3.1 FINDING THE SPECTRUM OF A SIGNAL

A signal $s(t)$ can often be expressed in analytical form as a function of time t, and the Fourier transform is defined as in (2.1) as the integral of $s(t)e^{-2\pi jft}$. The resulting transform $S(f)$ is a function of frequency. $S(f)$ is called the spectrum of the signal $s(t)$ and describes the frequencies present in the signal. For example, if the time signal is created as a sum of three sine waves, then the spectrum will have spikes corresponding to each of the constituent sines. If the time signal contains only frequencies between 100 and 200 Hz, then the spectrum will be zero for all frequencies outside of this range. A brief guide to Fourier transforms appears in Appendix D, and a summary of all the transforms and properties that are used throughout **Telecommunication Breakdown** appears in Appendix A.

Often however, there is no analytical expression for a signal, that is, there is no (known) equation that represents the value of the signal over time. Instead, the signal is defined by measurements of some physical process. For instance, the signal might be the waveform at the input to the receiver, the output of a linear filter, or a sound waveform encoded as an mp3 file. In all these cases, it is not possible to find the spectrum by calculating a Fourier transform.

Rather, the discrete Fourier transform (and its cousin, the more rapidly computable fast Fourier transform, or FFT) can be used to find the spectrum or frequency content of a measured signal. The `Matlab` function `plotspec.m`, which plots the spectrum of a signal, is available on the CD. Its help file[1] notes:

```
% plotspec(x,Ts) plots the spectrum of the signal x
% Ts = time (in seconds) between adjacent samples in x
```

The function `plotspec.m` is easy to use. For instance, the spectrum of a square wave can be found using:

specsquare.m: plot the spectrum of a square wave

```
f=10;                        % "frequency" of square wave
time=2;                      % length of time
Ts=1/1000;                   % time interval between samples
t=Ts:Ts:time;                % create a time vector
x=sign(cos(2*pi*f*t));       % square wave = sign of cos wave
plotspec(x,Ts)               % call plotspec to draw spectrum
```

[1] You can view the help file for the `Matlab` function `xxx` by typing `help xxx` at the `Matlab` prompt. If you get an error such as `xxx not found`, then this means either that the function does not exist, or that it needs to be moved into the same directory as the `Matlab` application. If you don't know what the proper command to do a job is, then use `lookfor`. For instance, to find the command that inverts a matrix, type `lookfor inverse`. You will find the desired command `inv`.

The output of `specsquare.m` is shown[2] in Figure 3.2. The top plot shows `time=2` seconds of a square wave with `f=10` cycles per second. The bottom plot shows a series of spikes that define the frequency content. In this case, the largest spike occurs at ±10 Hz, followed by smaller spikes at all the odd-integer multiples (i.e., at ±30, ±50, ±70, etc.).

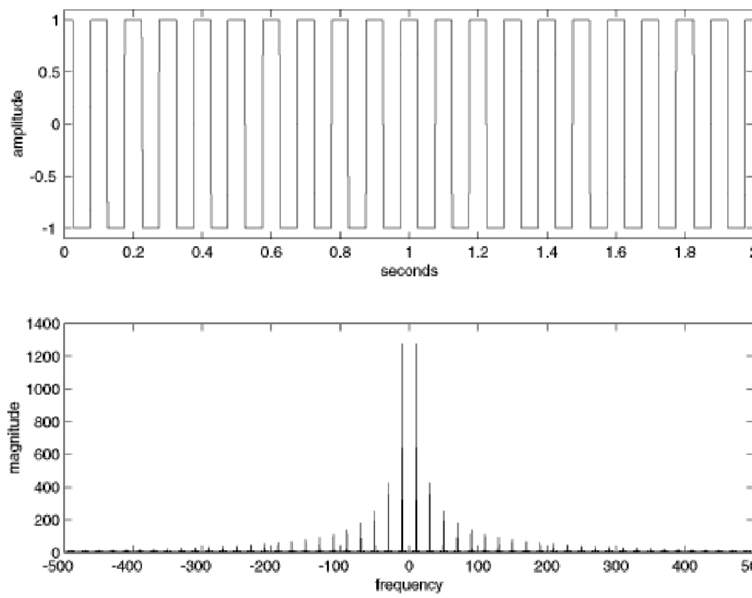

FIGURE 3.2: A square wave and its spectrum, as calculated using `plotspec.m`.

Similarly, the spectrum of a noise signal can be calculated as

specnoise.m: plot the spectrum of a noise signal

```
time=1;                    % length of time
Ts=1/10000;                % time interval between samples
x=randn(1,time/Ts);        % Ts points of noise for time seconds
plotspec(x,Ts)             % call plotspec to draw spectrum
```

A typical run of `specnoise.m` is shown in Figure 3.3. The top plot shows the noisy signal as a function of time, while the bottom shows the magnitude spectrum. Because successive values of the noise are generated independently, all frequencies are roughly equal in magnitude. Each run of `specnoise.m` produces plots that are qualitatively similar, though the details will differ.

PROBLEMS

3.1. Use `specquare.m` to investigate the relationship between the time behavior of the

[2] All code listings in **Telecommunication Breakdown** can be found on the CD. We encourage you to open `Matlab` and explore the code as you read.

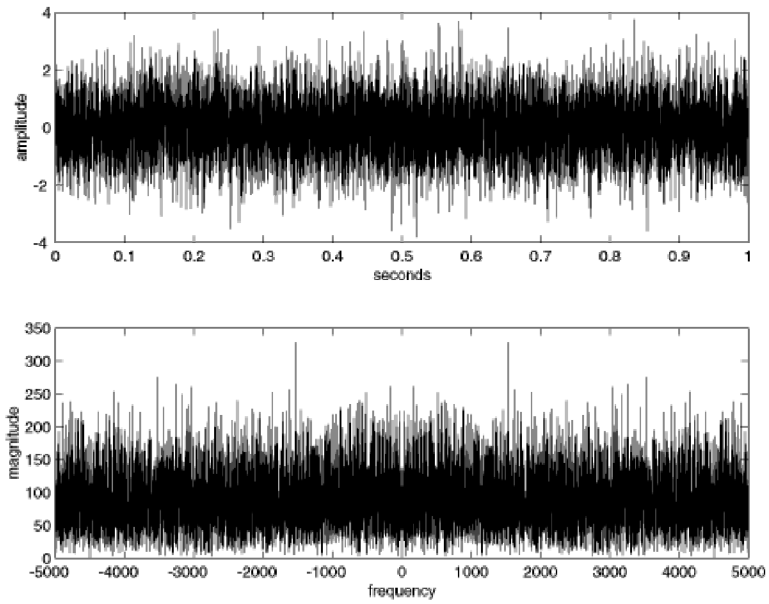

FIGURE 3.3: A noise signal and its spectrum, as calculated using `plotspec.m`.

square wave and its spectrum. The `Matlab` command `zoom on` is often helpful for viewing details of the plots.

(a) Try square waves with different frequencies: `f=20, 40, 100, 300` Hz. How do the time plots change? How do the spectra change?

(b) Try square waves of different lengths, `time=1, 10, 100` seconds. How does the spectrum change in each case?

(c) Try different sampling times, `Ts=1/100, 1/10000`. seconds. How does the spectrum change in each case?

3.2. In your *Signal and Systems* course, you probably calculated (analytically) the spectrum of a square wave using the Fourier series. How does this calculation compare to the discrete data version found by `specsquare.m`?

3.3. Mimic the code in `specsquare.m` to find the spectrum of

(a) an exponential pulse $s(t) = e^{-t}$

(b) a scaled exponential pulse $s(t) = 5e^{-t}$

(c) a Gaussian pulse $s(t) = e^{-t^2}$

(d) the sinusoids $s(t) = \sin(2\pi ft + \phi)$ for $f = 20, 100, 1000$ and $\phi = 0, \pi/4, \pi/2$.

3.4. `Matlab` has several commands that create random numbers.

(a) Use `rand` to create a signal that is uniformly distributed on $[-1, 1]$. Find the spectrum of the signal by mimicking the code in `specnoise.m`.

(b) Use `rand` and the `sign` function to create a signal that is $+1$ with probability $1/2$ and -1 with probability $1/2$. Find the spectrum of the signal.

(c) Use `randn` to create a signal that is normally distributed with mean 0 and variance 3. Find the spectrum of the signal.

While `plotspec.m` can be quite useful, ultimately it will be necessary to have more flexibility, which in turn requires understanding how the `FFT` function inside `plotspec.m` works. This will be discussed at length in Chapter 7. The next five sections describe the five elements that are at the heart of communications systems. The elements are described in both the time domain and in the frequency domain.

3.2 THE FIRST ELEMENT: OSCILLATORS

The Latin word *oscillare* means "to ride in a swing". It is the origin of *oscillate*, which means to move back and forth in steady unvarying rhythm. Thus, a device that creates a signal that moves back and forth in a steady, unvarying rhythm is called an *oscillator*. An electronic oscillator is a device that produces a repetitive electronic signal, usually a sinusoidal wave.

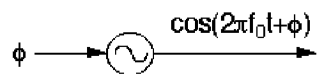

FIGURE 3.4: An oscillator creates a sinusoidal oscillation with a specified frequency f_0 and input ϕ.

A basic oscillator is diagrammed in Figure 3.4. Oscillators are typically designed to operate at a specified frequency f_0, and the input specifies the phase ϕ of the output waveform

$$s(t) = \cos(2\pi f_0 t + \phi).$$

The input may be a fixed number, but it may also be a signal, that is, it may change over time. In this case, the output is no longer a pure sinusoid of frequency f_0. For instance, suppose the phase is a 'ramp' or line with slope $2\pi c$, that is, $\phi(t) = 2\pi ct$. Then $s(t) = \cos(2\pi f_0 t + 2\pi ct) = \cos(2\pi(f_0 + c)t)$, and the 'actual' frequency of oscillation is $f_0 + c$.

There are many ways to build oscillators from analog components. Generally, there is an amplifier and a feedback circuit that returns a portion of the amplified wave back to the input. When the feedback is aligned properly in phase, sustained oscillations occur.

Digital oscillators are simpler, since they can be directly calculated; no amplifier or feedback are needed. For example, a 'digital' sine wave of frequency f Hz and a phase of ϕ radians can be represented mathematically as

$$s(kT_s) = \cos(2\pi f kT_s + \phi) \tag{3.1}$$

where T_s is the time between samples and where k is an integer counter $k = 1, 2, 3, \ldots$. Equation (3.1) can be directly implemented in `Matlab`:

speccos.m: plot the spectrum of a cosine wave

```
f=10; phi=0;                 % specify frequency and phase
time=2;                      % length of time
Ts=1/100;                    % time interval between samples
```

```
t=Ts:Ts:time;              % create a time vector
x=cos(2*pi*f*t+phi);       % create cos wave
plotspec(x,Ts)             % draw waveform and spectrum
```

The output of `speccos.m` is shown in Figure 3.5. As expected, the time plot shows an undulating sinusoidal signal with $f = 10$ repetitions in each second. The actual data is discrete, with one hundred data points in each second. Do not be fooled by the default method of plotting where `Matlab` 'connects the dots' with short line segments for a smoother appearance. The spectrum shows two spikes, one at $f = 10$ Hz and one at $f = -10$ Hz. Why are there *two* spikes? Basic Fourier theory shows that the Fourier transform of a cosine wave is a pair of delta functions at plus and minus the frequency of the cosine wave (see Appendix (A.18)). The two spikes of Figure 3.5 mirror these two delta functions. Alternatively, recall that a cosine wave can be written using Euler's formula as the sum of two complex exponentials, as in (A.2). The spikes of Figure 3.5 represent the magnitudes of these two (complex valued) exponentials.

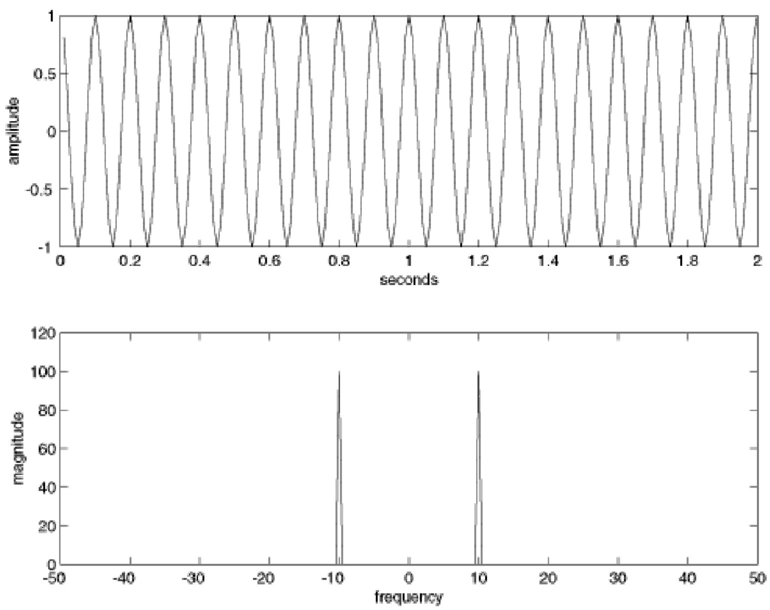

FIGURE 3.5: A sinusoidal oscillator creates a signal that can be viewed in the time domain as in the top plot, or in the frequency domain as in the bottom plot.

PROBLEMS

3.5. Mimic the code in `speccos.m` to find the spectrum of a cosine wave:

- **(a)** for different frequencies `f=1, 2, 20, 30` Hz.
- **(b)** for different phases $\phi = 0, 0.1, \pi/8, \pi/2$ radians.
- **(c)** for different sampling rates `Ts=1/10, 1/1000, 1/100000`.

3.6. Let $x_1(t)$ be a cosine wave of frequency $f = 10$, $x_2(t)$ be a cosine wave of frequency $f = 18$, and $x_3(t)$ be a cosine wave of frequency $f = 33$. Let $x(t) = x_1(t) + 0.5 * x_2(t) + 2 * x_3(t)$. Find the spectrum of $x(t)$. What property of the Fourier transform does this illustrate?

3.7. Find the spectrum of a cosine wave when

(a) ϕ is a function of time. Try $\phi(t) = 10\pi t$.

(b) ϕ is a function of time. Try $\phi(t) = \pi t^2$.

(c) f is a function of time. Try $f(t) = sin(2\pi t)$.

(d) f is a function of time. Try $f(t) = t^2$.

4.2 LINEAR SYSTEMS: LINEAR FILTERS

Linear systems appear in many places in communication systems. The transmission channel is often modeled as a linear system as in (4.1). The bandpass filters used in the front end to remove other users (and to remove noises) are linear. Lowpass filters are crucial to the operation of the demodulators of Chapter 5. The equalizers of Chapter 14 are linear filters that are designed during the operation of the receiver based on certain characteristics of the received signal.

Linear systems can be described in any one of three equivalent ways.

- The *impulse response* is a function of time $h(t)$ that defines the output of a linear system when the input is an impulse (or δ) function. When the input to the linear system is more complicated than a single impulse, the output can be calculated from the impulse response via the *convolution* operator.
- The *frequency response* is a function of frequency $H(f)$ that defines how the spectrum of the input is changed into the spectrum of the output. The frequency response and the impulse response are intimately related: $H(f)$ is the Fourier transform of $h(t)$. Sometimes $H(f)$ is called the *transfer function*.
- A linear *difference or differential equation* (such as (4.1)) shows explicitly how the linear system can be implemented and can be useful in assessing stability and performance.

This chapter describes the three representations of linear systems and shows how they inter-relate. The discussion begins by exploring the δ-function, and then showing how it is used to define the impulse response. The convolution property of the Fourier transform then shows that the transform of the impulse response describes how the system behaves in terms of the input and output spectra, and so is called the frequency response. The final step is to show how the action of the linear system can be redescribed in the time domain as a difference (or as a differential) equation. This is postponed to Chapter 7, and is also discussed in some detail in Appendix F.

4.3 THE DELTA "FUNCTION"

One way to see how a system behaves is to kick it and see how it responds. Some systems react sluggishly, barely moving away from their resting state, while others respond quickly and vigorously. Defining exactly what is meant mathematically by a "kick" is trickier than it seems because the kick must occur over a very short amount of time, yet must be energetic in order to have any effect. This section defines the impulse (or delta) function $\delta(t)$, which is a useful "kick" for the study of linear systems.

The criterion that the impulse be energetic is translated to the mathematical statement that its integral over all time must be nonzero, and it is typically scaled to unity, that is,

$$\int_{-\infty}^{\infty} \delta(t)dt = 1. \tag{4.2}$$

The criterion that it occur over a very short time span is translated to the statement that for every positive ϵ

$$\delta(t) = \begin{cases} 0, & t < -\epsilon \\ 0, & t > \epsilon \end{cases} . \tag{4.3}$$

Thus the impulse $\delta(t)$ is explicitly defined to be equal to zero for all $t \neq 0$. On the other hand, $\delta(t)$ is implicitly defined when $t = 0$ by the requirement that its integral be unity. Together, these guarantee that $\delta(t)$ is no ordinary function[1].

The most important consequence of the definitions (4.2) and (4.3) is the *sifting property*

$$\int_{-\infty}^{\infty} w(t)\delta(t-t_0)dt = w(t)|_{t=t_0} = w(t_0) \tag{4.4}$$

which says that the delta function picks out the value of the function $w(t)$ from under the integral at exactly the time when the argument of the δ function is zero, that is, when $t = t_0$. To see this, observe that $\delta(t-t_0)$ is zero whenever $t \neq t_0$, and hence $w(t)\delta(t-t_0)$ is zero whenever $t \neq t_0$. Thus

$$\begin{aligned} \int_{-\infty}^{\infty} w(t)\delta(t-t_0)dt &= \int_{-\infty}^{\infty} w(t_0)\delta(t-t_0)dt \\ &= w(t_0)\int_{-\infty}^{\infty} \delta(t-t_0)dt = w(t_0)\cdot 1 = w(t_0). \end{aligned}$$

Sometimes it is helpful to think of the impulse as a limit. For instance, define the recatangular pulse of width $1/n$ and height n by

$$\delta_n(t) = \begin{cases} 0, & t < -1/2n \\ n, & -1/2n \leq t \leq 1/2n \\ 0, & t > 1/2n \end{cases} .$$

Then $\delta(t) = \lim_{n\to\infty} \delta_n(t)$ fulfills both criteria (4.2) and (4.3). Informally, it is not unreasonable to think of $\delta(t)$ as being zero everywhere except at $t = 0$, where it is infinite. While it is not really possible to "plot" the delta function $\delta(t-t_0)$, it

[1] The impulse is called a *distribution* and is the subject of considerable mathematical investigation.

can be represented in graphical form as zero everywhere except for an up-pointing arrow at t_0. When the δ function is scaled by a constant, the value of the constant is often placed in parenthesis near the arrowhead. Sometimes, when the constant is negative, the arrow is drawn pointing down. For instance, Figure 4.7 shows a graphical representation of the function $w(t) = \delta(t+10) - 2\delta(t+1) + 3\delta(t-5)$.

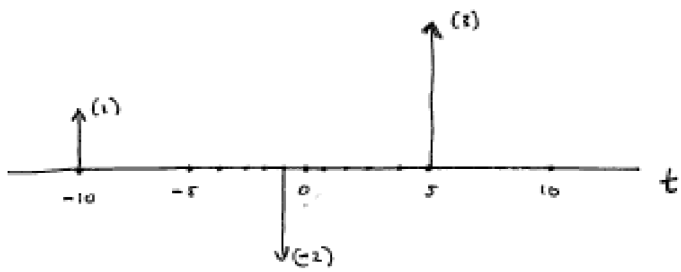

FIGURE 4.7: The function $w(t) = \delta(t+10) - 2\delta(t+1) + 3\delta(t-5)$ consisting of three weighted δ functions is represented graphically as three weighted arrows at $t = -10, -1, 5$, weighted by the appropriate constants.

What is the spectrum (Fourier transform) of $\delta(t)$? This can be calculated directly from the definition by replacing $w(t)$ in (2.1) with $\delta(t)$:

$$\mathcal{F}\{\delta(t)\} = \int_{-\infty}^{\infty} \delta(t) e^{-j2\pi f t} dt. \tag{4.5}$$

Apply the sifting property (4.4) with $w(t) = e^{-j2\pi f t}$ and $t_0 = 0$. Thus $\mathcal{F}\{\delta(t)\} = e^{-j2\pi f t}|_{t=0} = 1$.

Alternatively, suppose that δ is a function of frequency, that is, a spike at zero frequency. The corresponding time domain function can be calculated analogously using the definition of the inverse Fourier transform, that is, by substituting $\delta(f)$ for $W(f)$ in (A.16) and integrating:

$$\mathcal{F}^{-1}\{\delta(f)\} = \int_{-\infty}^{\infty} \delta(f) e^{j2\pi f t} df = e^{-j2\pi f t}|_{f=0} = 1.$$

Thus a spike at frequency zero is a "DC signal" (a constant) in time.

The discrete time counterpart of $\delta(t)$ is the (discrete) delta function

$$\delta[k] = \begin{cases} 1, & k = 0 \\ 0, & k \neq 0 \end{cases}.$$

While there are a few subtleties (i.e., differences) between $\delta(t)$ and $\delta[k]$, for the most part they act analogously. For example, the program `specdelta.m` calculates the spectrum of the (discrete) delta function.

specdelta.m plot the spectrum of a delta function

```
time=2;                    % length of time
Ts=1/100;                  % time interval between samples
t=Ts:Ts:time;              % create time vector
x=zeros(size(t));          % create signal of all zeros
x(1)=1;                    % delta function
plotspec(x,Ts)             % draw waveform and spectrum
```

The output of `specdelta.m` is shown in Figure 4.8. As expected from (4.5), the magnitude spectrum of the delta function is equal to 1 at all frequencies.

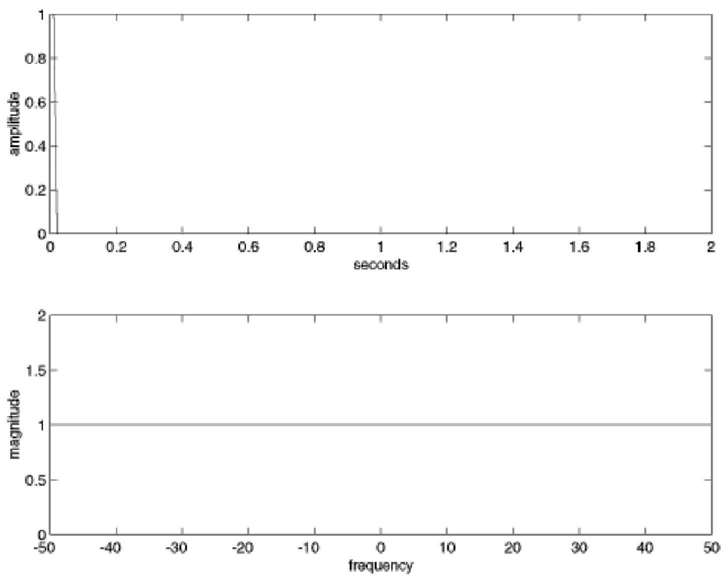

FIGURE 4.8: A (discrete) delta function at time 0 has a magnitude spectrum equal to 1 for all frequencies.

PROBLEMS

4.1. Calculate the Fourier transform of $\delta(t - t_0)$ from the definition. Now calculate it using the time shift property (A.38). Are they the same? Hint: They better be.

4.2. Use the definition of the IFT (D.2) to show that

$$\delta(f - f_0) \Leftrightarrow e^{j2\pi f_0 t}.$$

4.3. Mimic the code in `specdelta.m` to find the spectrum of the discrete delta function when

(a) the delta does not occur at the start of `x`. Try `x[10]=1`, `x[100]=1`, and `x[110]=1`. How do the spectra differ? Can you use the time shift property (A.38) to explain what you see?

(b) the delta changes magnitude `x`. Try `x[1]=10`, `x[10]=3`, and `x[110]=0.1`. How do the spectra differ? Can you use the linearity property (A.31) to explain what you see?

4.4. Mimic the code in `specdelta.m` to find the spectrum of a signal containing two delta functions when

(a) the deltas are located at the start and the end, i.e., `x(1)=1; x(end)=1;`

(b) the deltas are located symmetrically from the start and end, for instance, `x(90)=1; x(end-90)=1;`

(c) the deltas are located arbitrarily, for instance, `x(33)=1; x(120)=1;`

4.5. Mimic the code in `specdelta.m` to find the spectrum of a train of equally spaced pulses. For instance, `x(1:20:end)=1` spaces the pulses 20 samples apart, and `x(1:25:end)=1` places the pulses 25 samples apart.

(a) Can you predict how far apart the resulting pulses in the spectrum will be?

(b) Show that

$$\sum_{k=-\infty}^{\infty} \delta(t-kT_s) \quad \Leftrightarrow \quad \frac{1}{T_s}\sum_{n=-\infty}^{\infty} \delta(f-nf_s) \tag{4.6}$$

where $f_s = 1/T_s$. Hint: Let $w(t) = 1$ in (A.27) and (A.28).

(c) Now can you predict how far apart the pulses in the spectrum are? Your answer should be in terms of how far apart the pulses are in the time signal.

In Section 3.2, the spectrum of a sinusoid was shown to consist of two symmetrical spikes in the frequency domain, (recall Figure 3.5 on page 54). The next example shows why this is true by explicitly taking the Fourier transform.

EXAMPLE 4.1 Spectrum of a Sinusoid

Let $w(t) = A\sin(2\pi f_0 t)$, and use Euler's identity (A.3) to rewrite $w(t)$ as

$$w(t) = \frac{A}{2j}\left[e^{j2\pi f_0 t} - e^{-j2\pi f_0 t}\right].$$

Applying the linearity property (A.31) and the result of Exercise 4.2 gives

$$\begin{aligned} \mathcal{F}\{w(t)\} &= \frac{A}{2j}\left[\mathcal{F}\{e^{j2\pi f_0 t}\} - \mathcal{F}\{e^{-j2\pi f_0 t}\}\right] \\ &= j\frac{A}{2}\left[-\delta(f-f_0) + \delta(f+f_0)\right]. \end{aligned} \tag{4.7}$$

Thus, the magnitude spectrum of a sine wave is a pair of δ functions with opposite signs, located symmetrically about zero frequency, as shown in Figure 4.9. This magnitude spectrum is at the heart of one important interpretation of the Fourier transform: it shows the frequency content of any signal by displaying which frequencies are present (and which frequencies are absent) from the waveform. For example, Figure 4.10(a) shows the magnitude spectrum $W(f)$ of a real valued signal $w(t)$. This can be interpreted as saying that $w(t)$ contains (or is made up of) "all the frequencies" up to B Hz, and that it contains no sinusoids with higher frequency. Similarly, the modulated signal $s(t)$ in Figure 4.10(b) contains all positive frequencies between $f_c - B$ and $f_c + B$, and no others.

Note that the Fourier transform in (4.7) is purely imaginary, as it must be because $w(t)$ is odd (see A.37). The phase spectrum is a flat line at $-90°$ because of the factor j.

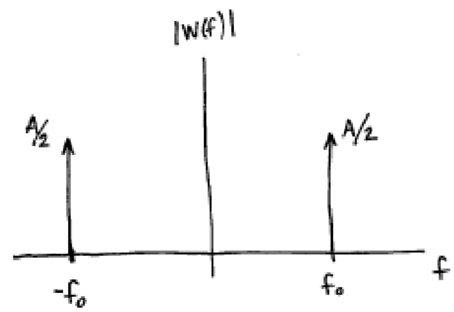

FIGURE 4.9: Magnitude spectrum of a sinusoid with frequency f_0 and amplitude A contains two δ function spikes, one at $f = f_0$ and the other at $f = -f_0$.

PROBLEMS

4.6. What is the magnitude spectrum of $\sin(2\pi f_0 t + \theta)$?. Hint: Use the frequency shift property (A.34). Show that the spectrum of $\cos(2\pi f_0 t)$ is $\frac{1}{2}(\delta(f - f_0) + \delta(f + f_0))$. Compare this analytical result to the numerical results from Exercise 3.5.

4.7. Let $w_i(t) = a_i \sin(2\pi f_i t)$ for $i = 1, 2, 3$. Without doing any calculations, write down the spectrum of $v(t) = w_1(t) + w_2(t) + w_3(t)$. Hint: Use linearity. Graph the magnitude spectrum of $v(t)$ in the same manner as in Figure 4.9. Verify your results with a simulation mimicking that in Exercise 3.6.

4.8. Let $W(f) = \sin(2\pi f t_0)$. What is the corresponding time function?

4.4 CONVOLUTION IN TIME: IT'S WHAT LINEAR SYSTEMS DO

Suppose that a system has impulse response $h(t)$, and that the input consists of a sum of three impulses occurring at times t_0, t_1, and t_2, with amplitudes a_0, a_1, and a_2 (for example, the signal $w(t)$ of Figure 4.7). By linearity of the Fourier transform, property (A.31), the output is a superposition of the outputs due to each of the input pulses. The output due to the first impulse is $a_0 h(t - t_0)$, which is the impulse response scaled by the size of the input and shifted to begin when the first input pulse arrives. Similarly, the outputs to the second and third input impulses are $a_1 h(t - t_1)$ and $a_2 h(t - t_2)$, respectively, and the complete output is the sum $a_0 h(t - t_0) + a_1 h(t - t_1) + a_2 h(t - t_2)$.

Now suppose that the input is a continuous function $x(t)$. At any time instant λ, the input can be thought of as consisting of an impulse scaled by the amplitude $x(\lambda)$, and the corresponding output will be $x(\lambda)h(t - \lambda)$, which is the impulse

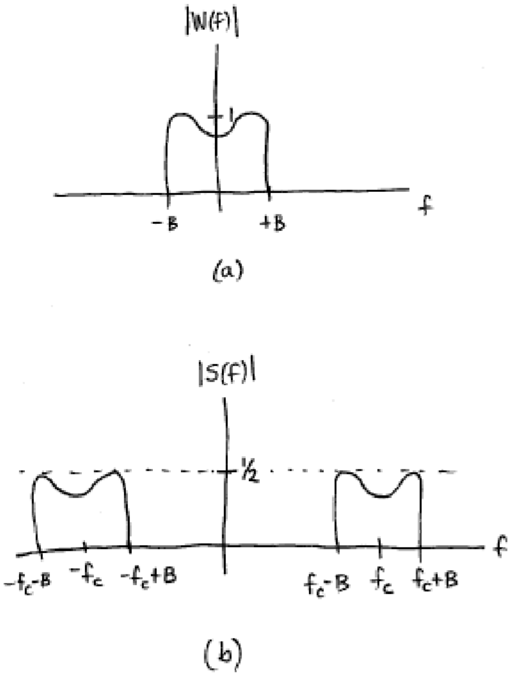

FIGURE 4.10: The magnitude spectrum of a message signal $w(t)$ is shown in (a). When $w(t)$ is modulated by a cosine at frequency f_c, the spectrum of the resulting signal $s(t) = w(t)\cos(2\pi f_c t + \phi)$ is shown in (b).

response scaled by the size of the input and shifted to begin at time λ. The complete output is then given by summing over all λ. Since there is a continuum of possible values of λ, this "sum" is actually an integral, and the output is

$$y(t) = \int_{-\infty}^{\infty} x(\lambda)h(t-\lambda)d\lambda \equiv x(t) * h(t). \tag{4.8}$$

This integral defines the convolution operator $*$ and provides a way of finding the output $y(t)$ of any linear system, given its impulse response $h(t)$ and the input $x(t)$.

Matlab has several functions that simplify the numerical evaluation of convolutions. The most obvious of these is **conv**, which is used in **convolex.m** to calculate the convolution of an input **x** (consisting of two delta functions at times $t = 1$ and $t = 3$) and a system with impulse response **h** that is an exponential pulse. The convolution gives the output of the system.

convolex.m: example of numerical convolution

```
Ts=1/100; time=10;              % sampling interval and total time
t=0:Ts:time;                    % create time vector
h=exp(-t);                      % define impulse response
x=zeros(size(t));               % input is sum of two delta functions...
x(1/Ts)=3; x(3/Ts)=2;           % ...at times t=1 and t=3
y=conv(h,x);                    % do convolution
subplot(3,1,1), plot(t,x)       % and plot
subplot(3,1,2), plot(t,h)
subplot(3,1,3), plot(t,y(1:length(t)))
```

Figure 4.11 shows the input to the system in the top plot, the impulse response in the middle plot, and the output of the system in the bottom plot. Nothing happens before time $t = 1$, and the output is zero. When the first spike occurs, the system responds by jumping to 3 and then decaying slowly at a rate dictated by the shape of $h(t)$. The decay continues smoothly until time $t = 3$, when the second spike enters. At this point, the output jumps up by 2, and is the sum of the response to the second spike, plus the remainder of the response to the first spikes. Since there are no more inputs, the output slowly dies away.

PROBLEMS

4.9. Suppose that the impulse response $h(t)$ of a linear system is the exponential pulse

$$h(t) = \begin{cases} e^{-t} & t \geq 0 \\ 0 & t < 0 \end{cases}. \tag{4.9}$$

Suppose that the input to the system is $3\delta(t-1) + 2\delta(t-3)$. Use the definition of convolution (4.8) to show that the output is $3h(t-1) + 2h(t-3)$ where

$$h(t - t_0) = \begin{bmatrix} e^{-t+t_0} & t \geq t_0 \\ 0 & t < t_0 \end{bmatrix}.$$

How does your answer compare to Figure 4.11?

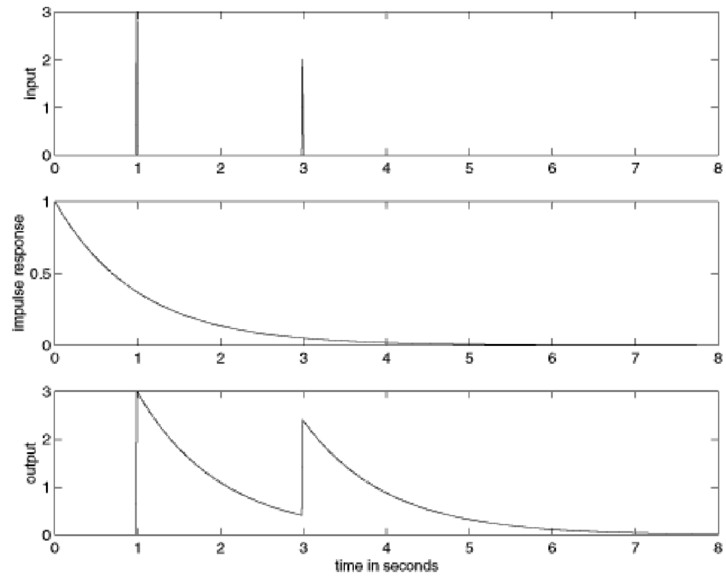

FIGURE 4.11: The convolution of the input (the top plot) with the impulse response of the system (the middle plot) gives the output in the bottom plot.

4.10. Suppose that a system has an impulse response that is an exponential pulse. Mimic the code in `convolex.m` to find its output when the input is a white noise (recall `specnoise.m` on page 52).

4.11. Mimic the code in `convolex.m` to find the output of a system when the input is an exponential pulse and the impulse response is a sum of two delta functions at times $t = 1$ and $t = 3$.

The next two Problems show that linear filters commute with differentiation, and with each other.

PROBLEMS

4.12. Use the definition to show that convolution is commutative, i.e., that $w_1(t) * w_2(t) = w_2(t) * w_1(t)$. Hint: Apply the change of variables $\tau = t - \lambda$ in (4.8).

4.13. Suppose a filter has impulse response $h(t)$. When the input is $x(t)$, the output is $y(t)$. If the input is $x_d(t) = \frac{\partial x(t)}{\partial t}$, the output is $y_d(t)$. Show that $y_d(t)$ is the derivative of $y(t)$. Hint: Use (4.8) and the result of Problem 4.12.

4.14. Let $w(t) = \Pi\left(\frac{t}{T}\right)$ be the rectangular pulse of (2.7). What is $w(t) * w(t)$? Hint: A pulse shaped like a triangle.

4.15. Redo Problem 4.14 numerically by suitably modifying `convolex.m`. Let $T = 1.5$ seconds.

4.16. Suppose that a system has an impulse response that is a sinc function (as defined in (2.8)), and that the input to the system is a white noise (as in `specnoise.m` on page 52).

(a) Mimic `convolex.m` to numerically find the output.

(b) Plot the spectrum of the input and the spectrum of the output (using `plotspec.m`). What kind of filter would you call this?

4.5 CONVOLUTION ⇔ MULTIPLICATION

While the convolution operator (4.8) describes mathematically how a linear system acts on a given input, time domain approaches are often not particularly revealing about the general behavior of the system. Who would guess, for instance in Problem 4.16, that convolution with a sinc function would act like a lowpass filter? By working in the frequency domain, however, the convolution operator is transformed into a simpler point-by-point multiplication, and the generic behavior of the system becomes clearer.

The first step is to understand the relationship between convolution in time, and multiplication in frequency. Suppose that the two time signals $w_1(t)$ and $w_2(t)$ have Fourier transforms $W_1(f)$ and $W_2(f)$. Then,

$$\mathcal{F}\{w_1(t) * w_2(t)\} = W_1(f)W_2(f). \tag{4.10}$$

To justify this property, begin with the definition of the Fourier transform (2.1) and apply the definition of convolution (4.8)

$$\begin{aligned}
\mathcal{F}\{w_1(t) * w_2(t)\} &= \int_{t=-\infty}^{\infty} w_1(t) * w_2(t) e^{-j2\pi f t} dt \\
&= \int_{t=-\infty}^{\infty} \left[\int_{\lambda=-\infty}^{\infty} w_1(\lambda) w_2(t-\lambda) d\lambda\right] e^{-j2\pi f t} dt.
\end{aligned}$$

Reversing the order of integration and using the time shift property (A.38) produces

$$\begin{aligned}
\mathcal{F}\{w_1(t) * w_2(t)\} &= \int_{\lambda=-\infty}^{\infty} w_1(\lambda) \left[\int_{t=-\infty}^{\infty} w_2(t-\lambda) e^{-j2\pi f t} dt\right] d\lambda \\
&= \int_{\lambda=-\infty}^{\infty} w_1(\lambda) \left[W_2(f) e^{-j2\pi f \lambda}\right] d\lambda \\
&= W_2(f) \int_{\lambda=-\infty}^{\infty} w_1(\lambda) e^{-j2\pi f \lambda} d\lambda = W_1(f) W_2(f).
\end{aligned}$$

Thus convolution in the time domain is the same as multiplication in the frequency domain. See (A.40).

The companion to the convolution property is the multiplication property, which says that multiplication in the time domain is equivalent to convolution in the frequency domain (see (A.41)), that is,

$$\mathcal{F}\{w_1(t) w_2(t)\} = W_1(f) * W_2(f) = \int_{-\infty}^{\infty} W_1(\lambda) W_2(f-\lambda) d\lambda. \tag{4.11}$$

The usefulness of these convolution properties is apparent when applying them to linear systems. Suppose that $H(f)$ is the Fourier transform of the impulse response $h(t)$. Suppose that $X(f)$ is the Fourier transform of the input $x(t)$ that is applied to the system. Then (4.8) and (4.10) show that the Fourier transform of the output is exactly equal to the product of the transforms of the input and the impulse response, that is,

$$Y(f) = \mathcal{F}\{y(t)\} = \mathcal{F}\{x(t) * h(t)\} = \mathcal{F}\{h(t)\}\mathcal{F}\{x(t)\} = H(f)X(f).$$

This can be rearranged to solve for

$$H(f) = \frac{Y(f)}{X(f)} \tag{4.12}$$

which is called the *frequency response* of the system because it shows, for each frequency f, how the system responds. For instance, suppose that $H(f_1) = 3$ at some frequency f_1. Then whenever a sinusoid of frequency f_1 is input into the system, it will be amplified by a factor of 3. Alternatively, suppose that $H(f_2) = 0$ at some frequency f_2. Then whenever a sinusoid of frequency f_2 is input into the system, it is removed from the output (because it has been multiplied by a factor of 0).

The frequency response shows how the system treats inputs containing various frequencies. In fact, this property was already used repeatedly in Chapter 1 when drawing curves that describe the behavior of lowpass and bandpass filters. For example, the filters of Figures 2.5, 2.4, and 2.6 are used to remove unwanted frequencies from the communications system. In each of these cases, the plot of the frequency response describes concretely and concisely how the system (or filter) effects the input, and how the frequency content of the output relates to that of the input. Sometimes, the frequency response $H(f)$ is called the *transfer function* of the system, since it "transfers" the input $x(t)$ (with transform $X(f)$) into the output $y(t)$ (with transform $Y(f)$).

Thus, the impulse response describes how a system behaves directly in time, while the frequency response describes how it behaves in frequency. The two descriptions are intimately related because the frequency response is the Fourier transform of the impulse response. This will be used repeatedly in Section 7.2 to design filters for the manipulation (augmentation or removal) of specified frequencies.

EXAMPLE 4.2

In Problem 4.16, a system was defined to have an impulse response that is a sinc function. The Fourier transform of a sinc function in time is a rect function in frequency (A.22). Hence the frequency response of the system is a rectangle that passes all frequencies below $f_c = 1/T$ and removes all frequencies above, i.e., the system is a lowpass filter.

`Matlab` can help to visualize the relationship between the impulse response and the frequency response. For instance, the system in `convolex.m` is defined via its impulse response, which is a decaying exponential. Figure 4.11 shows its output when the input is a simple sum of deltas, and Problem 4.10 explores the output when the input is a white noise. In `freqresp.m`, the behavior of this system is explained by looking at its frequency response.

freqresp.m: numerical example of impulse and frequency response

```
Ts=1/100; time=10;              % sampling interval and total time
t=0:Ts:time;                    % create time vector
h=exp(-t);                      % define impulse response
plotspec(h,Ts)                  % find and plot frequency response
```

The output of `freqresp.m` is shown in Figure 4.12. The frequency response of the system (which is just the magnitude spectrum of the impulse response) is found using `plotspec.m`. In this case, the frequency response amplifies low frequencies and attenuates other frequencies more as the frequency increases. This explains, for instance, why the output of the convolution in Problem 4.10 contained (primarily) lower frequencies, as evidenced by the slower undulations in time.

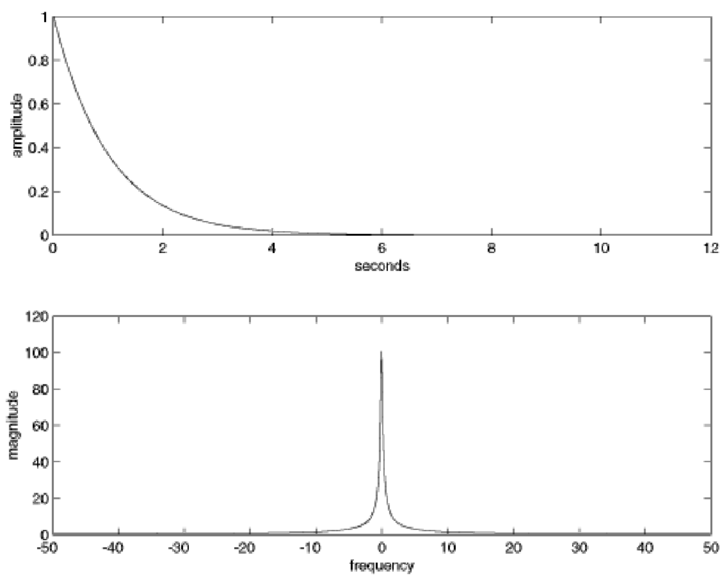

FIGURE 4.12: The action of a system in time is defined by its impulse response (in the top plot). The action of the system in frequency is defined by its frequency response (in the bottom plot), a kind of low pass filter.

PROBLEMS

4.17. Suppose a system has an impulse response that is a sinc function. Using `freqresp.m`, find the frequency response of the system. What kind of filter does this represent? Hint: center the sinc in time, for instance, use `h=sinc(10*(t-time/2));`

4.18. Suppose a system has an impulse response that is a sin function. Using `freqresp.m`, find the frequency response of the system. What kind of filter does this represent? Can you predict the relationship between the frequency of the sine wave and the location of the peaks in the spectrum? Hint: try `h=sin(25*t);`