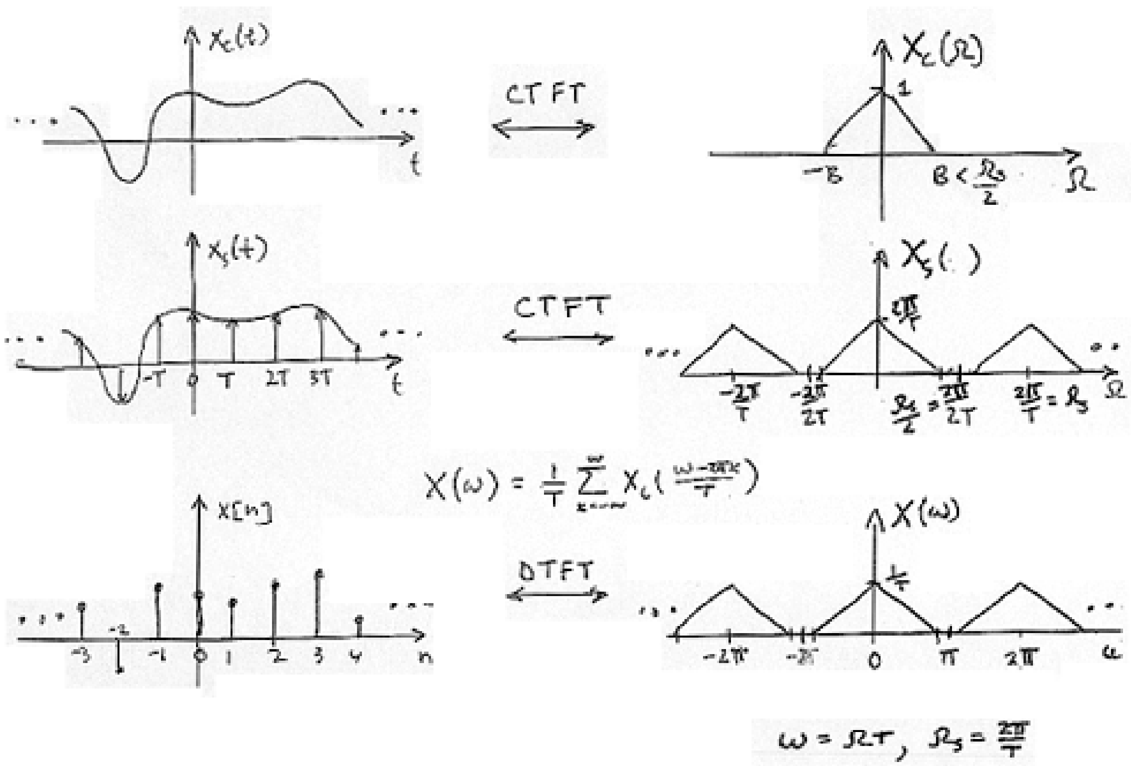


Figure 1.80

Example 1.23: Speech
Speech is intelligible if bandlimited by a CT lowpass filter to the band ±4 kHz. We can sample speech as slowly as _ _ _ _ _?

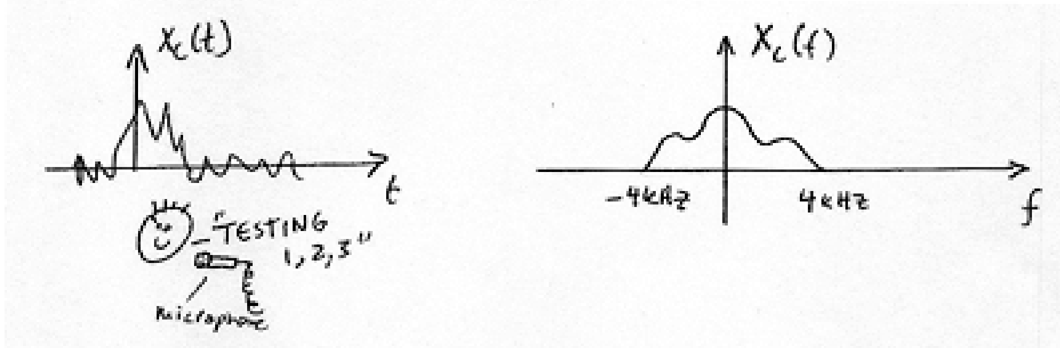


Figure 1.81

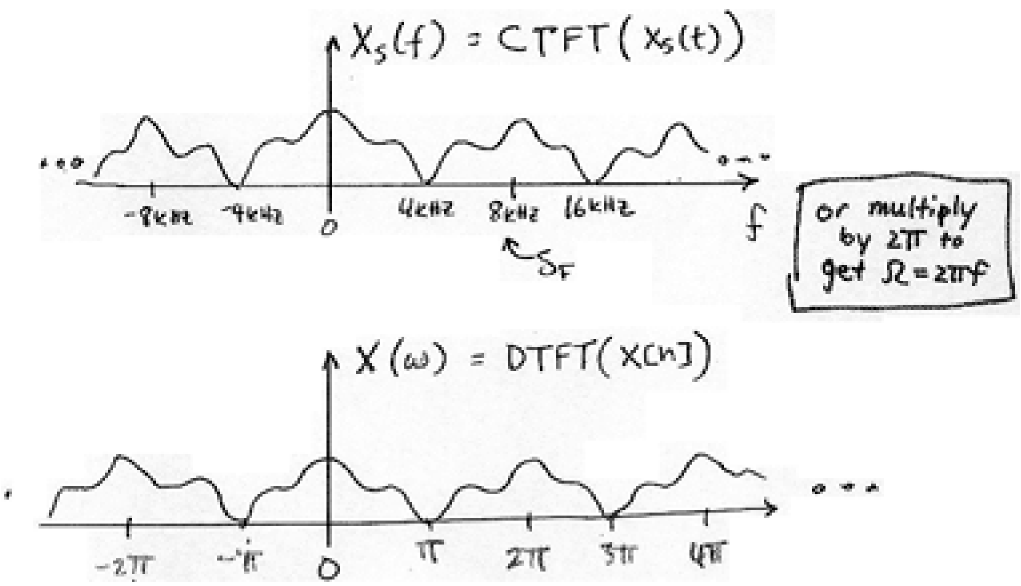


Figure 1.82: Note that there is no mention of T or Ω_s!

1.6.5.2 Relating x[n] to sampled x(t)

Recall the following equality:

$$x_s(t) = \sum_{nn} x(nT)\,\delta(t - nT)$$

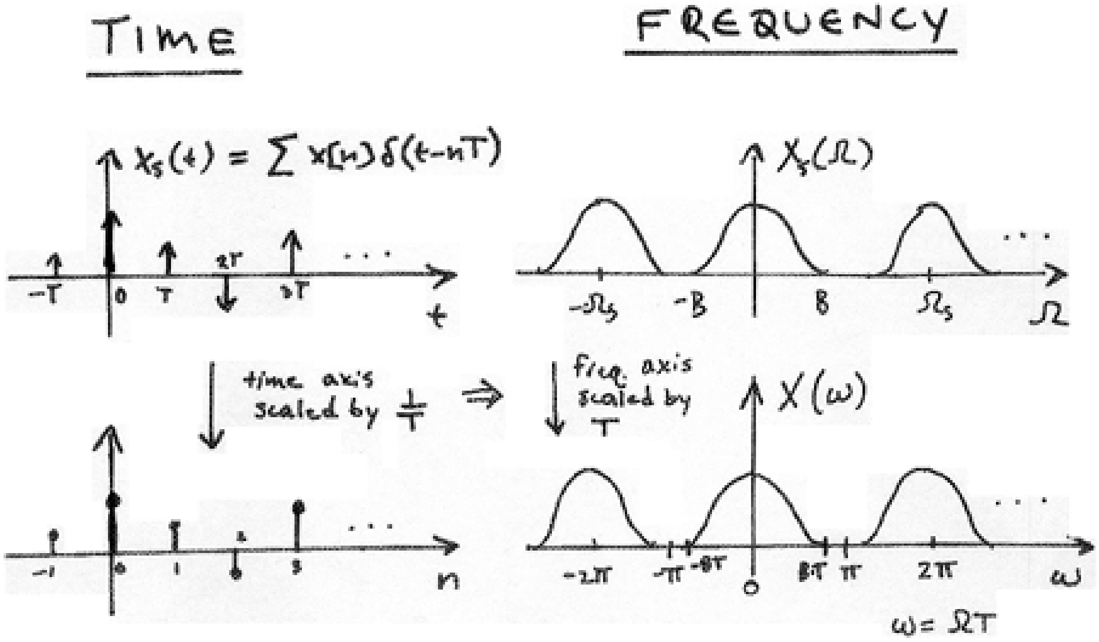


Figure 1.83

Recall the CTFT relation:

$$x(\alpha t) \leftrightarrow \frac{1}{\alpha} X\left(\frac{\Omega}{\alpha}\right) \tag{1.169}$$

where α is a scaling of time and $\frac{1}{\alpha}$ is a scaling in frequency.

$$X_s(\Omega) \equiv X(\Omega T) \tag{1.170}$$

1.7 Time Domain Analysis of Discrete Time Systems

1.7.1 Discrete-Time Systems in the Time-Domain[88]

A discrete-time signal $s(n)$ is **delayed** by n_0 samples when we write $s(n-n_0)$, with $n_0 > 0$. Choosing n_0 to be negative advances the signal along the integers. As opposed to analog delays[89], discrete-time delays can **only** be integer valued. In the frequency domain, delaying a signal corresponds to a linear phase shift of the signal's discrete-time Fourier transform: $s(n-n_0) \leftrightarrow e^{-(i2\pi f n_0)} S\left(e^{i2\pi f}\right)$.

Linear discrete-time systems have the superposition property.

$$S(a_1 x_1(n) + a_2 x_2(n)) = a_1 S(x_1(n)) + a_2 S(x_2(n)) \tag{1.171}$$

[88]This content is available online at <http://cnx.org/content/m10251/2.25/>.
[89]"Simple Systems": Section Delay <http://cnx.org/content/m0006/latest/#delay>

A discrete-time system is called **shift-invariant** (analogous to time-invariant analog systems[90]) if delaying the input delays the corresponding output. If $S(x(n)) = y(n)$, then a shift-invariant system has the property

$$S(x(n-n_0)) = y(n-n_0) \tag{1.172}$$

We use the term shift-invariant to emphasize that delays can only have integer values in discrete-time, while in analog signals, delays can be arbitrarily valued.

We want to concentrate on systems that are both linear and shift-invariant. It will be these that allow us the full power of frequency-domain analysis and implementations. Because we have no physical constraints in "constructing" such systems, we need only a mathematical specification. In analog systems, the differential equation specifies the input-output relationship in the time-domain. The corresponding discrete-time specification is the **difference equation**.

$$y(n) = a_1 y(n-1) + \dots + a_p y(n-p) + b_0 x(n) + b_1 x(n-1) + \dots + b_q x(n-q) \tag{1.173}$$

Here, the output signal $y(n)$ is related to its **past** values $y(n-l)$, $l = \{1, \dots, p\}$, and to the current and past values of the input signal $x(n)$. The system's characteristics are determined by the choices for the number of coefficients p and q and the coefficients' values $\{a_1, \dots, a_p\}$ and $\{b_0, b_1, \dots, b_q\}$.

> ASIDE: There is an asymmetry in the coefficients: where is a_0? This coefficient would multiply the $y(n)$ term in (1.173). We have essentially divided the equation by it, which does not change the input-output relationship. We have thus created the convention that a_0 is always one.

As opposed to differential equations, which only provide an **implicit** description of a system (we must somehow solve the differential equation), difference equations provide an **explicit** way of computing the output for any input. We simply express the difference equation by a program that calculates each output from the previous output values, and the current and previous inputs.

Difference equations are usually expressed in software with **for** loops. A MATLAB program that would compute the first 1000 values of the output has the form

```
for n=1:1000
    y(n) = sum(a.*y(n-1:-1:n-p)) + sum(b.*x(n:-1:n-q));
end
```

An important detail emerges when we consider making this program work; in fact, as written it has (at least) two bugs. What input and output values enter into the computation of $y(1)$? We need values for $y(0)$, $y(-1)$, ..., values we have not yet computed. To compute them, we would need more previous values of the output, which we have not yet computed. To compute these values, we would need even earlier values, ad infinitum. The way out of this predicament is to specify the system's **initial conditions**: we must provide the p output values that occurred before the input started. These values can be arbitrary, but the choice does impact how the system responds to a given input. **One** choice gives rise to a linear system: Make the initial conditions zero. The reason lies in the definition of a linear system[91]: The only way that the output to a sum of signals can be the sum of the individual outputs occurs when the initial conditions in each case are zero.

Exercise 1.7.1.1 *(Solution on p. 141.)*

The initial condition issue resolves making sense of the difference equation for inputs that start at some index. However, the program will not work because of a programming, not conceptual, error. What is it? How can it be "fixed?"

[90]"Simple Systems" <http://cnx.org/content/m0006/latest/#para4wra>
[91]"Simple Systems": Section Linear Systems <http://cnx.org/content/m0006/latest/#linearsys>

Example 1.24
Let's consider the simple system having $p = 1$ and $q = 0$.

$$y(n) = ay(n-1) + bx(n) \tag{1.174}$$

To compute the output at some index, this difference equation says we need to know what the previous output $y(n-1)$ and what the input signal is at that moment of time. In more detail, let's compute this system's output to a unit-sample input: $x(n) = \delta(n)$. Because the input is zero for negative indices, we start by trying to compute the output at $n = 0$.

$$y(0) = ay(-1) + b \tag{1.175}$$

What is the value of $y(-1)$? Because we have used an input that is zero for all negative indices, it is reasonable to assume that the output is also zero. Certainly, the difference equation would not describe a linear system[92] if the input that is zero for **all** time did not produce a zero output. With this assumption, $y(-1) = 0$, leaving $y(0) = b$. For $n > 0$, the input unit-sample is zero, which leaves us with the difference equation $\forall n, n > 0 : (y(n) = ay(n-1))$. We can envision how the filter responds to this input by making a table.

$$y(n) = ay(n-1) + b\delta(n) \tag{1.176}$$

n	$x(n)$	$y(n)$
-1	0	0
0	1	b
1	0	ba
2	0	ba^2
:	0	:
n	0	ba^n

Table 1.4

Coefficient values determine how the output behaves. The parameter b can be any value, and serves as a gain. The effect of the parameter a is more complicated (Table 1.4). If it equals zero, the output simply equals the input times the gain b. For all non-zero values of a, the output lasts forever; such systems are said to be **IIR** (**I**nfinite **I**mpulse **R**esponse). The reason for this terminology is that the unit sample also known as the impulse (especially in analog situations), and the system's response to the "impulse" lasts forever. If a is positive and less than one, the output is a decaying exponential. When $a = 1$, the output is a unit step. If a is negative and greater than -1, the output oscillates while decaying exponentially. When $a = -1$, the output changes sign forever, alternating between b and $-b$. More dramatic effects when $|a| > 1$; whether positive or negative, the output signal becomes larger and larger, **growing** exponentially.

[92]"Simple Systems": Section Linear Systems <http://cnx.org/content/m0006/latest/#linearsys>

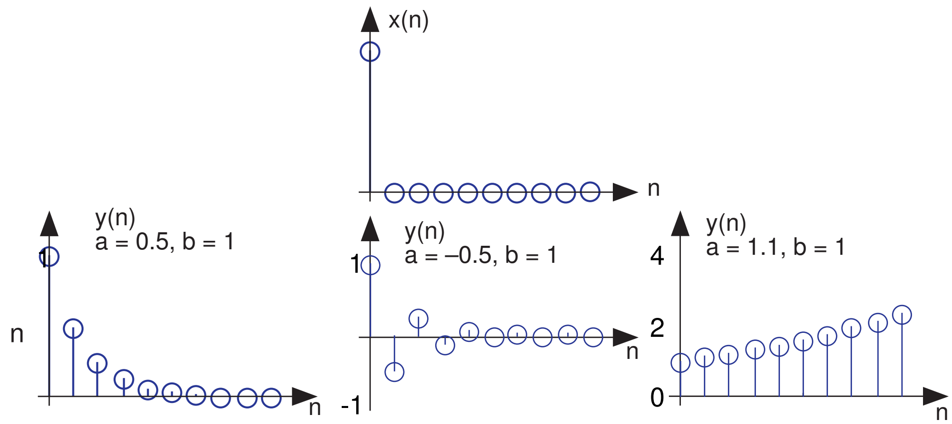


Figure 1.84: The input to the simple example system, a unit sample, is shown at the top, with the outputs for several system parameter values shown below.

Positive values of a are used in population models to describe how population size increases over time. Here, n might correspond to generation. The difference equation says that the number in the next generation is some multiple of the previous one. If this multiple is less than one, the population becomes extinct; if greater than one, the population flourishes. The same difference equation also describes the effect of compound interest on deposits. Here, n indexes the times at which compounding occurs (daily, monthly, etc.), a equals the compound interest rate plus one, and $b = 1$ (the bank provides no gain). In signal processing applications, we typically require that the output remain bounded for any input. For our example, that means that we restrict $|a| < 1$ and choose values for it and the gain according to the application.

Exercise 1.7.1.2 ***(Solution on p. 141.)***

Note that the difference equation (1.173),

$$y(n) = a_1 y(n-1) + \cdots + a_p y(n-p) + b_0 x(n) + b_1 x(n-1) + \cdots + b_q x(n-q)$$

does not involve terms like $y(n+1)$ or $x(n+1)$ on the equation's right side. Can such terms also be included? Why or why not?

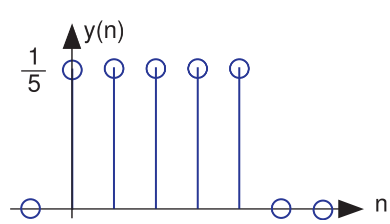


Figure 1.85: The plot shows the unit-sample response of a length-5 boxcar filter.

Example 1.25
A somewhat different system has no "a" coefficients. Consider the difference equation

$$y(n) = \frac{1}{q}(x(n) + \dots + x(n-q+1)) \quad (1.177)$$

Because this system's output depends only on current and previous input values, we need not be concerned with initial conditions. When the input is a unit-sample, the output equals $\frac{1}{q}$ for $n = \{0, \dots, q-1\}$, then equals zero thereafter. Such systems are said to be **FIR** (**F**inite **I**mpulse **R**esponse) because their unit sample responses have finite duration. Plotting this response (Figure 1.85) shows that the unit-sample response is a pulse of width q and height $\frac{1}{q}$. This waveform is also known as a boxcar, hence the name **boxcar filter** given to this system. We'll derive its frequency response and develop its filtering interpretation in the next section. For now, note that the difference equation says that each output value equals the **average** of the input's current and previous values. Thus, the output equals the running average of input's previous q values. Such a system could be used to produce the average weekly temperature ($q = 7$) that could be updated daily.

[Media Object][93]

1.7.2 Discrete Time Convolution[94]

1.7.2.1 Introduction

Convolution, one of the most important concepts in electrical engineering, can be used to determine the output a system produces for a given input signal. It can be shown that a linear time invariant system is completely characterized by its impulse response. The sifting property of the discrete time impulse function tells us that the input signal to a system can be represented as a sum of scaled and shifted unit impulses. Thus, by linearity, it would seem reasonable to compute of the output signal as the sum of scaled and shifted unit impulse responses. That is exactly what the operation of convolution accomplishes. Hence, convolution can be used to determine a linear time invariant system's output from knowledge of the input and the impulse response.

1.7.2.2 Convolution and Circular Convolution

1.7.2.2.1 Convolution

1.7.2.2.1.1 Operation Definition

Discrete time convolution is an operation on two discrete time signals defined by the integral

$$(f * g)[n] = \sum_{k=-\infty}^{\infty} f[k]\, g[n-k] \quad (1.178)$$

for all signals f, g defined on $\mathbb{Z}$. It is important to note that the operation of convolution is commutative, meaning that

$$f * g = g * f \quad (1.179)$$

[93]This media object is a LabVIEW VI. Please view or download it at
<DiscreteTimeSys.llb>
[94]This content is available online at <http://cnx.org/content/m10087/2.30/>.

for all signals f, g defined on $\mathbb{Z}$. Thus, the convolution operation could have been just as easily stated using the equivalent definition

$$(f * g)[n] = \sum_{k=-\infty}^{\infty} f[n-k] g[k] \tag{1.180}$$

for all signals f, g defined on $\mathbb{Z}$. Convolution has several other important properties not listed here but explained and derived in a later module.

1.7.2.2.1.2 Definition Motivation

The above operation definition has been chosen to be particularly useful in the study of linear time invariant systems. In order to see this, consider a linear time invariant system H with unit impulse response h. Given a system input signal x we would like to compute the system output signal $H(x)$. First, we note that the input can be expressed as the convolution

$$x[n] = \sum_{k=-\infty}^{\infty} x[k] \delta[n-k] \tag{1.181}$$

by the sifting property of the unit impulse function. By linearity

$$H(x[n]) = \sum_{k=-\infty}^{\infty} x[k] H(\delta[n-k]). \tag{1.182}$$

Since $H(\delta[n-k])$ is the shifted unit impulse response $h[n-k]$, this gives the result

$$H(x[n]) = \sum_{k=-\infty}^{\infty} x[k] h[n-k] = (x * h)[n]. \tag{1.183}$$

Hence, convolution has been defined such that the output of a linear time invariant system is given by the convolution of the system input with the system unit impulse response.

1.7.2.2.1.3 Graphical Intuition

It is often helpful to be able to visualize the computation of a convolution in terms of graphical processes. Consider the convolution of two functions f, g given by

$$(f * g)[n] = \sum_{k=-\infty}^{\infty} f[k] g[n-k] = \sum_{k=-\infty}^{\infty} f[n-k] g[k]. \tag{1.184}$$

The first step in graphically understanding the operation of convolution is to plot each of the functions. Next, one of the functions must be selected, and its plot reflected across the $k = 0$ axis. For each real n, that same function must be shifted left by n. The point-wise product of the two resulting plots is then computed, and then all of the values are summed.

Example 1.26

Recall that the impulse response for a discrete time echoing feedback system with gain a is

$$h[n] = a^n u[n], \tag{1.185}$$

and consider the response to an input signal that is another exponential

$$x[n] = b^n u[n]. \tag{1.186}$$

We know that the output for this input is given by the convolution of the impulse response with the input signal

$$y[n] = x[n] * h[n]. \tag{1.187}$$

We would like to compute this operation by beginning in a way that minimizes the algebraic complexity of the expression. However, in this case, each possible choice is equally simple. Thus, we would like to compute

$$y[n] = \sum_{k=-\infty}^{\infty} a^k u[k] b^{n-k} u[n-k]. \tag{1.188}$$

The step functions can be used to further simplify this sum. Therefore,

$$y[n] = 0 \tag{1.189}$$

for $n < 0$ and

$$y[n] = \sum_{k=0}^{n} [ab]^k \tag{1.190}$$

for $n \geq 0$. Hence, provided $ab \neq 1$, we have that

$$y[n] = \{ \begin{array}{cc} 0 & n < 0 \\ \frac{1-(ab)^{n+1}}{1-(ab)} & n \geq 0 \end{array}. \tag{1.191}$$

1.7.2.2.2 Circular Convolution

Discrete time circular convolution is an operation on two finite length or periodic discrete time signals defined by the sum

$$(f \circledast g)[n] = \sum_{k=0}^{N-1} \hat{f}[k]\, \hat{g}[n-k] \tag{1.192}$$

for all signals f, g defined on $\mathbb{Z}[0, N-1]$ where $\hat{f}, \hat{g}$ are periodic extensions of f and g. It is important to note that the operation of circular convolution is commutative, meaning that

$$f \circledast g = g \circledast f \tag{1.193}$$

for all signals f, g defined on $\mathbb{Z}[0, N-1]$. Thus, the circular convolution operation could have been just as easily stated using the equivalent definition

$$(f \circledast g)[n] = \sum_{k=0}^{N-1} \hat{f}[n-k]\, \hat{g}[k] \tag{1.194}$$

for all signals f, g defined on $\mathbb{Z}[0, N-1]$ where $\hat{f}, \hat{g}$ are periodic extensions of f and g. Circular convolution has several other important properties not listed here but explained and derived in a later module.

Alternatively, discrete time circular convolution can be expressed as the sum of two summations given by

$$(f \circledast g)[n] = \sum_{k=0}^{n} f[k]\, g[n-k] + \sum_{k=n+1}^{N-1} f[k]\, g[n-k+N] \tag{1.195}$$

for all signals f, g defined on $\mathbb{Z}[0, N-1]$.

Meaningful examples of computing discrete time circular convolutions in the time domain would involve complicated algebraic manipulations dealing with the wrap around behavior, which would ultimately be more confusing than helpful. Thus, none will be provided in this section. Of course, example computations in the time domain are easy to program and demonstrate. However, disrete time circular convolutions are more easily computed using frequency domain tools as will be shown in the discrete time Fourier series section.

1.7.2.2.2.1 Definition Motivation

The above operation definition has been chosen to be particularly useful in the study of linear time invariant systems. In order to see this, consider a linear time invariant system H with unit impulse response h. Given a periodic system input signal x we would like to compute the system output signal $H(x)$. First, we note that the input can be expressed as the circular convolution

$$x[n] = \sum_{k=0}^{N-1} \hat{x}[k]\, \hat{\delta}[n-k] \tag{1.196}$$

by the sifting property of the unit impulse function. By linearity,

$$H(x[n]) = \sum_{k=0}^{N-1} \hat{x}[k]\, H\left(\hat{\delta}[n-k]\right). \tag{1.197}$$

Since $H(\delta[n-k])$ is the shifted unit impulse response $h[n-k]$, this gives the result

$$H(x[n]) = \sum_{k=0}^{N-1} \hat{x}[k]\, \hat{h}[n-k] = (x \circledast h)[n]. \tag{1.198}$$

Hence, circular convolution has been defined such that the output of a linear time invariant system is given by the convolution of the system input with the system unit impulse response.

1.7.2.2.2.2 Graphical Intuition

It is often helpful to be able to visualize the computation of a circular convolution in terms of graphical processes. Consider the circular convolution of two finite length functions f, g given by

$$(f \circledast g)[n] = \sum_{k=0}^{N-1} \hat{f}[k]\, \hat{g}[n-k] = \sum_{k=0}^{N-1} \hat{f}[n-k]\, \hat{g}[k]. \tag{1.199}$$

The first step in graphically understanding the operation of convolution is to plot each of the periodic extensions of the functions. Next, one of the functions must be selected, and its plot reflected across the $k = 0$ axis. For each $n \in \mathbb{Z}[0, N-1]$, that same function must be shifted left by n. The point-wise product of the two resulting plots is then computed, and finally all of these values are summed.

1.7.2.3 Interactive Element

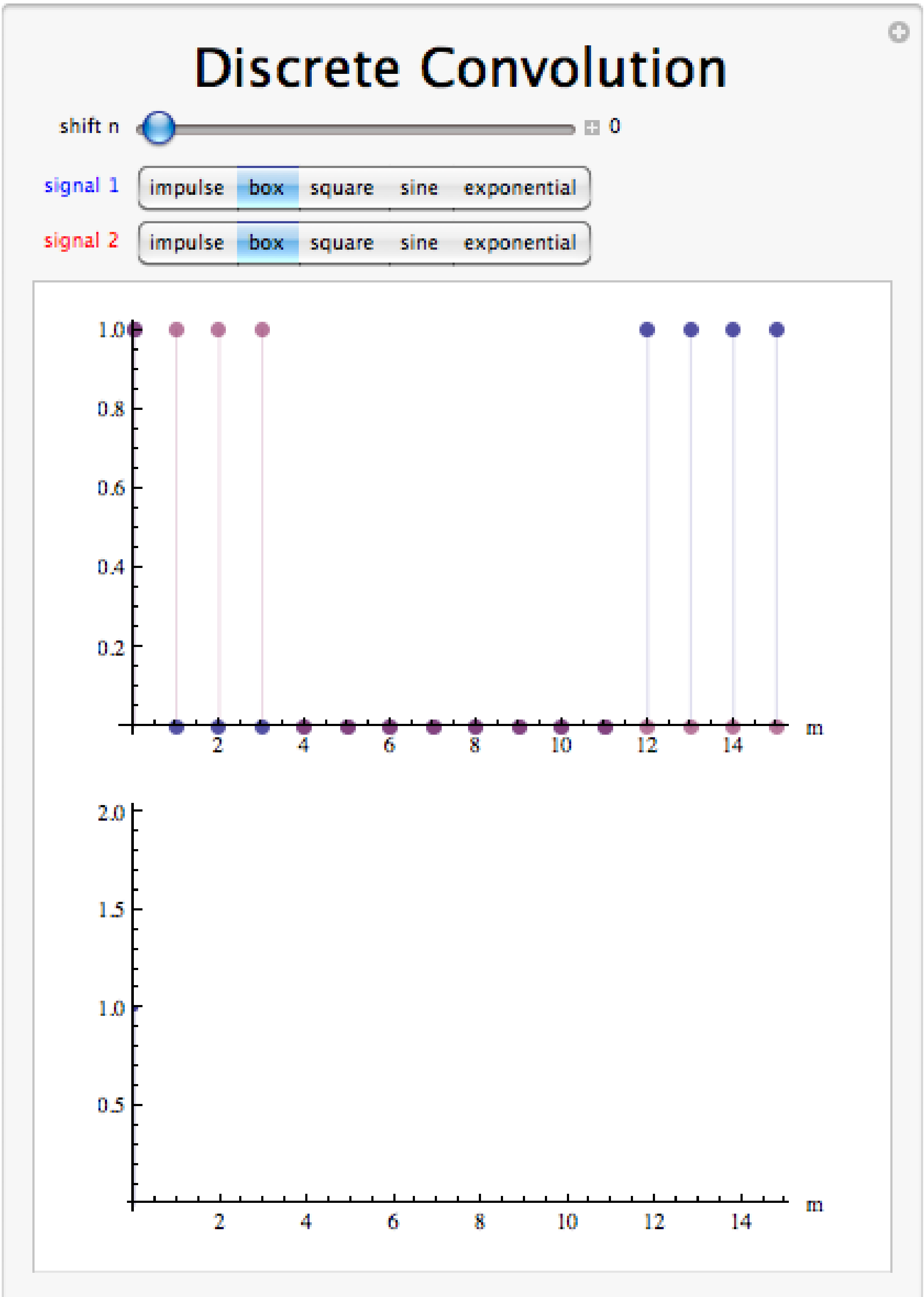


Figure 1.86: Interact (when online) with the Mathematica CDF demonstrating Discrete Linear Convolution. To download, right click and save file as .cdf. Available for free at Connexions <http://cnx.org/content/col10631/1.3>

1.7.2.4 Convolution Summary

Convolution, one of the most important concepts in electrical engineering, can be used to determine the output signal of a linear time invariant system for a given input signal with knowledge of the system's unit impulse response. The operation of discrete time convolution is defined such that it performs this function for infinite length discrete time signals and systems. The operation of discrete time circular convolution is defined such that it performs this function for finite length and periodic discrete time signals. In each case, the output of the system is the convolution or circular convolution of the input signal with the unit impulse response.

1.7.3 Discrete Time Circular Convolution and the DTFS[95]

1.7.3.1 Introduction

This module relates circular convolution of periodic signals in one domain to multiplication in the other domain.

You should be familiar with Discrete-Time Convolution (Section 1.7.2), which tells us that given two discrete-time signals $x[n]$, the system's input, and $h[n]$, the system's response, we define the output of the system as

$$\begin{aligned} y[n] &= x[n] * h[n] \\ &= \sum_{k=-\infty}^{\infty} x[k]\, h[n-k] \end{aligned} \tag{1.200}$$

When we are given two DFTs (finite-length sequences usually of length N), we cannot just multiply them together as we do in the above convolution formula, often referred to as **linear convolution**. Because the DFTs are periodic, they have nonzero values for $n \geq N$ and thus the multiplication of these two DFTs will be nonzero for $n \geq N$. We need to define a new type of convolution operation that will result in our convolved signal being zero outside of the range $n = \{0, 1, \dots, N-1\}$. This idea led to the development of **circular convolution**, also called cyclic or periodic convolution.

1.7.3.2 Signal Circular Convolution

Given a signal $f[n]$ with Fourier coefficients c_k and a signal $g[n]$ with Fourier coefficients d_k, we can define a new signal, $v[n]$, where $v[n] = f[n] \circledast g[n]$ We find that the Fourier Series[96] representation of $v[n]$, a_k, is such that $a_k = c_k d_k$. $f[n] \circledast g[n]$ is the circular convolution (Section 1.7.3) of two periodic signals and is equivalent to the convolution over one interval, *i.e.* $f[n] \circledast g[n] = \sum_{n=0}^{N} \sum_{\eta=0}^{N} f[\eta]\, g[n-\eta]$.

> NOTE: Circular convolution in the time domain is equivalent to multiplication of the Fourier coefficients.

This is proved as follows

$$\begin{aligned} a_k &= \frac{1}{N} \sum_{n=0}^{N} v[n]\, e^{-(j\omega_0 k n)} \\ &= \frac{1}{N^2} \sum_{n=0}^{N} \sum_{\eta=0}^{N} f[\eta]\, g[n-\eta]\, e^{-(\omega j_0 k n)} \\ &= \frac{1}{N} \sum_{\eta=0}^{N} f[\eta] \left(\frac{1}{N} \sum_{n=0}^{N} g[n-\eta]\, e^{-(j\omega_0 k n)} \right) \\ &= \forall \nu, \nu = n-\eta : \left(\frac{1}{N} \sum_{\eta=0}^{N} f[\eta] \left(\frac{1}{N} \sum_{\nu=-\eta}^{N-\eta} g[\nu]\, e^{-(j\omega_0 (\nu+\eta))} \right) \right) \\ &= \frac{1}{N} \sum_{\eta=0}^{N} f[\eta] \left(\frac{1}{N} \sum_{\nu=-\eta}^{N-\eta} g[\nu]\, e^{-(j\omega_0 k \nu)} \right) e^{-(j\omega_0 k \eta)} \\ &= \frac{1}{N} \sum_{\eta=0}^{N} f[\eta]\, d_k e^{-(j\omega_0 k \eta)} \\ &= d_k \left(\frac{1}{N} \sum_{\eta=0}^{N} f[\eta]\, e^{-(j\omega_0 k \eta)} \right) \\ &= c_k d_k \end{aligned} \tag{1.201}$$

[95] This content is available online at <http://cnx.org/content/m10786/2.16/>.
[96] "Fourier Series: Eigenfunction Approach" <http://cnx.org/content/m10496/latest/>

1.7.3.2.1 Circular Convolution Formula

What happens when we multiply two DFT's together, where $Y[k]$ is the DFT of $y[n]$?

$$Y[k] = F[k]H[k] \tag{1.202}$$

when $0 \leq k \leq N-1$

Using the DFT synthesis formula for $y[n]$

$$y[n] = \frac{1}{N}\sum_{k=0}^{N-1} F[k]H[k]e^{j\frac{2\pi}{N}kn} \tag{1.203}$$

And then applying the analysis formula $F[k] = \sum_{m=0}^{N-1} f[m]e^{(-j)\frac{2\pi}{N}kn}$

$$\begin{array}{rcl} y[n] & = & \frac{1}{N}\sum_{k=0}^{N-1}\sum_{m=0}^{N-1} f[m]e^{(-j)\frac{2\pi}{N}kn}H[k]e^{j\frac{2\pi}{N}kn} \\ & = & \sum_{m=0}^{N-1} f[m]\left(\frac{1}{N}\sum_{k=0}^{N-1} H[k]e^{j\frac{2\pi}{N}k(n-m)}\right) \end{array} \tag{1.204}$$

where we can reduce the second summation found in the above equation into $h[((n-m))_N] = \frac{1}{N}\sum_{k=0}^{N-1} H[k]e^{j\frac{2\pi}{N}k(n-m)}$ $y[n] = \sum_{m=0}^{N-1} f[m]h[((n-m))_N]$ which equals circular convolution! When we have $0 \leq n \leq N-1$ in the above, then we get:

$$y[n] \equiv f[n] \circledast h[n] \tag{1.205}$$

NOTE: The notation $\circledast$ represents cyclic convolution "mod N".

1.7.3.2.1.1 Alternative Convolution Formula

Alternative Circular Convolution Algorithm

- Step 1: Calculate the DFT of $f[n]$ which yields $F[k]$ and calculate the DFT of $h[n]$ which yields $H[k]$.
- Step 2: Pointwise multiply $Y[k] = F[k]H[k]$
- Step 3: Inverse DFT $Y[k]$ which yields $y[n]$

Seems like a roundabout way of doing things, **but** it turns out that there are **extremely** fast ways to calculate the DFT of a sequence.

To circularily convolve 2 N-point sequences: $y[n] = \sum_{m=0}^{N-1} f[m]h[((n-m))_N]$ For each $n :$ N multiples, $N-1$ additions

N points implies N^2 multiplications, $N(N-1)$ additions implies $O(N^2)$ complexity.

1.7.3.2.2 Steps for Circular Convolution

We can picture periodic[97] sequences as having discrete points on a circle as the domain

[97] "Continuous Time Periodic Signals" <http://cnx.org/content/m10744/latest/>

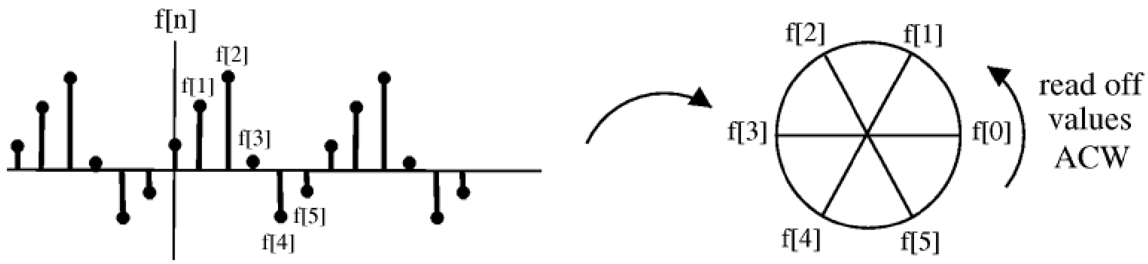


Figure 1.87

Shifting by m, $f(n+m)$, corresponds to rotating the cylinder m notches ACW (counter clockwise). For $m=-2$, we get a shift equal to that in the following illustration:

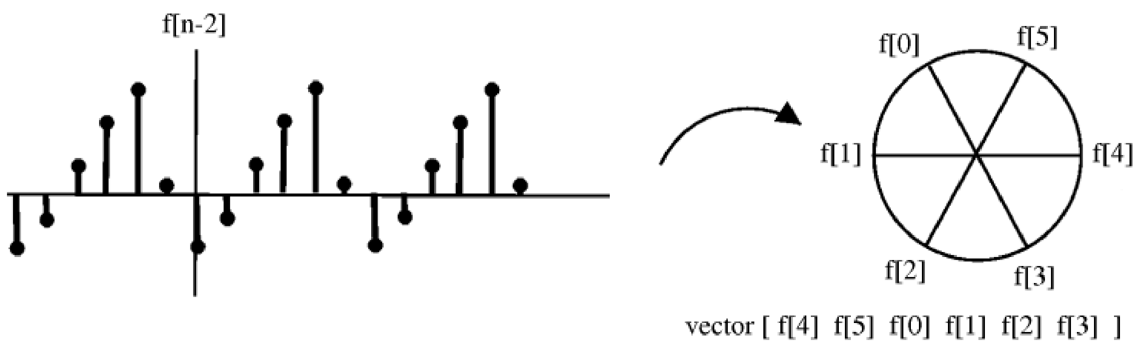


Figure 1.88: for $m=-2$

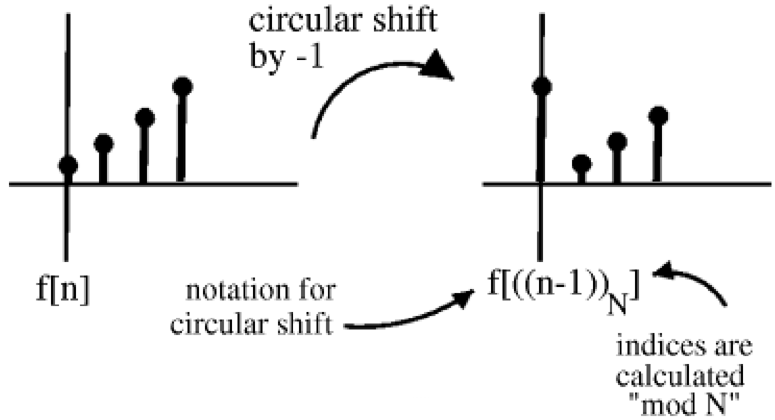


Figure 1.89

To cyclic shift we follow these steps:
1) Write $f(n)$ on a cylinder, ACW

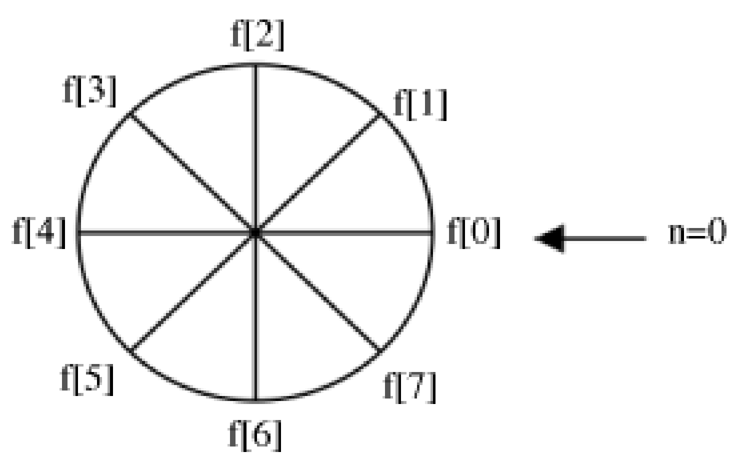


Figure 1.90: $N = 8$

2) To cyclic shift by m, spin cylinder m spots ACW

$$f[n] \to f[((n+m))_N]$$

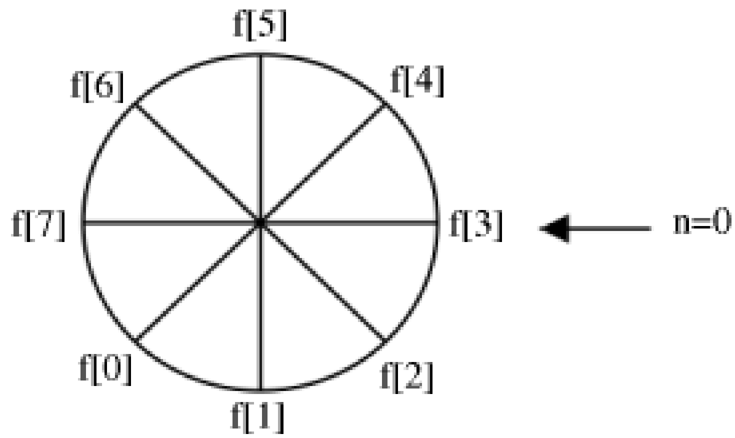


Figure 1.91: $m = -3$

1.7.3.2.2.1 Notes on circular shifting

$f[((n+N))_N] = f[n]$ Spinning N spots is the same as spinning all the way around, or not spinning at all.

$f[((n+N))_N] = f[((n-(N-m)))_N]$ Shifting ACW m is equivalent to shifting CW $N-m$

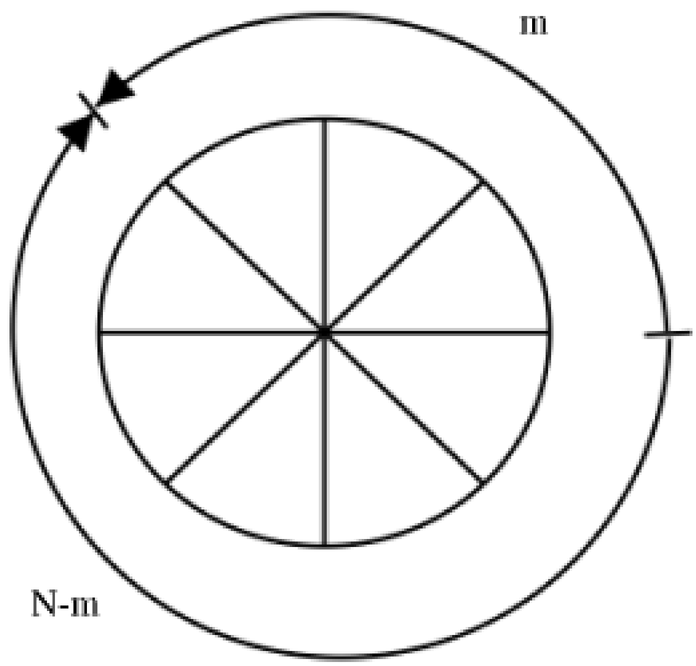


Figure 1.92

$f[((-n))_N]$ The above expression, simply writes the values of $f[n]$ clockwise.

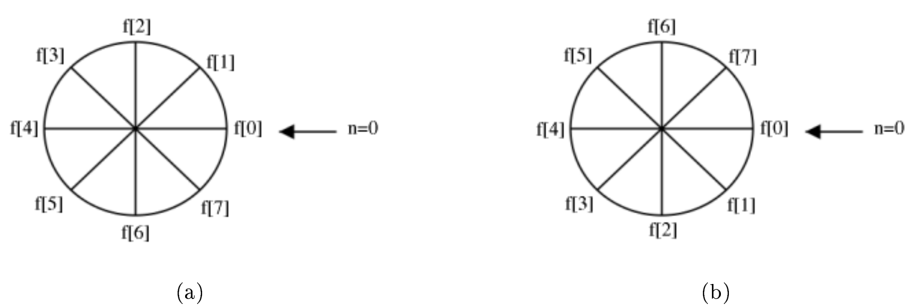


Figure 1.93: (a) $f[n]$ (b) $f[((-n))_N]$

Example 1.27: Convolve (n = 4)

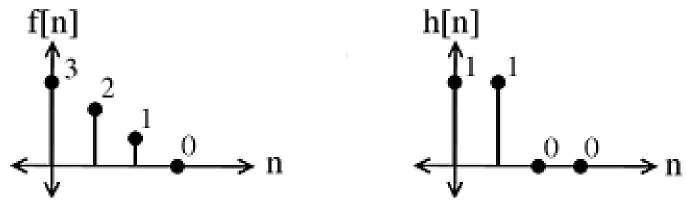


Figure 1.94: Two discrete-time signals to be convolved.

- $h\left[\left(\left(-\left(m\right)\right)\right)_N\right]$

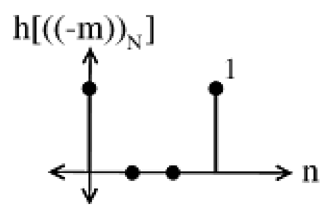


Figure 1.95

Multiply $f[m]$ and sum to yield: $y[0] = 3$

- $h\left[\left(\left(1\left(-\left(m\right)\right)\right)\right)_N\right]$

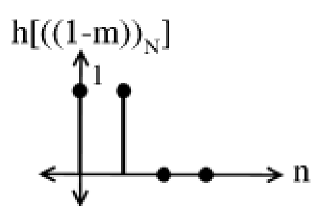


Figure 1.96

Multiply $f[m]$ and sum to yield: $y[1] = 5$

- $h\left[\left(\left(2\left(-\left(m\right)\right)\right)\right)_N\right]$

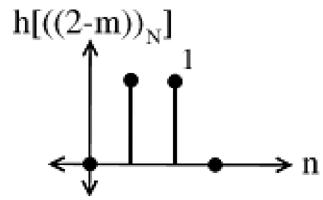


Figure 1.97

Multiply $f[m]$ and sum to yield: $y[2] = 3$

- $h[((3(-(m()()_N]$

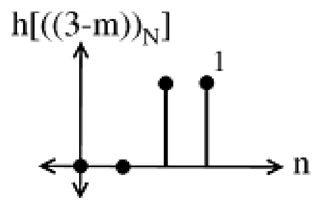


Figure 1.98

Multiply $f[m]$ and sum to yield: $y[3] = 1$

1.7.3.2.3 Exercise

Take a look at a square pulse with a period of T.

For this signal $c_k = \begin{cases} \frac{1}{N} & \text{if } k = 0 \\ \frac{1}{2}\frac{\sin\left(\frac{\pi}{2}k\right)}{\frac{\pi}{2}k} & \text{otherwise} \end{cases}$

Take a look at a triangle pulse train with a period of T.

This signal is created by circularly convolving the square pulse with itself. The Fourier coefficients for this signal are $a_k = {c_k}^2 = \frac{1}{4}\frac{\sin^2}{\left(\frac{\pi}{2}k\right)}$

Exercise 1.7.3.1 ***(Solution on p. 141.)***

Find the Fourier coefficients of the signal that is created when the square pulse and the triangle pulse are convolved.

1.7.3.3 Circular Shifts and the DFT

Theorem 1.2: Circular Shifts and DFT

If $f[n] \overset{\text{DFT}}{\leftrightarrow} F[k]$ then $f[((n-m))_N] \overset{\text{DFT}}{\leftrightarrow} e^{-\left(i\frac{2\pi}{N}km\right)}F[k]$ (*i.e.* circular shift in time domain = phase shift in DFT)

Proof:

$$f[n] = \frac{1}{N} \sum_{k=0}^{N-1} F[k] e^{i \frac{2\pi}{N} kn} \tag{1.206}$$

so phase shifting the DFT

$$\begin{aligned} f[n] &= \frac{1}{N} \sum_{k=0}^{N-1} F[k] e^{-\left(i \frac{2\pi}{N} kn\right)} e^{i \frac{2\pi}{N} kn} \\ &= \frac{1}{N} \sum_{k=0}^{N-1} F[k] e^{i \frac{2\pi}{N} k(n-m)} \\ &= f[((n-m))_N] \end{aligned} \tag{1.207}$$

1.7.3.4 Circular Convolution Demonstration

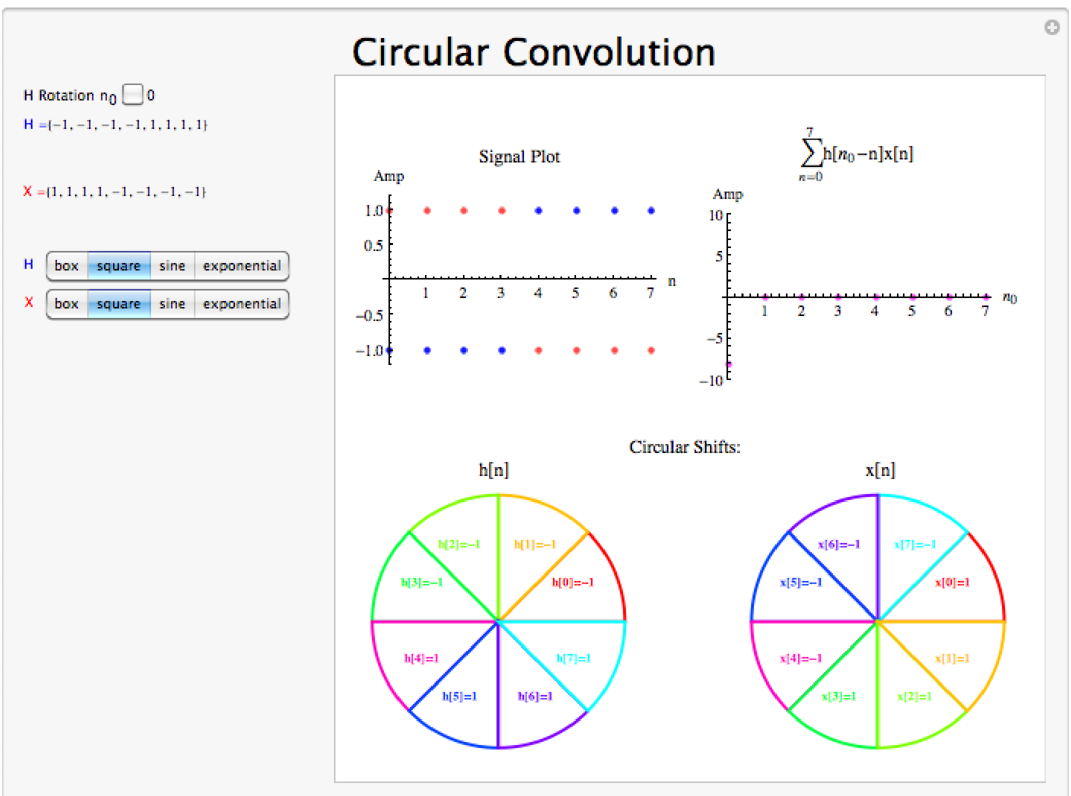


Figure 1.99: Interact (when online) with a Mathematica CDF demonstrating Circular Shifts.

1.7.3.5 Conclusion

Circular convolution in the time domain is equivalent to multiplication of the Fourier coefficients in the frequency domain.

1.8 Discrete Time Fourier Transform (DTFT)

1.8.1 Discrete Time Fourier Transform (DTFT)[98]

1.8.1.1 Introduction

In this module, we will derive an expansion for arbitrary discrete-time functions, and in doing so, derive the **Discrete Time Fourier Transform** (DTFT).

Since complex exponentials (Section 1.1.5) are eigenfunctions of linear time-invariant (LTI) systems[99], calculating the output of an LTI system $\mathcal{H}$ given $e^{i\omega n}$ as an input amounts to simple multiplication, where $\omega_0 = \frac{2\pi k}{N}$, and where $H[k] \in \mathbb{C}$ is the eigenvalue corresponding to k. As shown in the figure, a simple exponential input would yield the output

$$y[n] = H[k]\, e^{i\omega n} \tag{1.208}$$

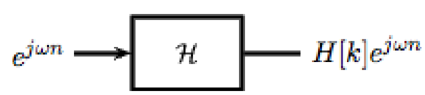


Figure 1.100: Simple LTI system.

Using this and the fact that $\mathcal{H}$ is linear, calculating $y[n]$ for combinations of **complex exponentials** is also straightforward.

$$c_1 e^{i\omega_1 n} + c_2 e^{i\omega_2 n} \rightarrow c_1 H[k_1]\, e^{i\omega_1 n} + c_2 H[k_2]\, e^{i\omega_1 n}$$

$$\sum_l c_l e^{i\omega_l n} \rightarrow \sum_l c_l H[k_l]\, e^{i\omega_l n}$$

The action of H on an input such as those in the two equations above is easy to explain. $\mathcal{H}$ **independently scales** each exponential component $e^{i\omega_l n}$ by a different complex number $H[k_l] \in \mathbb{C}$. As such, if we can write a function $y[n]$ as a combination of complex exponentials it allows us to easily calculate the output of a system.

Now, we will look to use the power of complex exponentials to see how we may represent arbitrary signals in terms of a set of simpler functions by superposition of a number of complex exponentials. Below we will present the **Discrete-Time Fourier Transform** (DTFT). Because the DTFT deals with nonperiodic signals, we must find a way to include all real frequencies in the general equations. For the DTFT we simply utilize summation over all real numbers rather than summation over integers in order to express the aperiodic signals.

[98]This content is available online at <http://cnx.org/content/m10108/2.18/>.
[99]"Eigenfunctions of LTI Systems" <http://cnx.org/content/m10500/latest/>

1.8.1.2 DTFT synthesis

It can be demonstrated that an arbitrary Discrete Time-periodic function $f[n]$ can be written as a linear combination of harmonic complex sinusoids

$$f[n] = \sum_{k=0}^{N-1} c_k e^{i\omega_0 kn} \quad (1.209)$$

where $\omega_0 = \frac{2\pi}{N}$ is the fundamental frequency. For almost all $f[n]$ of practical interest, there exists c_n to make (1.209) true. If $f[n]$ is finite energy ($f[n] \in L^2[0,N]$), then the equality in (1.209) holds in the sense of energy convergence; with discrete-time signals, there are no concerns for divergence as there are with continuous-time signals.

The c_n - called the Fourier coefficients - tell us "how much" of the sinusoid $e^{j\omega_0 kn}$ is in $f[n]$. The formula shows $f[n]$ as a sum of complex exponentials, each of which is easily processed by an LTI system (since it is an eigenfunction of **every** LTI system). Mathematically, it tells us that the set of complex exponentials $\left\{\forall k, k \in \mathbb{Z} : \left(e^{j\omega_0 kn}\right)\right\}$ form a basis for the space of N-periodic discrete time functions.

1.8.1.2.1 Equations

Now, in order to take this useful tool and apply it to arbitrary non-periodic signals, we will have to delve deeper into the use of the superposition principle. Let $s_T(t)$ be a periodic signal having period T. We want to consider what happens to this signal's spectrum as the period goes to infinity. We denote the spectrum for any assumed value of the period by $c_n(T)$. We calculate the spectrum according to the Fourier formula for a periodic signal, known as the Fourier Series (for more on this derivation, see the section on **Fourier Series**.)

$$c_n = \frac{1}{T}\int_0^T s(t)\, exp(-ß\omega_0 t)\ dt \quad (1.210)$$

where $\omega_0 = \frac{2\pi}{T}$ and where we have used a symmetric placement of the integration interval about the origin for subsequent derivational convenience. We vary the frequency index n proportionally as we increase the period. Define

$$S_T(f) \equiv Tc_n = \frac{1}{T}\int_0^T (S_T(f)\, exp(ß\omega_0 t)\ dt(1.211)$$

making the corresponding Fourier Series

$$s_T(t) = \sum_{-\infty}^{\infty} f(t)\, exp(ß\omega_0 t)\ \frac{1}{T} \quad (1.212)$$

As the period increases, the spectral lines become closer together, becoming a continuum. Therefore,

$$\lim_{T\to\infty} s_T(t) \equiv s(t) = \int_{-\infty}^{\infty} S(f)\, exp(ß\omega_0 t)\ df \quad (1.213)$$

with

$$S(f) = \int_{-\infty}^{\infty} s(t)\, exp(-ß\omega_0 t)\ dt \quad (1.214)$$

Discrete-Time Fourier Transform

$$\mathcal{F}(\omega) = \sum_{n=-\infty}^{\infty} f[n]\, e^{-(i\omega n)} \quad (1.215)$$

Inverse DTFT

$$f[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} \mathcal{F}(\omega) e^{i\omega n} d\omega \quad (1.216)$$

WARNING: It is not uncommon to see the above formula written slightly different. One of the most common differences is the way that the exponential is written. The above equations use the radial frequency variable ω in the exponential, where $\omega = 2\pi f$, but it is also common to include the more explicit expression, $i2\pi ft$, in the exponential. Sometimes DTFT notation is expressed as $F(e^{i\omega})$, to make it clear that it is not a CTFT (which is denoted as $F(\Omega)$). Click here[100] for an overview of the notation used in Connexion's DSP modules.

1.8.1.3 DTFT Definition demonstration

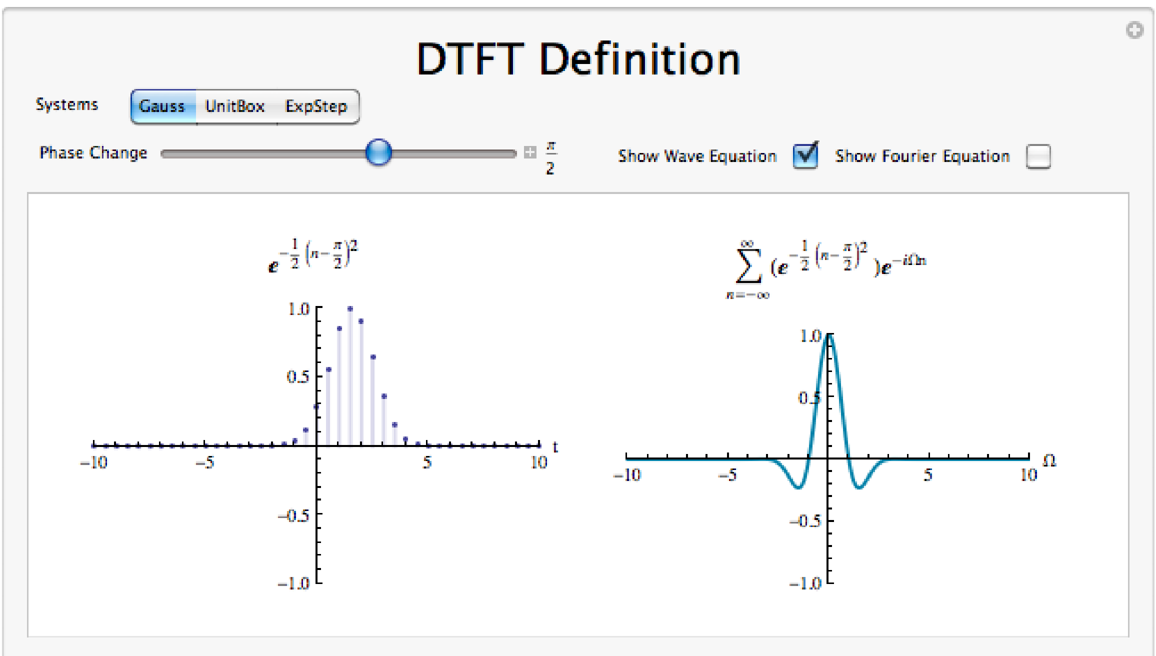


Figure 1.101: Click on the above thumbnail image (when online) to download an interactive Mathematica Player demonstrating Discrete Time Fourier Transform. To Download, right-click and save target as .cdf.

1.8.1.4 DTFT Summary

Because complex exponentials are eigenfunctions of LTI systems, it is often useful to represent signals using a set of complex exponentials as a basis. The discrete time Fourier transform synthesis formula expresses a discrete time, aperiodic function as the infinite sum of continuous frequency complex exponentials.

$$\mathcal{F}(\omega) = \sum_{n=-\infty}^{\infty} f[n] e^{-(i\omega n)} \quad (1.217)$$

[100] "DSP notation" <http://cnx.org/content/m10161/latest/>

The discrete time Fourier transform analysis formula takes the same discrete time domain signal and represents the signal in the continuous frequency domain.

$$f[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} \mathcal{F}(\omega) e^{i\omega n} d\omega \tag{1.218}$$

1.8.2 Properties of the DTFT[101]

1.8.2.1 Introduction

This module will look at some of the basic properties of the Discrete-Time Fourier Transform (Section 1.8.1) (DTFT).

> NOTE: We will be discussing these properties for aperiodic, discrete-time signals but understand that very similar properties hold for continuous-time signals and periodic signals as well.

1.8.2.2 Discussion of Fourier Transform Properties

1.8.2.2.1 Linearity

The combined addition and scalar multiplication properties in the table above demonstrate the basic property of linearity. What you should see is that if one takes the Fourier transform of a linear combination of signals then it will be the same as the linear combination of the Fourier transforms of each of the individual signals. This is crucial when using a table (Section 1.8.7) of transforms to find the transform of a more complicated signal.

Example 1.28

We will begin with the following signal:

$$z[n] = af_1[n] + bf_2[n] \tag{1.219}$$

Now, after we take the Fourier transform, shown in the equation below, notice that the linear combination of the terms is unaffected by the transform.

$$Z(\omega) = aF_1(\omega) + bF_2(\omega) \tag{1.220}$$

1.8.2.2.2 Symmetry

Symmetry is a property that can make life quite easy when solving problems involving Fourier transforms. Basically what this property says is that since a rectangular function in time is a sinc function in frequency, then a sinc function in time will be a rectangular function in frequency. This is a direct result of the similarity between the forward DTFT and the inverse DTFT. The only difference is the scaling by 2π and a frequency reversal.

[101]This content is available online at <http://cnx.org/content/m0506/2.8/>.

1.8.2.2.3 Time Scaling

This property deals with the effect on the frequency-domain representation of a signal if the time variable is altered. The most important concept to understand for the time scaling property is that signals that are narrow in time will be broad in frequency and *vice versa*. The simplest example of this is a delta function, a unit pulse[102] with a **very** small duration, in time that becomes an infinite-length constant function in frequency.

The table above shows this idea for the general transformation from the time-domain to the frequency-domain of a signal. You should be able to easily notice that these equations show the relationship mentioned previously: if the time variable is increased then the frequency range will be decreased.

1.8.2.2.4 Time Shifting

Time shifting shows that a shift in time is equivalent to a linear phase shift in frequency. Since the frequency content depends only on the shape of a signal, which is unchanged in a time shift, then only the phase spectrum will be altered. This property is proven below:

Example 1.29

We will begin by letting $z[n] = f[n-\eta]$. Now let us take the Fourier transform with the previous expression substituted in for $z[n]$.

$$Z(\omega) = \int_{-\infty}^{\infty} f[n-\eta] e^{-(i\omega n)} dn \tag{1.221}$$

Now let us make a simple change of variables, where $\sigma = n - \eta$. Through the calculations below, you can see that only the variable in the exponential are altered thus only changing the phase in the frequency domain.

$$\begin{aligned} Z(\omega) &= \int_{-\infty}^{\infty} f[\sigma] e^{-(i\omega(\sigma+\eta)n)} d\eta \\ &= e^{-(i\omega\eta)} \int_{-\infty}^{\infty} f[\sigma] e^{-(i\omega\sigma)} d\sigma \\ &= e^{-(i\omega\eta)} F(\omega) \end{aligned} \tag{1.222}$$

1.8.2.2.5 Convolution

Convolution is one of the big reasons for converting signals to the frequency domain, since convolution in time becomes multiplication in frequency. This property is also another excellent example of symmetry between time and frequency. It also shows that there may be little to gain by changing to the frequency domain when multiplication in time is involved.

We will introduce the convolution integral here, but if you have not seen this before or need to refresh your memory, then look at the discrete-time convolution (Section 1.7.2) module for a more in depth explanation and derivation.

$$\begin{aligned} y[n] &= (f_1[n], f_2[n]) \\ &= \sum_{\eta=-\infty}^{\infty} f_1[\eta] f_2[n-\eta] \end{aligned} \tag{1.223}$$

[102]"Elemental Signals": Section Pulse <http://cnx.org/content/m0004/latest/#pulsedef>

1.8.2.2.6 Time Differentiation

Since LTI (Section 1.2.1) systems can be represented in terms of differential equations, it is apparent with this property that converting to the frequency domain may allow us to convert these complicated differential equations to simpler equations involving multiplication and addition. This is often looked at in more detail during the study of the Z Transform[103].

1.8.2.2.7 Parseval's Relation

$$\sum_{n=-\infty}^{\infty} \left(\left|f\left[n\right]\right|\right)^2 = \int_{-\pi}^{\pi} \left(\left|F\left(\omega\right)\right|\right)^2 d\omega \tag{1.224}$$

Parseval's relation tells us that the energy of a signal is equal to the energy of its Fourier transform.

Figure 1.102

1.8.2.2.8 Modulation (Frequency Shift)

Modulation is absolutely imperative to communications applications. Being able to shift a signal to a different frequency, allows us to take advantage of different parts of the electromagnetic spectrum is what allows us to transmit television, radio and other applications through the same space without significant interference.

The proof of the frequency shift property is very similar to that of the time shift (Section 1.8.2.2.4: Time Shifting); however, here we would use the inverse Fourier transform in place of the Fourier transform. Since we went through the steps in the previous, time-shift proof, below we will just show the initial and final step to this proof:

$$z\left(t\right) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F\left(\omega - \phi\right) e^{i\omega t} d\omega \tag{1.225}$$

Now we would simply reduce this equation through another change of variables and simplify the terms. Then we will prove the property expressed in the table above:

$$z\left(t\right) = f\left(t\right) e^{i\phi t} \tag{1.226}$$

1.8.2.3 Properties Demonstration

An interactive example demonstration of the properties is included below:

[103] "The Laplace Transform" <http://cnx.org/content/m10110/latest/>

This media object is a LabVIEW VI. Please view or download it at
<CTFTSPlab.llb>

Figure 1.103: Interactive Signal Processing Laboratory Virtual Instrument created using NI's Labview.

1.8.2.4 Summary Table of DTFT Properties

Discrete-Time Fourier Transform Properties

	Sequence Domain	**Frequency Domain**
Linearity	$a_1 s_1(n) + a_2 s_2(n)$	$a_1 S_1\left(e^{i2\pi f}\right) + a_2 S_2\left(e^{i2\pi f}\right)$
Conjugate Symmetry	$s(n)$ real	$S\left(e^{i2\pi f}\right) = \overline{S\left(e^{-(i2\pi f)}\right)}$
Even Symmetry	$s(n) = s(-n)$	$S\left(e^{i2\pi f}\right) = S\left(e^{-(i2\pi f)}\right)$
Odd Symmetry	$s(n) = -s(-n)$	$S\left(e^{i2\pi f}\right) = -S\left(e^{-(i2\pi f)}\right)$
Time Delay	$s(n - n_0)$	$e^{-(i2\pi f n_0)} S\left(e^{i2\pi f}\right)$
Multiplication by n	$ns(n)$	$\frac{1}{-(2i\pi)} \frac{dS\left(e^{i2\pi f}\right)}{df}$
Sum	$\sum_{n=-\infty}^{\infty} s(n)$	$S\left(e^{i2\pi 0}\right)$
Value at Origin	$s(0)$	$\int_{-\frac{1}{2}}^{\frac{1}{2}} S\left(e^{i2\pi f}\right) df$
Parseval's Theorem	$\sum_{n=-\infty}^{\infty} \left(\|s(n)\|\right)^2$	$\int_{-\frac{1}{2}}^{\frac{1}{2}} \left(\|S\left(e^{i2\pi f}\right)\|\right)^2 df$
Complex Modulation	$e^{i2\pi f_0 n} s(n)$	$S\left(e^{i2\pi(f-f_0)}\right)$
Amplitude Modulation	$s(n)\cos(2\pi f_0 n)$	$\frac{S\left(e^{i2\pi(f-f_0)}\right)+S\left(e^{i2\pi(f+f_0)}\right)}{2}$
	$s(n)\sin(2\pi f_0 n)$	$\frac{S\left(e^{i2\pi(f-f_0)}\right)-S\left(e^{i2\pi(f+f_0)}\right)}{2i}$

Table 1.5: Discrete-time Fourier transform properties and relations.

1.8.3 Discrete Time Fourier Transform Pair[104]

When we obtain the discrete-time signal via sampling an analog signal, the Nyquist frequency corresponds to the discrete-time frequency $\frac{1}{2}$. To show this, note that a sinusoid at the Nyquist frequency $\frac{1}{2T_s}$ has a sampled waveform that equals

Sinusoid at Nyquist Frequency 1/2T

$$\begin{aligned} \cos\left(2\pi \times \frac{1}{2T_s} nT_s\right) &= \cos(\pi n) \\ &= (-1)^n \end{aligned} \quad (1.227)$$

The exponential in the DTFT at frequency $\frac{1}{2}$ equals $e^{\frac{-(i2\pi n)}{2}} = e^{-(i\pi n)} = (-1)^n$, meaning that the correspondence between analog and discrete-time frequency is established:

Analog, Discrete-Time Frequency Relationship

$$f_D = f_A T_s \quad (1.228)$$

[104]This content is available online at <http://cnx.org/content/m0525/2.7/>.

where f_D and f_A represent discrete-time and analog frequency variables, respectively. The aliasing figure[105] provides another way of deriving this result. As the duration of each pulse in the periodic sampling signal $p_{T_s}(t)$ narrows, the amplitudes of the signal's spectral repetitions, which are governed by the Fourier series coefficients of $p_{T_s}(t)$, become increasingly equal. [106] Thus, the sampled signal's spectrum becomes periodic with period $\frac{1}{T_s}$. Thus, the Nyquist frequency $\frac{1}{2T_s}$ corresponds to the frequency $\frac{1}{2}$.

The inverse discrete-time Fourier transform is easily derived from the following relationship:

$$\int_{-\frac{1}{2}}^{\frac{1}{2}} e^{-(i2\pi fm)} e^{i\pi fn} df = \begin{cases} 1 \text{ if } m = n \\ 0 \text{ if } m \neq n \end{cases} \quad (1.229)$$

Therefore, we find that

$$\begin{aligned} \int_{-\frac{1}{2}}^{\frac{1}{2}} S\left(e^{i2\pi f}\right) e^{i2\pi fn} df &= \int_{-\frac{1}{2}}^{\frac{1}{2}} \sum_{mm} s(m) e^{-(i2\pi fm)} e^{i2\pi fn} df \\ &= \sum_{mm} s(m) \int_{-\frac{1}{2}}^{\frac{1}{2}} e^{(-(i2\pi f))(m-n)} df \\ &= s(n) \end{aligned} \quad (1.230)$$

The Fourier transform pairs in discrete-time are

Fourier Transform Pairs in Discrete Time

$$S\left(e^{i2\pi f}\right) = \sum_{nn} s(n) e^{-(i2\pi fn)} \quad (1.231)$$

Fourier Transform Pairs in Discrete Time

$$s(n) = \int_{-\frac{1}{2}}^{\frac{1}{2}} S\left(e^{i2\pi f}\right) e^{i2\pi fn} df \quad (1.232)$$

1.8.4 DTFT Examples[107]

Example 1.30

Let's compute the discrete-time Fourier transform of the exponentially decaying sequence $s(n) = a^n u(n)$, where $u(n)$ is the unit-step sequence. Simply plugging the signal's expression into the Fourier transform formula,

Fourier Transform Formula

$$\begin{aligned} S\left(e^{i2\pi f}\right) &= \sum_{n=-\infty}^{\infty} a^n u(n) e^{-(i2\pi fn)} \\ &= \sum_{n=0}^{\infty} \left(ae^{-(i2\pi f)}\right)^n \end{aligned} \quad (1.233)$$

This sum is a special case of the **geometric series**.

Geometric Series

$$\forall \alpha, |\alpha| < 1 : \left(\sum_{n=0}^{\infty} \alpha^n = \frac{1}{1-\alpha}\right) \quad (1.234)$$

[105] "The Sampling Theorem", Figure 2: aliasing <http://cnx.org/content/m0050/latest/#alias>

[106] Examination of the periodic pulse signal reveals that as Δ decreases, the value of c_0 , the largest Fourier coefficient, decreases to zero: $|c_0| = \frac{A\Delta}{T}$. Thus, to maintain a mathematically viable Sampling Theorem, the amplitude A must increase as $\frac{1}{\Delta}$, becoming infinitely large as the pulse duration decreases. Practical systems use a small value of Δ , say $0.1T_s$ and use amplifiers to rescale the signal.

[107] This content is available online at <http://cnx.org/content/m0524/2.11/>.

Thus, as long as $|a| < 1$, we have our Fourier transform.

$$S\left(e^{i2\pi f}\right) = \frac{1}{1 - ae^{-(i2\pi f)}} \tag{1.235}$$

Using Euler's relation, we can express the magnitude and phase of this spectrum.

$$|S\left(e^{i2\pi f}\right)| = \frac{1}{\sqrt{\left(1 - a\cos\left(2\pi f\right)\right)^2 + a^2\sin^2\left(2\pi f\right)}} \tag{1.236}$$

$$\angle\left(S\left(e^{i2\pi f}\right)\right) = -\arctan\left(\frac{a\sin\left(2\pi f\right)}{1 - a\cos\left(2\pi f\right)}\right) \tag{1.237}$$

No matter what value of a we choose, the above formulae clearly demonstrate the periodic nature of the spectra of discrete-time signals. Figure 1.104 shows indeed that the spectrum is a periodic function. We need only consider the spectrum between $-\frac{1}{2}$ and $\frac{1}{2}$ to unambiguously define it. When $a > 0$, we have a lowpass spectrum — the spectrum diminishes as frequency increases from 0 to $\frac{1}{2}$ — with increasing a leading to a greater low frequency content; for $a < 0$, we have a highpass spectrum (Figure 1.105).

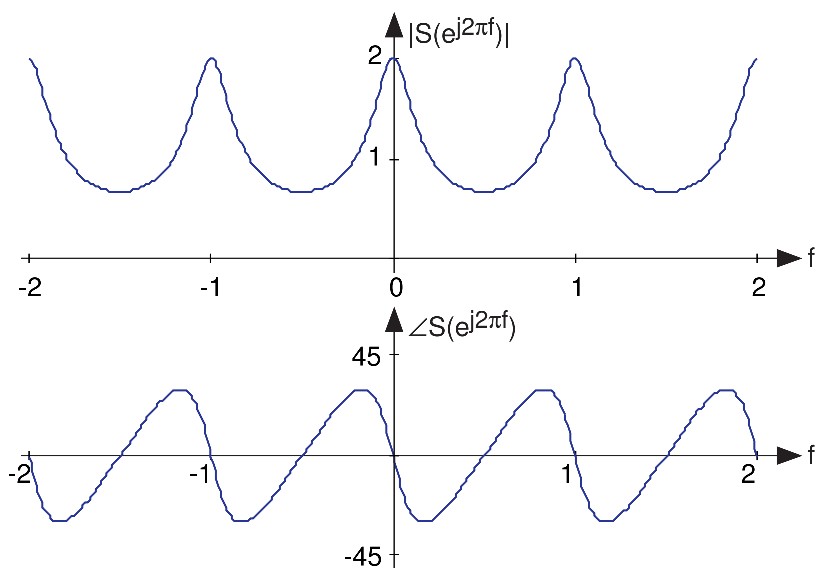


Figure 1.104: The spectrum of the exponential signal ($a = 0.5$) is shown over the frequency range $[-2, 2]$, clearly demonstrating the periodicity of all discrete-time spectra. The angle has units of degrees.

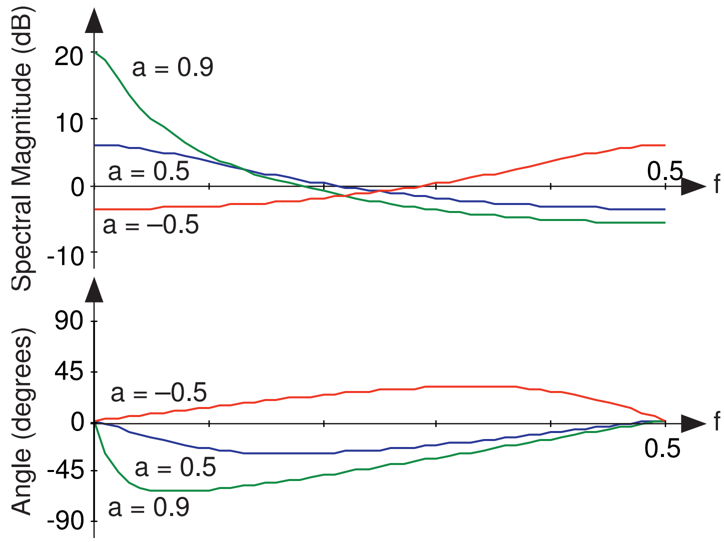


Figure 1.105: The spectra of several exponential signals are shown. What is the apparent relationship between the spectra for $a = 0.5$ and $a = -0.5$?

Example 1.31

Analogous to the analog pulse signal, let's find the spectrum of the length- N pulse sequence.

$$s(n) = \begin{cases} 1 & \text{if } 0 \leq n \leq N-1 \\ 0 & \text{otherwise} \end{cases} \quad (1.238)$$

The Fourier transform of this sequence has the form of a truncated geometric series.

$$S\left(e^{i2\pi f}\right) = \sum_{n=0}^{N-1} e^{-(i2\pi f n)} \quad (1.239)$$

For the so-called finite geometric series, we know that

Finite Geometric Series

$$\sum_{n=n_0}^{N+n_0-1} \alpha^n = \alpha^{n_0} \frac{1-\alpha^N}{1-\alpha} \quad (1.240)$$

for **all** values of α .

Exercise 1.8.4.1 ***(Solution on p. 141.)***

Derive this formula for the finite geometric series sum. The "trick" is to consider the difference between the series'; sum and the sum of the series multiplied by α .

Applying this result yields (Figure 1.106.)

$$\begin{aligned} S\left(e^{i2\pi f}\right) &= \frac{1-e^{-(i2\pi f N)}}{1-e^{-(i2\pi f)}} \\ &= e^{(-(i\pi f))(N-1)} \frac{\sin(\pi f N)}{\sin(\pi f)} \end{aligned} \quad (1.241)$$

The ratio of sine functions has the generic form of $\frac{\sin(Nx)}{\sin(x)}$, which is known as the **discrete-time sinc function**, $\mathrm{dsinc}(x)$. Thus, our transform can be concisely expressed as $S\left(e^{i2\pi f}\right) = e^{(-(i\pi f))(N-1)}\mathrm{dsinc}(\pi f)$. The discrete-time pulse's spectrum contains many ripples, the number of which increase with N , the pulse's duration.

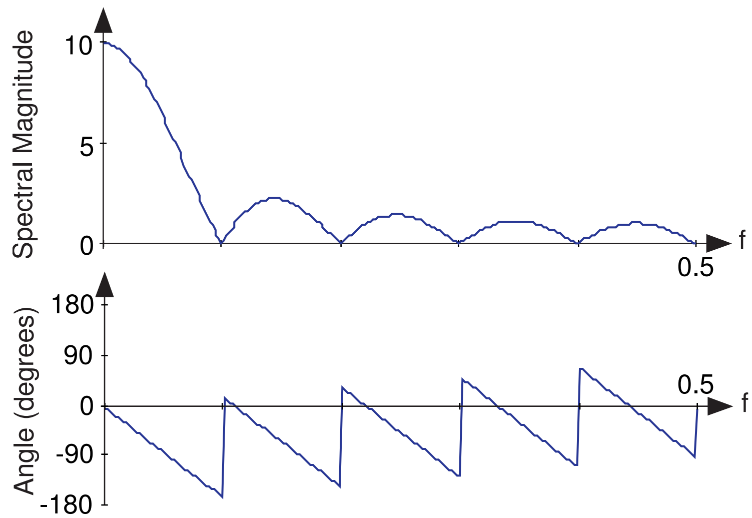


Figure 1.106: The spectrum of a length-ten pulse is shown. Can you explain the rather complicated appearance of the phase?

1.8.5 Discrete Fourier Transformation[108]

1.8.5.1 N-point Discrete Fourier Transform (DFT)

$$X[k] = \sum_{n=0}^{N-1} x[n]\, e^{(-i)\frac{2\pi}{n}kn} \forall k, k = \{0, \dots, N-1\} : (k = \{0, \dots, N-1\}) \tag{1.242}$$

$$x[n] = \frac{1}{N}\sum_{k=0}^{N-1} X[k]\, e^{i\frac{2\pi}{n}kn} \forall n, n = \{0, \dots, N-1\} : (n = \{0, \dots, N-1\}) \tag{1.243}$$

Note that:

- $X[k]$ is the DTFT evaluated at $\omega = \frac{2\pi}{N}k \forall k, k = \{0, \dots, N-1\} : (k = \{0, \dots, N-1\})$
- Zero-padding $x[n]$ to M samples prior to the DFT yields an M-point uniform sampled version of the DTFT:

$$X\left(e^{i\frac{2\pi}{M}k}\right) = \sum_{n=0}^{N-1} x[n]\, e^{(-i)\frac{2\pi}{M}k} \tag{1.244}$$

[108]This content is available online at <http://cnx.org/content/m10421/2.11/>.

$$X\left(e^{i\frac{2\pi}{M}k}\right)=\sum_{n=0}^{N-1}x_{\mathrm{zp}}[n]\,e^{(-i)\frac{2\pi}{M}k}$$

$$X\left(e^{i\frac{2\pi}{M}k}\right)=X_{\mathrm{zp}}[k]\,\forall k,k=\{0,\ldots,M-1\}:(k=\{0,\ldots,M-1\})$$

- The N-pt DFT is sufficient to reconstruct the entire DTFT of an N-pt sequence:

$$X\left(e^{i\omega}\right)=\sum_{n=0}^{N-1}x[n]\,e^{(-i)\omega n} \tag{1.245}$$

$$X\left(e^{i\omega}\right)=\sum_{n=0}^{N-1}\frac{1}{N}\sum_{k=0}^{N-1}X[k]\,e^{i\frac{2\pi}{N}kn}e^{(-i)\omega n}$$

$$X\left(e^{i\omega}\right)=\sum_{k=0}^{N-1}X[k]\,\frac{1}{N}\sum_{k=0}^{N-1}e^{(-i)\left(\omega-\frac{2\pi}{N}k\right)n}$$

$$X\left(e^{i\omega}\right)=\sum_{k=0}^{N-1}X[k]\,\frac{1}{N}\left(\frac{\sin\left(\frac{\omega N-2\pi k}{2}\right)}{\sin\left(\frac{\omega N-2\pi k}{2N}\right)}e^{(-i)\left(\omega-\frac{2\pi}{N}k\right)\frac{N-1}{2}}\right)$$

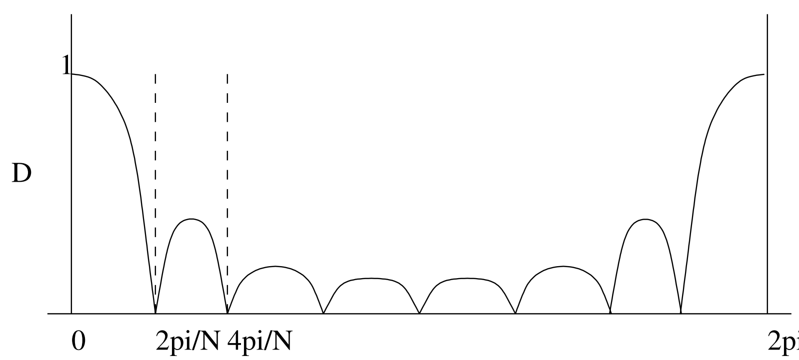


Figure 1.107: Dirichlet sinc, $\frac{1}{N}\frac{\sin\left(\frac{\omega N}{2}\right)}{\sin\left(\frac{\omega}{2}\right)}$

- The DFT has a convenient matrix representation. Defining $W_N=e^{(-i)\frac{2\pi}{N}}$,

$$\begin{pmatrix}X[0]\\X[1]\\\vdots\\X[N-1]\end{pmatrix}=\begin{pmatrix}W_N^0&W_N^0&W_N^0&W_N^0&\ldots\\W_N^0&W_N^1&W_N^2&W_N^3&\ldots\\W_N^0&W_N^2&W_N^4&W_N^6&\ldots\\\vdots&\vdots&\vdots&\vdots&\vdots\end{pmatrix}\begin{pmatrix}x[0]\\x[1]\\\vdots\\x[N-1]\end{pmatrix} \tag{1.246}$$

where $X=W(x)$ respectively. W has the following properties:

- W is Vandermonde: the nth column of W is a polynomial in W_N^n
- W is symmetric: $W = W^T$
- $\frac{1}{\sqrt{N}}W$ is unitary: $\left(\frac{1}{\sqrt{N}}W\right)\left(\frac{1}{\sqrt{N}}W\right)^H = \left(\frac{1}{\sqrt{N}}W\right)^H\left(\frac{1}{\sqrt{N}}W\right) = I$
- $\frac{1}{N}\overline{W} = W^{-1}$, the IDFT matrix.

- For N a power of 2, the FFT can be used to compute the DFT using about $\frac{N}{2}\log_2 N$ rather than N^2 operations.

N	$\frac{N}{2}\log_2 N$	N^2
16	32	256
64	192	4096
256	1024	65536
1024	5120	1048576

Table 1.6

1.8.6 Discrete Fourier Transform (DFT)[109]

The discrete-time Fourier transform (and the continuous-time transform as well) can be evaluated when we have an analytic expression for the signal. Suppose we just have a signal, such as the speech signal used in the previous chapter, for which there is no formula. How then would you compute the spectrum? For example, how did we compute a spectrogram such as the one shown in the speech signal example[110]? The Discrete Fourier Transform (DFT) allows the computation of spectra from discrete-time data. While in discrete-time we can **exactly** calculate spectra, for analog signals no similar exact spectrum computation exists. For analog-signal spectra, use must build special devices, which turn out in most cases to consist of A/D converters and discrete-time computations. Certainly discrete-time spectral analysis is more flexible than continuous-time spectral analysis.

The formula for the DTFT[111] is a sum, which conceptually can be easily computed save for two issues.

- **Signal duration**. The sum extends over the signal's duration, which must be finite to compute the signal's spectrum. It is exceedingly difficult to store an infinite-length signal in any case, so we'll assume that the signal extends over $[0, N-1]$.
- **Continuous frequency**. Subtler than the signal duration issue is the fact that the frequency variable is continuous: It may only need to span one period, like $\left[-\frac{1}{2}, \frac{1}{2}\right]$ or $[0,1]$, but the DTFT formula as it stands requires evaluating the spectra at **all** frequencies within a period. Let's compute the spectrum at a few frequencies; the most obvious ones are the equally spaced ones $f = \frac{k}{K}$, $k \in \{0, \dots, K-1\}$.

We thus define the **discrete Fourier transform** (DFT) to be

$$\forall k, k \in \{0, \dots, K-1\} : \left(S(k) = \sum_{n=0}^{N-1} s(n) e^{-\frac{i2\pi nk}{K}}\right) \tag{1.247}$$

Here, $S(k)$ is shorthand for $S\left(e^{i2\pi\frac{k}{K}}\right)$.

We can compute the spectrum at as many equally spaced frequencies as we like. Note that you can think about this computationally motivated choice as **sampling** the spectrum; more about this interpretation later.

[109] This content is available online at <http://cnx.org/content/m10249/2.28/>.
[110] "Modeling the Speech Signal", Figure 5: spectrogram <http://cnx.org/content/m0049/latest/#spectrogram>
[111] "Discrete-Time Fourier Transform (DTFT)", (1) <http://cnx.org/content/m10247/latest/#eqn1>

The issue now is how many frequencies are enough to capture how the spectrum changes with frequency. One way of answering this question is determining an inverse discrete Fourier transform formula: given $S(k)$, $k = \{0, \dots, K-1\}$ how do we find $s(n)$, $n = \{0, \dots, N-1\}$? Presumably, the formula will be of the form $s(n) = \sum_{k=0}^{K-1} S(k) e^{\frac{i2\pi nk}{K}}$. Substituting the DFT formula in this prototype inverse transform yields

$$s(n) = \sum_{k=0}^{K-1} \sum_{m=0}^{N-1} s(m) e^{-\left(i\frac{2\pi mk}{K}\right)} e^{i\frac{2\pi nk}{K}} \tag{1.248}$$

Note that the orthogonality relation we use so often has a different character now.

$$\sum_{k=0}^{K-1} e^{-\left(i\frac{2\pi km}{K}\right)} e^{i\frac{2\pi kn}{K}} = \begin{cases} K \text{ if } (m = \{n, n \pm K, n \pm 2K, \dots\}) \\ 0 \text{ otherwise} \end{cases} \tag{1.249}$$

We obtain nonzero value whenever the two indices differ by multiples of K. We can express this result as $K \sum_l \delta(m - n - lK)$. Thus, our formula becomes

$$s(n) = \sum_{m=0}^{N-1} s(m) K \sum_{l=-\infty}^{\infty} \delta(m - n - lK) \tag{1.250}$$

The integers n and m both range over $\{0, \dots, N-1\}$. To have an inverse transform, we need the sum to be a **single** unit sample for m, n in this range. If it did not, then $s(n)$ would equal a sum of values, and we would not have a valid transform: Once going into the frequency domain, we could not get back unambiguously! Clearly, the term $l = 0$ always provides a unit sample (we'll take care of the factor of K soon). If we evaluate the spectrum at **fewer** frequencies than the signal's duration, the term corresponding to $m = n + K$ will also appear for some values of m, $n = \{0, \dots, N-1\}$. This situation means that our prototype transform equals $s(n) + s(n+K)$ for some values of n. The only way to eliminate this problem is to require $K \geq N$: We **must** have at least as many frequency samples as the signal's duration. In this way, we can return from the frequency domain we entered via the DFT.

Exercise 1.8.6.1 *(Solution on p. 141.)*

When we have fewer frequency samples than the signal's duration, some discrete-time signal values equal the sum of the original signal values. Given the sampling interpretation of the spectrum, characterize this effect a different way.

Another way to understand this requirement is to use the theory of linear equations. If we write out the expression for the DFT as a set of linear equations,

$$s(0) + s(1) + \dots + s(N-1) = S(0) \tag{1.251}$$

$$s(0) + s(1) e^{(-i)\frac{2\pi}{K}} + \dots + s(N-1) e^{(-i)\frac{2\pi(N-1)}{K}} = S(1)$$

$$\vdots$$

$$s(0) + s(1) e^{(-i)\frac{2\pi(K-1)}{K}} + \dots + s(N-1) e^{(-i)\frac{2\pi(N-1)(K-1)}{K}} = S(K-1)$$

we have K equations in N unknowns if we want to find the signal from its sampled spectrum. This requirement is impossible to fulfill if $K < N$; we must have $K \geq N$. Our orthogonality relation essentially says that if we have a sufficient number of equations (frequency samples), the resulting set of equations can indeed be solved.

By convention, the number of DFT frequency values K is chosen to equal the signal's duration N. The discrete Fourier transform pair consists of

Discrete Fourier Transform Pair

$$S(k) = \sum_{n=0}^{N-1} s(n) e^{-\left(i\frac{2\pi nk}{N}\right)}$$
$$s(n) = \frac{1}{N} \sum_{k=0}^{N-1} S(k) e^{i\frac{2\pi nk}{N}} \quad (1.252)$$

Example 1.32

Use this demonstration to perform DFT analysis of a signal.

This media object is a LabVIEW VI. Please view or download it at <DFTanalysis.llb>

Example 1.33

Use this demonstration to synthesize a signal from a DFT sequence.

This media object is a LabVIEW VI. Please view or download it at <DFT_Component_Manipulation.llb>

1.8.7 Common Fourier Transforms[112]

1.8.7.1 Common CTFT Properties

Time Domain Signal	Frequency Domain Signal	Condition
$e^{-(at)}u(t)$	$\frac{1}{a+i\omega}$	$a>0$
$e^{at}u(-t)$	$\frac{1}{a-i\omega}$	$a>0$
$e^{-(a\|t\|)}$	$\frac{2a}{a^2+\omega^2}$	$a>0$
$te^{-(at)}u(t)$	$\frac{1}{(a+i\omega)^2}$	$a>0$
$t^n e^{-(at)}u(t)$	$\frac{n!}{(a+i\omega)^{n+1}}$	$a>0$
$\delta(t)$	1	
1	$2\pi\delta(\omega)$	
$e^{i\omega_0 t}$	$2\pi\delta(\omega-\omega_0)$	
$\cos(\omega_0 t)$	$\pi(\delta(\omega-\omega_0)+\delta(\omega+\omega_0))$	
$\sin(\omega_0 t)$	$i\pi(\delta(\omega+\omega_0)-\delta(\omega-\omega_0))$	
$u(t)$	$\pi\delta(\omega)+\frac{1}{i\omega}$	
$\mathrm{sgn}(t)$	$\frac{2}{i\omega}$	
$\cos(\omega_0 t)u(t)$	$\frac{\pi}{2}(\delta(\omega-\omega_0)+\delta(\omega+\omega_0)) + \frac{i\omega}{{\omega_0}^2-\omega^2}$	
		continued on next page

[112]This content is available online at <http://cnx.org/content/m10099/2.12/>.

$\sin(\omega_0 t)\, u(t)$	$\frac{\pi}{2i}\left(\delta(\omega-\omega_0)-\delta(\omega+\omega_0)\right) + \frac{\omega_0}{\omega_0{}^2-\omega^2}$	
$e^{-(at)}\sin(\omega_0 t)\, u(t)$	$\frac{\omega_0}{(a+i\omega)^2+\omega_0{}^2}$	$a>0$
$e^{-(at)}\cos(\omega_0 t)\, u(t)$	$\frac{a+i\omega}{(a+i\omega)^2+\omega_0{}^2}$	$a>0$
$u(t+\tau)-u(t-\tau)$	$2\tau\frac{\sin(\omega\tau)}{\omega\tau}=2\tau\mathrm{sinc}(\omega t)$	
$\frac{\omega_0}{\pi}\frac{\sin(\omega_0 t)}{\omega_0 t}=\frac{\omega_0}{\pi}\mathrm{sinc}(\omega_0)$	$u(\omega+\omega_0)-u(\omega-\omega_0)$	
$\left(\frac{t}{\tau}+1\right)\left(u\left(\frac{t}{\tau}+1\right)-u\left(\frac{t}{\tau}\right)\right) + \left(-\frac{t}{\tau}+1\right)\left(u\left(\frac{t}{\tau}\right)-u\left(\frac{t}{\tau}-1\right)\right) = \mathrm{triag}\left(\frac{t}{2\tau}\right)$	$\tau\mathrm{sinc}^2\left(\frac{\omega\tau}{2}\right)$	
$\frac{\omega_0}{2\pi}\mathrm{sinc}^2\left(\frac{\omega_0 t}{2}\right)$	$\left(\frac{\omega}{\omega_0}+1\right)\left(u\left(\frac{\omega}{\omega_0}+1\right)-u\left(\frac{\omega}{\omega_0}\right)\right)+\left(-\frac{\omega}{\omega_0}+1\right)\left(u\left(\frac{\omega}{\omega_0}\right)-u\left(\frac{\omega}{\omega_0}-1\right)\right) = \mathrm{triag}\left(\frac{\omega}{2\omega_0}\right)$	
$\sum_{n=-\infty}^{\infty}\delta(t-nT)$	$\omega_0\sum_{n=-\infty}^{\infty}\delta(\omega-n\omega_0)$	$\omega_0=\frac{2\pi}{T}$
$e^{-\frac{t^2}{2\sigma^2}}$	$\sigma\sqrt{2\pi}e^{-\frac{\sigma^2\omega^2}{2}}$	

Table 1.7

triag[n] is the triangle function for arbitrary real-valued n.

$$\mathrm{triag}[n] = \begin{cases} 1+n & \text{if } -1\le n\le 0 \\ 1-n & \text{if } 0<n\le 1 \\ 0 & \text{otherwise} \end{cases}$$

1.8.8 Linear Constant Coefficient Difference Equations[113]

1.8.8.1 Introduction: Difference Equations

In our study of signals and systems, it will often be useful to describe systems using equations involving the rate of change in some quantity. In discrete time, this is modeled through difference equations, which are a specific type of recurrance relation. For instance, recall that the funds in an account with discretely componded interest rate r will increase by r times the previous balance. Thus, a discretely compounded interest system is described by the first order difference equation shown in (1.253).

$$y(n)=(1+r)\,y(n-1) \tag{1.253}$$

Given a sufficiently descriptive set of initial conditions or boundary conditions, if there is a solution to the difference equation, that solution is unique and describes the behavior of the system. Of course, the results are only accurate to the degree that the model mirrors reality.

1.8.8.2 Linear Constant Coefficient Difference Equations

An important subclass of difference equations is the set of linear constant coefficient difference equations. These equations are of the form

$$Cy(n)=f(n) \tag{1.254}$$

[113]This content is available online at <http://cnx.org/content/m12325/1.5/>.

where C is a difference operator of the form given

$$C = c_N D^N + c_{N-1} D^{N-1} + ... + c_1 D + c_0 \tag{1.255}$$

in which D is the first difference operator

$$D(y(n)) = y(n) - y(n-1). \tag{1.256}$$

Note that operators of this type satisfy the linearity conditions, and $c_0, ..., c_n$ are real constants.

However, (1.254) can easily be written as a linear constant coefficient recurrence equation without difference operators. Conversely, linear constant coefficient recurrence equations can also be written in the form of a difference equation, so the two types of equations are different representations of the same relationship. Although we will still call them linear constant coefficient difference equations in this course, we typically will not write them using difference operators. Instead, we will write them in the simpler recurrence relation form

$$\sum_{k=0}^{N} a_k y(n-k) = \sum_{k=0}^{M} b_k x(n-k) \tag{1.257}$$

where x is the input to the system and y is the output. This can be rearranged to find $y(n)$ as

$$y(n) = \frac{1}{a_0} \left(-\sum_{k=1}^{N} a_k y(n-k) + \sum_{k=0}^{M} b_k x(n-k) \right) \tag{1.258}$$

The forms provided by (1.257) and (1.258) will be used in the remainder of this course.

A similar concept for continuous time setting, differential equations, is discussed in the chapter on time domain analysis of continuous time systems. There are many parallels between the discussion of linear constant coefficient ordinary differential equations and linear constant coefficient differece equations.

1.8.8.3 Applications of Difference Equations

Difference equations can be used to describe many useful digital filters as described in the chapter discussing the z-transform. An additional illustrative example is provided here.

Example 1.34

Recall that the Fibonacci sequence describes a (very unrealistic) model of what happens when a pair rabbits get left alone in a black box... The assumptions are that a pair of rabits never die and produce a pair of offspring every month starting on their second month of life. This system is defined by the recursion relation for the number of rabit pairs $y(n)$ at month n

$$y(n) = y(n-1) + y(n-2) \tag{1.259}$$

with the initial conditions $y(0) = 0$ and $y(1) = 1$. The result is a very fast growth in the sequence. This is why we do not open black boxes.

1.8.8.4 Linear Constant Coefficient Difference Equations Summary

Difference equations are an important mathematical tool for modeling discrete time systems. An important subclass of these is the class of linear constant coefficient difference equations. Linear constant coefficient difference equations are often particularly easy to solve as will be described in the module on solutions to linear constant coefficient difference equations and are useful in describing a wide range of situations that arise in electrical engineering and in other fields.

1.8.9 Solving Linear Constant Coefficient Difference Equations[114]

1.8.9.1 Introduction

The approach to solving linear constant coefficient difference equations is to find the general form of all possible solutions to the equation and then apply a number of conditions to find the appropriate solution. The two main types of problems are initial value problems, which involve constraints on the solution at several consecutive points, and boundary value problems, which involve constraints on the solution at nonconsecutive points.

The number of initial conditions needed for an Nth order difference equation, which is the order of the highest order difference or the largest delay parameter of the output in the equation, is N, and a unique solution is always guaranteed if these are supplied. Boundary value probelms can be slightly more complicated and will not necessarily have a unique solution or even a solution at all for a given set of conditions. Thus, this section will focus exclusively on initial value problems.

1.8.9.2 Solving Linear Constant Coefficient Difference Equations

Consider some linear constant coefficient difference equation given by $Ay(n) = f(n)$, in which A is a difference operator of the form

$$A = a_N D^N + a_{N-1} D^{N-1} + ... + a_1 D + a_0 \tag{1.260}$$

where D is the first difference operator

$$D(y(n)) = y(n) - y(n-1). \tag{1.261}$$

Let $y_h(n)$ and $y_p(n)$ be two functions such that $Ay_h(n) = 0$ and $Ay_p(n) = f(n)$. By the linearity of A, note that $L(y_h(n) + y_p(n)) = 0 + f(n) = f(n)$. Thus, the form of the general solution $y_g(n)$ to any linear constant coefficient ordinary differential equation is the sum of a homogeneous solution $y_h(n)$ to the equation $Ay(n) = 0$ and a particular solution $y_p(n)$ that is specific to the forcing function $f(n)$.

We wish to determine the forms of the homogeneous and nonhomogeneous solutions in full generality in order to avoid incorrectly restricting the form of the solution before applying any conditions. Otherwise, a valid set of initial or boundary conditions might appear to have no corresponding solution trajectory. The following sections discuss how to accomplish this for linear constant coefficient difference equations.

1.8.9.2.1 Finding the Homogeneous Solution

In order to find the homogeneous solution to a difference equation described by the recurrence relation $\sum_{k=0}^{N} a_k y(n-k) = f(n)$, consider the difference equation $\sum_{k=0}^{N} a_k y(n-k) = 0$. We know that the solutions have the form $c\lambda^n$ for some complex constants c, λ. Since $\sum_{k=0}^{N} a_k c\lambda^{n-k} = 0$ for a solution it follows that

$$c\lambda^{n-N} \sum_{k=0}^{N} a_k \lambda^{N-k} = 0 \tag{1.262}$$

so it also follows that

$$a_0 \lambda^N + ... + a_N = 0. \tag{1.263}$$

Therefore, the solution exponential are the roots of the above polynomial, called the characteristic polynomial.

[114] This content is available online at <http://cnx.org/content/m12326/1.6/>.

For equations of order two or more, there will be several roots. If all of the roots are distinct, then the general form of the homogeneous solution is simply

$$y_h(n) = c_1 \lambda_1^n + \ldots + c_2 \lambda_2^n. \tag{1.264}$$

If a root has multiplicity that is greater than one, the repeated solutions must be multiplied by each power of n from 0 to one less than the root multiplicity (in order to ensure linearly independent solutions). For instance, if λ_1 had a multiplicity of 2 and λ_2 had multiplicity 3, the homogeneous solution would be

$$y_h(n) = c_1 \lambda_1^n + c_2 n \lambda_1^n + c_3 \lambda_2^n + c_4 n \lambda_2^n + c_5 n^2 \lambda_2^n. \tag{1.265}$$

Example 1.35

Recall that the Fibonacci sequence describes a (very unrealistic) model of what happens when a pair rabbits get left alone in a black box... The assumptions are that a pair of rabits never die and produce a pair of offspring every month starting on their second month of life. This system is defined by the recursion relation for the number of rabit pairs $y(n)$ at month n

$$y(n) - y(n-1) - y(n-2) = 0 \tag{1.266}$$

with the initial conditions $y(0) = 0$ and $y(1) = 1$.

Note that the forcing function is zero, so only the homogenous solution is needed. It is easy to see that the characteristic polynomial is $\lambda^2 - \lambda - 1 = 0$, so there are two roots with multiplicity one. These are $\lambda_1 = \frac{1+\sqrt{5}}{2}$ and $\lambda_2 = \frac{1-\sqrt{5}}{2}$. Thus, the solution is of the form

$$y(n) = c_1 \left(\frac{1+\sqrt{5}}{2}\right)^n + c_2 \left(\frac{1-\sqrt{5}}{2}\right)^n. \tag{1.267}$$

Using the initial conditions, we determine that

$$c_1 = \frac{\sqrt{5}}{5} \tag{1.268}$$

and

$$c_2 = -\frac{\sqrt{5}}{5}. \tag{1.269}$$

Hence, the Fibonacci sequence is given by

$$y(n) = \frac{\sqrt{5}}{5} \left(\frac{1+\sqrt{5}}{2}\right)^n - \frac{\sqrt{5}}{5} \left(\frac{1-\sqrt{5}}{2}\right)^n. \tag{1.270}$$

1.8.9.2.2 Finding the Particular Solution

Finding the particular solution is a slightly more complicated task than finding the homogeneous solution. It can be found through convolution of the input with the unit impulse response once the unit impulse response is known. Finding the particular solution ot a differential equation is discussed further in the chapter concerning the z-transform, which greatly simplifies the procedure for solving linear constant coefficient differential equations using frequency domain tools.

Example 1.36
Consider the following difference equation describing a system with feedback

$$y(n) - ay(n-1) = x(n). \tag{1.271}$$

In order to find the homogeneous solution, consider the difference equation

$$y(n) - ay(n-1) = 0. \tag{1.272}$$

It is easy to see that the characteristic polynomial is $\lambda - a = 0$, so $\lambda = a$ is the only root. Thus the homogeneous solution is of the form

$$y_h(n) = c_1 a^n. \tag{1.273}$$

In order to find the particular solution, consider the output for the $x(n) = \delta(n)$ unit impulse case

$$y(n) - ay(n-1) = \delta(n). \tag{1.274}$$

By inspection, it is clear that the impulse response is $a^n u(n)$. Hence, the particular solution for a given $x(n)$ is

$$y_p(n) = x(n) * (a^n u(n)). \tag{1.275}$$

Therefore, the general solution is

$$y_g(n) = y_h(n) + y_p(n) = c_1 a^n + x(n) * (a^n u(n)). \tag{1.276}$$

Initial conditions and a specific input can further tailor this solution to a specific situation.

1.8.9.3 Solving Difference Equations Summary

Linear constant coefficient difference equations are useful for modeling a wide variety of discrete time systems. The approach to solving them is to find the general form of all possible solutions to the equation and then apply a number of conditions to find the appropriate solution. This is done by finding the homogeneous solution to the difference equation that does not depend on the forcing function input and a particular solution to the difference equation that does depend on the forcing function input.

1.9 Viewing Embedded LabVIEW Content[115]

In order to view LabVIEW content embedded in Connexions modules, you must install the LabVIEW Run-time Engine on your computer. The following are sets of instructions for installing the software on different platforms.

> NOTE: Embedded LabVIEW content is currently supported only under Windows 2000/XP. Also, you must have version 8.0.1 of the LabView Run-time Engine to run much of the embedded content in Connexions.

[115]This content is available online at <http://cnx.org/content/m13753/1.3/>.

1.9.1 Installing the LabVIEW Run-time Engine on Microsoft Windows 2000/XP

1. Point your web browser to the LabVIEW Run-time Engine download page at: http://digital.ni.com/softlib.nsf/websearch/077b51e8d15604bd8625711c006240e7[116] .
2. If you're not logged in to NI, click the link to continue the download process at the bottom of the page.
3. Login or create a profile with NI to continue.
4. Once logged in, click the `LabVIEW_8.0.1_Runtime_Engine.exe` link and save the file to disk.
5. Once the file has downloaded, double click it and follow the steps to install the run-time engine.
6. Download the LabVIEW Browser Plug-in at: http://zone.ni.com/devzone/conceptd.nsf/webmain/7DBFD404C6AD0B2
.
7. Put the `LVBrowserPlugin.ini` file in the `My Documents\LabVIEW Data` folder. (You may have to create this folder if it doesn't already exist.)
8. Restart your web browser to complete the installation of the plug-in.

[116] http://digital.ni.com/softlib.nsf/websearch/077b51e8d15604bd8625711c006240e7
[117] http://zone.ni.com/devzone/conceptd.nsf/webmain/7DBFD404C6AD0B24862570BB0072F83B/$FILE/LVBrowserPlugin.ini

Solutions to Exercises in Chapter 1

Solution to Exercise 1.4.2.1 (p. 40)
Because of Euler's relation,

$$\sin(2\pi ft) = \frac{1}{2i}e^{i2\pi ft} - \frac{1}{2i}e^{-(i2\pi ft)} \tag{1.277}$$

Thus, $c_1 = \frac{1}{2i}$, $c_{-1} = -\frac{1}{2i}$, and the other coefficients are zero.
Solution to Exercise 1.4.2.2 (p. 43)
$c_0 = \frac{A\Delta}{T}$. This quantity clearly corresponds to the periodic pulse signal's average value.
Solution to Exercise 1.4.3.1 (p. 44)
Write the coefficients of the complex Fourier series in Cartesian form as $c_k = A_k + iB_k$ and substitute into the expression for the complex Fourier series.

$$\sum_{k=-\infty}^{\infty} c_k e^{i\frac{2\pi kt}{T}} = \sum_{k=-\infty}^{\infty} (A_k + iB_k)\, e^{i\frac{2\pi kt}{T}}$$

Simplifying each term in the sum using Euler's formula,

$$\begin{aligned}(A_k + iB_k)\, e^{i\frac{2\pi kt}{T}} &= (A_k + iB_k)\left(\cos\left(\frac{2\pi kt}{T}\right) + i\sin\left(\frac{2\pi kt}{T}\right)\right)\\ &= A_k\cos\left(\frac{2\pi kt}{T}\right) - B_k\sin\left(\frac{2\pi kt}{T}\right) + i\left(A_k\sin\left(\frac{2\pi kt}{T}\right) + B_k\cos\left(\frac{2\pi kt}{T}\right)\right)\end{aligned}$$

We now combine terms that have the same frequency index **in magnitude**. Because the signal is real-valued, the coefficients of the complex Fourier series have conjugate symmetry: $c_{-k} = \overline{c_k}$ or $A_{-k} = A_k$ and $B_{-k} = -B_k$. After we add the positive-indexed and negative-indexed terms, each term in the Fourier series becomes $2A_k\cos\left(\frac{2\pi kt}{T}\right) - 2B_k\sin\left(\frac{2\pi kt}{T}\right)$. To obtain the classic Fourier series (1.97), we must have $2A_k = a_k$ and $2B_k = -b_k$.
Solution to Exercise 1.4.3.2 (p. 45)
The average of a set of numbers is the sum divided by the number of terms. Viewing signal integration as the limit of a Riemann sum, the integral corresponds to the average.
Solution to Exercise 1.4.3.3 (p. 45)
We found that the complex Fourier series coefficients are given by $c_k = \frac{2}{i\pi k}$. The coefficients are pure imaginary, which means $a_k = 0$. The coefficients of the sine terms are given by $b_k = -(2\Im(c_k))$ so that

$$b_k = \begin{cases} \frac{4}{\pi k} & \text{if } k \text{ odd} \\ 0 & \text{if } k \text{ even} \end{cases}$$

Thus, the Fourier series for the square wave is

$$\mathrm{sq}(t) = \sum_{k\in\{1,3,\dots\}} \frac{4}{\pi k}\sin\left(\frac{2\pi kt}{T}\right) \tag{1.278}$$

Solution to Exercise 1.4.4.1 (p. 47)
The rms value of a sinusoid equals its amplitude divided by $\sqrt{22}$. As a half-wave rectified sine wave is zero during half of the period, its rms value is $\frac{A}{2}$ since the integral of the squared half-wave rectified sine wave equals half that of a squared sinusoid.
Solution to Exercise 1.4.4.2 (p. 48)
Total harmonic distortion equals $\frac{\sum_{k=2}^{\infty} {a_k}^2 + {b_k}^2}{{a_1}^2 + {b_1}^2}$. Clearly, this quantity is most easily computed in the frequency domain. However, the numerator equals the square of the signal's rms value minus the power in the average and the power in the first harmonic.
Solution to Exercise 1.4.5.1 (p. 51)
Total harmonic distortion in the square wave is $1 - \frac{1}{2}\left(\frac{4}{\pi}\right)^2 = 20\%$.

Solution to Exercise 1.4.6.1 (p. 54)

N signals directly encoded require a bandwidth of $\frac{N}{T}$. Using a binary representation, we need $\frac{\log_2 N}{T}$. For $N = 128$, the binary-encoding scheme has a factor of $\frac{7}{128} = 0.05$ smaller bandwidth. Clearly, binary encoding is superior.

Solution to Exercise 1.4.6.2 (p. 54)

We can use N different amplitude values at only one frequency to represent the various letters.

Solution to Exercise 1.4.7.1 (p. 57)

Because the filter's gain at zero frequency equals one, the average output values equal the respective average input values.

Solution to Exercise 1.4.8.1 (p. 59)

$$\mathcal{F}(S(f)) = \int_{-\infty}^{\infty} S(f)\, e^{-(i2\pi ft)} df = \int_{-\infty}^{\infty} S(f)\, e^{i2\pi f(-t)} df = s(-t)$$

Solution to Exercise 1.4.8.2 (p. 59)

$\mathcal{F}(\mathcal{F}(\mathcal{F}(\mathcal{F}(s(t))))) = s(t)$. We know that $\mathcal{F}(S(f)) = \int_{-\infty}^{\infty} S(f)\, e^{-(i2\pi ft)} df = \int_{-\infty}^{\infty} S(f)\, e^{i2\pi f(-t)} df = s(-t)$. Therefore, two Fourier transforms applied to $s(t)$ yields $s(-t)$. We need two more to get us back where we started.

Solution to Exercise 1.4.8.3 (p. 61)

The signal is the inverse Fourier transform of the triangularly shaped spectrum, and equals $s(t) = W\left(\frac{\sin(\pi Wt)}{\pi Wt}\right)^2$

Solution to Exercise 1.4.8.4 (p. 62)

The result is most easily found in the spectrum's formula: the power in the signal-related part of $x(t)$ is half the power of the signal $s(t)$.

Solution to Exercise 1.4.9.1 (p. 63)

The inverse transform of the frequency response is $\frac{1}{RC} e^{-\frac{t}{RC}} u(t)$. Multiplying the frequency response by $1 - e^{-(i2\pi f\Delta)}$ means subtract from the original signal its time-delayed version. Delaying the frequency response's time-domain version by Δ results in $\frac{1}{RC} e^{\frac{-(t-\Delta)}{RC}} u(t-\Delta)$. Subtracting from the undelayed signal yields $\frac{1}{RC} e^{\frac{-t}{RC}} u(t) - \frac{1}{RC} e^{\frac{-(t-\Delta)}{RC}} u(t-\Delta)$. Now we integrate this sum. Because the integral of a sum equals the sum of the component integrals (integration is linear), we can consider each separately. Because integration and signal-delay are linear, the integral of a delayed signal equals the delayed version of the integral. The integral is provided in the example (1.130).

Solution to Exercise 1.5.1.1 (p. 81)

In order to calculate the Fourier transform, all we need to use is (1.138) (Continuous-Time Fourier Transform), complex exponentials (Section 1.1.5), and basic calculus.

$$\begin{aligned} \mathcal{F}(\Omega) &= \int_{-\infty}^{\infty} f(t)\, e^{-(i\Omega t)} dt \\ &= \int_0^{\infty} e^{-(\alpha t)} e^{-(i\Omega t)} dt \\ &= \int_0^{\infty} e^{(-t)(\alpha + i\Omega)} dt \\ &= 0 - \frac{-1}{\alpha + i\Omega} \end{aligned} \tag{1.279}$$

$$\mathcal{F}(\Omega) = \frac{1}{\alpha + i\Omega} \tag{1.280}$$

Solution to Exercise 1.5.1.2 (p. 81)

Here we will use (1.139) (Inverse CTFT) to find the inverse FT given that $t \neq 0$.

$$\begin{aligned} x(t) &= \frac{1}{2\pi} \int_{-M}^{M} e^{i(\Omega,t)} d\Omega \\ &= \frac{1}{2\pi} e^{i(\Omega,t)}|_{\Omega,\Omega=e^{iw}} \\ &= \frac{1}{\pi t} \sin(Mt) \end{aligned} \tag{1.281}$$

$$x(t) = \frac{M}{\pi}\left(\text{sinc}\frac{Mt}{\pi}\right) \quad (1.282)$$

Solution to Exercise 1.7.1.1 (p. 100)
The indices can be negative, and this condition is not allowed in MATLAB. To fix it, we must start the signals later in the array.
Solution to Exercise 1.7.1.2 (p. 102)
Such terms would require the system to know what future input or output values would be before the current value was computed. Thus, such terms can cause difficulties.
Solution to Exercise 1.7.3.1 (p. 115)

$$a_k = \{ \begin{array}{cc} \text{undefined} & k=0 \\ \frac{1}{8}\frac{sin^3\left[\frac{\pi}{2}k\right]}{\left[\frac{\pi}{2}k\right]^3} & \text{otherwise} \end{array}$$

Solution to Exercise 1.8.4.1 (p. 126)

$$\alpha \sum_{n=n_0}^{N+n_0-1} \alpha^n - \sum_{n=n_0}^{N+n_0-1} \alpha^n = \alpha^{N+n_0} - \alpha^{n_0} \quad (1.283)$$

which, after manipulation, yields the geometric sum formula.
Solution to Exercise 1.8.6.1 (p. 130)
This situation amounts to aliasing in the time-domain.

Index of Keywords and Terms

Keywords are listed by the section with that keyword (page numbers are in parentheses). Keywords do not necessarily appear in the text of the page. They are merely associated with that section. *Ex.* apples, § 1.1 (1) **Terms** are referenced by the page they appear on. *Ex.* apples, 1

Attributions

Collection: *Communication Systems*
Edited by: Janko Calic
URL: http://cnx.org/content/col10631/1.3/
License: http://creativecommons.org/licenses/by/2.0/

Module: "Signal Classifications and Properties"
By: Melissa Selik, Richard Baraniuk, Michael Haag
URL: http://cnx.org/content/m10057/2.23/
Pages: 1-7
Copyright: Melissa Selik, Richard Baraniuk, Michael Haag
License: http://creativecommons.org/licenses/by/3.0/

Module: "Signal Operations"
By: Richard Baraniuk
URL: http://cnx.org/content/m10125/2.18/
Pages: 7-10
Copyright: Richard Baraniuk, Adam Blair
License: http://creativecommons.org/licenses/by/3.0/

Module: "Common Continuous Time Signals"
By: Melissa Selik, Richard Baraniuk
URL: http://cnx.org/content/m10058/2.17/
Pages: 10-12
Copyright: Melissa Selik, Richard Baraniuk
License: http://creativecommons.org/licenses/by/3.0/

Module: "Continuous Time Impulse Function"
By: Melissa Selik, Richard Baraniuk
URL: http://cnx.org/content/m10059/2.27/
Pages: 12-14
Copyright: Melissa Selik, Richard Baraniuk, Adam Blair
License: http://creativecommons.org/licenses/by/3.0/

Module: "Continuous Time Complex Exponential"
By: Richard Baraniuk, Stephen Kruzick
URL: http://cnx.org/content/m10060/2.25/
Pages: 14-17
Copyright: Richard Baraniuk, Stephen Kruzick, Adam Blair
License: http://creativecommons.org/licenses/by/3.0/

Module: "Discrete-Time Signals"
By: Don Johnson
URL: http://cnx.org/content/m0009/2.24/
Pages: 17-20
Copyright: Don Johnson
License: http://creativecommons.org/licenses/by/1.0

Module: "System Classifications and Properties"
By: Melissa Selik, Richard Baraniuk, Stephen Kruzick
URL: http://cnx.org/content/m10084/2.24/
Pages: 20-24
Copyright: Melissa Selik, Richard Baraniuk, Stephen Kruzick
License: http://creativecommons.org/licenses/by/3.0/

Module: "Linear Time Invariant Systems"
By: Thanos Antoulas, JP Slavinsky
URL: http://cnx.org/content/m2102/2.26/
Pages: 24-30
Copyright: Thanos Antoulas, JP Slavinsky
License: http://creativecommons.org/licenses/by/3.0/

Module: "Continuous Time Convolution"
By: Melissa Selik, Richard Baraniuk, Stephen Kruzick, Dan Calderon
URL: http://cnx.org/content/m10085/2.35/
Pages: 31-35
Copyright: Melissa Selik, Richard Baraniuk, Stephen Kruzick, Dan Calderon, Adam Blair
License: http://creativecommons.org/licenses/by/3.0/

Module: "Properties of Continuous Time Convolution"
By: Melissa Selik, Richard Baraniuk, Stephen Kruzick
URL: http://cnx.org/content/m10088/2.21/
Pages: 35-38
Copyright: Melissa Selik, Richard Baraniuk, Stephen Kruzick
License: http://creativecommons.org/licenses/by/4.0/

Module: "Introduction to the Frequency Domain"
By: Don Johnson
URL: http://cnx.org/content/m0038/2.10/
Page: 39
Copyright: Don Johnson
License: http://creativecommons.org/licenses/by/1.0

Module: "Complex Fourier Series"
By: Don Johnson
URL: http://cnx.org/content/m0042/2.33/
Pages: 39-43
Copyright: Don Johnson
License: http://creativecommons.org/licenses/by/3.0/

Module: "Classic Fourier Series"
By: Don Johnson
URL: http://cnx.org/content/m0039/2.24/
Pages: 43-45
Copyright: Don Johnson
License: http://creativecommons.org/licenses/by/3.0/

Module: "A Signal's Spectrum"
By: Don Johnson
URL: http://cnx.org/content/m0040/2.21/
Pages: 45-48
Copyright: Don Johnson
License: http://creativecommons.org/licenses/by/1.0

Module: "Fourier Series Approximation of Signals"
By: Don Johnson
URL: http://cnx.org/content/m10687/2.11/
Pages: 48-53
Copyright: Don Johnson
License: http://creativecommons.org/licenses/by/3.0/

Module: "Encoding Information in the Frequency Domain"
By: Don Johnson
URL: http://cnx.org/content/m0043/2.17/
Pages: 53-55
Copyright: Don Johnson
License: http://creativecommons.org/licenses/by/1.0

Module: "Filtering Periodic Signals"
By: Don Johnson
URL: http://cnx.org/content/m0044/2.12/
Pages: 55-57
Copyright: Don Johnson
License: http://creativecommons.org/licenses/by/3.0/

Module: "Derivation of the Fourier Transform"
By: Don Johnson
URL: http://cnx.org/content/m0046/2.22/
Pages: 57-62
Copyright: Don Johnson
License: http://creativecommons.org/licenses/by/3.0/

Module: "Linear Time Invariant Systems"
By: Don Johnson
URL: http://cnx.org/content/m0048/2.19/
Pages: 62-64
Copyright: Don Johnson
License: http://creativecommons.org/licenses/by/3.0/

Module: "Frequency Domain Problems"
By: Don Johnson
URL: http://cnx.org/content/m10350/2.43/
Pages: 64-78
Copyright: Don Johnson
License: http://creativecommons.org/licenses/by/4.0/

Module: "Continuous Time Fourier Transform (CTFT)"
By: Richard Baraniuk, Melissa Selik
URL: http://cnx.org/content/m10098/2.17/
Pages: 79-82
Copyright: Richard Baraniuk, Melissa Selik
License: http://creativecommons.org/licenses/by/3.0/

Module: "Properties of the CTFT"
By: Melissa Selik, Richard Baraniuk
URL: http://cnx.org/content/m10100/2.16/
Pages: 82-85
Copyright: Melissa Selik, Richard Baraniuk
License: http://creativecommons.org/licenses/by/3.0/

Module: "Introduction"
By: Anders Gjendemsjø
URL: http://cnx.org/content/m11419/1.29/
Pages: 85-87
Copyright: Anders Gjendemsjø
License: http://creativecommons.org/licenses/by/1.0

Module: "Proof"
By: Anders Gjendemsjø
URL: http://cnx.org/content/m11423/1.27/
Pages: 87-90
Copyright: Anders Gjendemsjø
License: http://creativecommons.org/licenses/by/1.0

Module: "Illustrations"
By: Anders Gjendemsjø
URL: http://cnx.org/content/m11443/1.33/
Pages: 90-94
Copyright: Anders Gjendemsjø
License: http://creativecommons.org/licenses/by/1.0

Module: "Systems view of sampling and reconstruction"
By: Anders Gjendemsjø
URL: http://cnx.org/content/m11465/1.20/
Pages: 94-96
Copyright: Anders Gjendemsjø
License: http://creativecommons.org/licenses/by/1.0

Module: "Sampling CT Signals: A Frequency Domain Perspective"
By: Robert Nowak
URL: http://cnx.org/content/m10994/2.2/
Pages: 96-99
Copyright: Robert Nowak
License: http://creativecommons.org/licenses/by/1.0

Module: "Discrete-Time Systems in the Time-Domain"
By: Don Johnson
URL: http://cnx.org/content/m10251/2.25/
Pages: 99-103
Copyright: Don Johnson
License: http://creativecommons.org/licenses/by/3.0/

Module: "Discrete Time Convolution"
By: Ricardo Radaelli-Sanchez, Richard Baraniuk, Stephen Kruzick, Catherine Elder
URL: http://cnx.org/content/m10087/2.30/
Pages: 103-108
Copyright: Ricardo Radaelli-Sanchez, Richard Baraniuk, Stephen Kruzick
License: http://creativecommons.org/licenses/by/4.0/

Module: "Discrete Time Circular Convolution and the DTFS"
By: Justin Romberg
URL: http://cnx.org/content/m10786/2.16/
Pages: 108-116
Copyright: Justin Romberg
License: http://creativecommons.org/licenses/by/3.0/

Module: "Discrete Time Fourier Transform (DTFT)"
By: Richard Baraniuk
URL: http://cnx.org/content/m10108/2.18/
Pages: 117-120
Copyright: Richard Baraniuk
License: http://creativecommons.org/licenses/by/3.0/

Module: "Properties of the DTFT"
By: Don Johnson
URL: http://cnx.org/content/m0506/2.8/
Pages: 120-123
Copyright: Don Johnson
License: http://creativecommons.org/licenses/by/3.0/

Module: "Discrete Time Fourier Transform Pair"
By: Don Johnson
URL: http://cnx.org/content/m0525/2.7/
Pages: 123-124
Copyright: Don Johnson
License: http://creativecommons.org/licenses/by/1.0

Module: "DTFT Examples"
By: Don Johnson
URL: http://cnx.org/content/m0524/2.11/
Pages: 124-127
Copyright: Don Johnson
License: http://creativecommons.org/licenses/by/1.0

Module: "Discrete Fourier Transformation"
By: Phil Schniter
URL: http://cnx.org/content/m10421/2.11/
Pages: 127-129
Copyright: Phil Schniter
License: http://creativecommons.org/licenses/by/1.0

Module: "Discrete Fourier Transform (DFT)"
By: Don Johnson
URL: http://cnx.org/content/m10249/2.28/
Pages: 129-131
Copyright: Don Johnson
License: http://creativecommons.org/licenses/by/1.0

Module: "Common Fourier Transforms"
By: Melissa Selik, Richard Baraniuk
URL: http://cnx.org/content/m10099/2.12/
Pages: 131-132
Copyright: Melissa Selik, Richard Baraniuk
License: http://creativecommons.org/licenses/by/3.0/

Module: "Linear Constant Coefficient Difference Equations"
By: Richard Baraniuk, Stephen Kruzick
URL: http://cnx.org/content/m12325/1.5/
Pages: 132-133
Copyright: Richard Baraniuk, Stephen Kruzick
License: http://creativecommons.org/licenses/by/3.0/

Module: "Solving Linear Constant Coefficient Difference Equations"
By: Richard Baraniuk, Stephen Kruzick
URL: http://cnx.org/content/m12326/1.6/
Pages: 133-136
Copyright: Richard Baraniuk, Stephen Kruzick
License: http://creativecommons.org/licenses/by/3.0/

Module: "Viewing Embedded LabVIEW Content"
By: Matthew Hutchinson
URL: http://cnx.org/content/m13753/1.3/
Pages: 137-138
Copyright: Matthew Hutchinson
License: http://creativecommons.org/licenses/by/2.0/

Communication Systems
A more advanced communications course covering elements of modern digital communications.

About OpenStax-CNX
Rhaptos is a web-based collaborative publishing system for educational material.