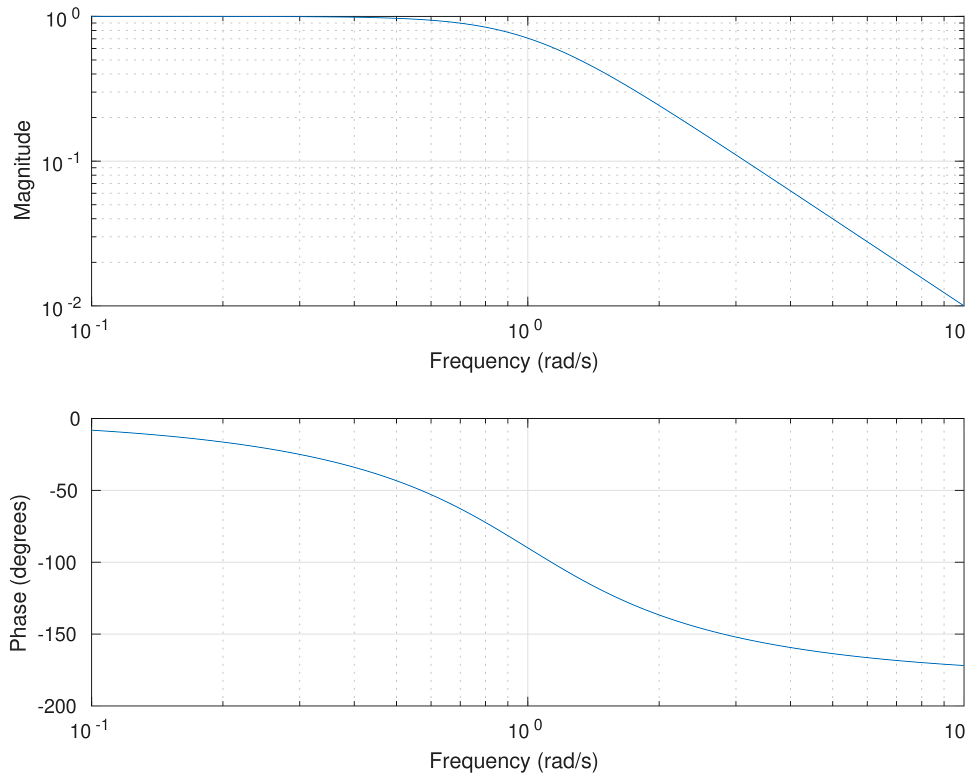


Figure D.5: The frequency response of the filter as produced by the `freqs` function.

Example D.28 (Computing and plotting frequency responses with freqs)**.** Consider the LTI system with transfer function

$$H(s) = \frac{1}{s^2 + \sqrt{2}s + 1}.$$

This system is a second-order Butterworth lowpass filter with a cutoff frequency of 1 rad/s. Suppose that we would like to evaluate the frequency response of this system. This is equivalent to evaluating the transfer function H at points on the imaginary axis. To this end, we can employ the `freqs` function in MATLAB. More specifically, we can calculate and plot the magnitude and phase responses of the above system with the following code:

```
% Initialize the numerator and denominator coefficients of the transfer
% function.
tf_num = [1];
tf_denom = [1 sqrt(2) 1];

% Plot the magnitude and phase responses.
freqs(tf_num, tf_denom);
```

The plot produced by the `freqs` function is shown in Figure D.5. ∎

Example D.29 (Plotting frequency responses)**.** Suppose that we would like to have a function that behaves in a similar manner as the MATLAB `freqs` function, but with a few differences in how plotting is performed. In particular, we would like the magnitude response plotted with a linear (instead of logarithmic) scale and the phase response plotted in unwrapped form. This can be accomplished with the code given in Listing D.5.

Listing D.5: `myfreqs.m`


```
function [freq_resp, omega] = myfreqs(tf_num, tf_denom, omega)
    % The myfreqs function has essentially the same interface as the
    % MATLAB freqs function, but performs plotting slightly differently.
    % The magnitude response is plotted as a unitless quantity (not in
    % decibels).
    % The phase response is plotted with the phase unwrapped.

    % If the frequencies have been specified as an input argument, then simply
    % pass them through to the real freqs function.
    if nargin >= 3
        [freq_resp, omega] = freqs(tf_num, tf_denom, omega);
    else
        [freq_resp, omega] = freqs(tf_num, tf_denom);
    end

    % If no output arguments were specified, plot the frequency response.
    if nargout == 0

        % Compute the magnitude response as a unitless quantity.
        mag_resp = abs(freq_resp);

        % Compute the phase response with the phase unwrapped.
        phase_resp = unwrap(angle(freq_resp)) / pi * 180;

        % On the first of two graphs, plot the magnitude response.
        subplot(2, 1, 1);
        plot(omega, mag_resp);
        title('Magnitude Response');
        xlabel('Frequency (rad/s)');
        ylabel('Magnitude (unitless)');

        % On the second of two graphs, plot the phase response.
        subplot(2, 1, 2);
        plot(omega, phase_resp);
        title('Phase Response');
        xlabel('Frequency (rad/s)');
        ylabel('Angle (degrees)');

    end
end
```

For the filter in Example D.28, the `myfreqs` function produces the frequency-response plots shown in Figure D.6. ■

D.15.1.2 Impulse and Step Responses

Sometimes, we need to determine the response of a LTI system to a specific input. Two inputs of particular interest are the unit-impulse function δ and unit-step function u. Fortunately, it is quite easy to compute impulse and step responses using the `impulse` and `step` functions in MATLAB, as illustrated by the example below.

Example D.30 (Computing impulse and step responses)**.** Consider the LTI system with the transfer function

$$H(s) = \frac{1}{s^2 + \sqrt{2}s + 1}.$$

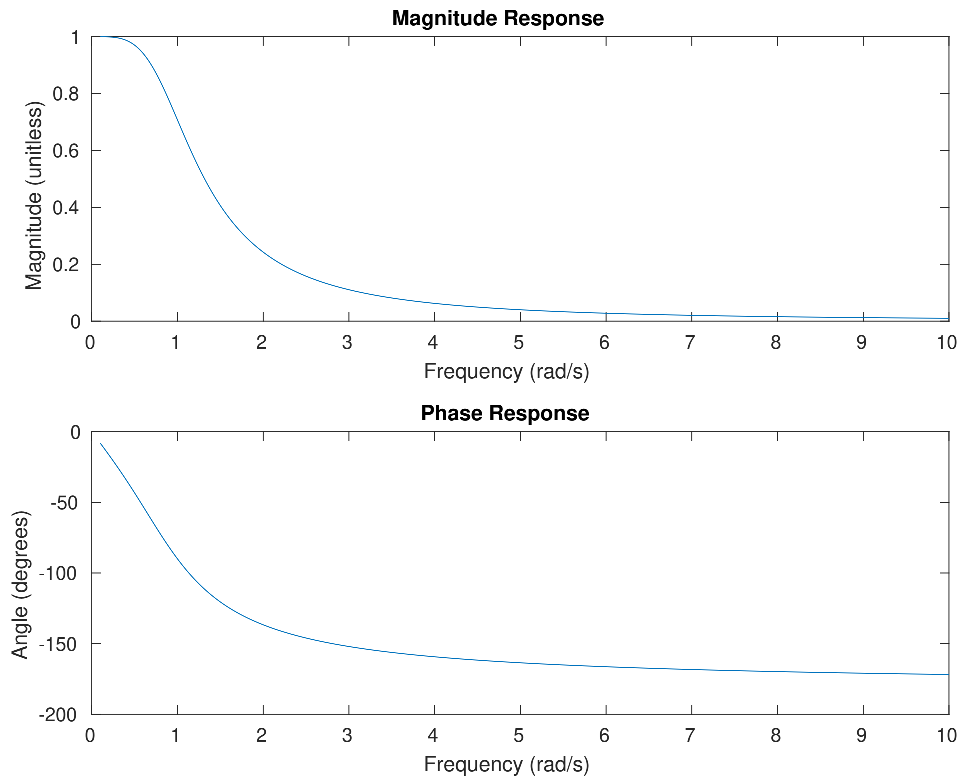


Figure D.6: The frequency response of the filter as produced by the `myfreqs` function.

Suppose that we wish to calculate and plot the impulse and step responses of this system. This can be accomplished with the code given in Listing D.6. Executing this code produces the plots shown in Figure D.7.

Listing D.6: Computing and plotting the impulse and step responses

```
% Initialize the numerator and denominator coefficients of the transfer
% function.
tf_num = [1];
tf_denom = [1 sqrt(2) 1];

% Determine the system model associated with the given transfer function.
sys = tf(tf_num, tf_denom);

% Plot the impulse response.
subplot(2, 1, 1);
impulse(sys);

% Plot the step response.
subplot(2, 1, 2);
step(sys);
```

■

D.15.1.3 Filter Design

A number of functions are provided in MATLAB to assist in the design of various types of filters. In what follows, we consider a few examples of using such functions.

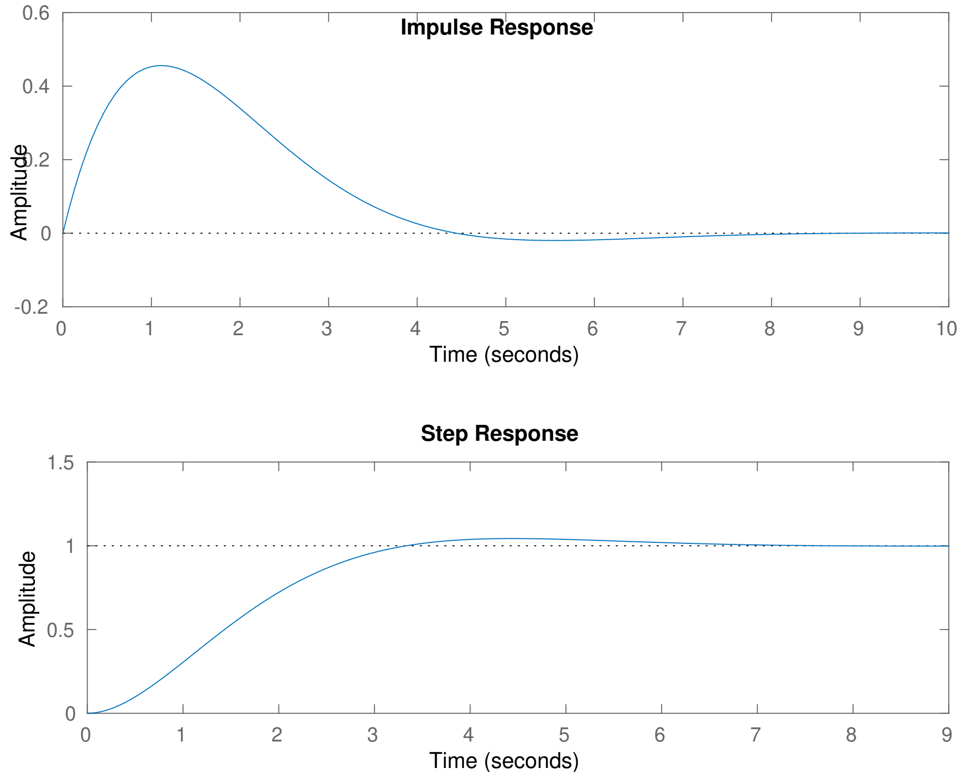


Figure D.7: The impulse and step responses of the system obtained from the code example.

Example D.31 (Butterworth lowpass filter design)**.** Suppose that we want to design a tenth-order Butterworth lowpass filter with a cutoff frequency of 100 rad/s. We can design such a filter as well as plot its frequency response using the code given in Listing D.7 below. The frequency response of the filter obtained from this code is shown in Figure D.8.

Listing D.7: Butterworth lowpass filter design

```
% Calculate the transfer function coefficients for a tenth-order Butterworth
% lowpass filter with a cutoff frequency of 100 rad/s.
[tf_num, tf_denom] = butter(10, 100, 's');

% Plot the frequency response of the filter.
freqs(tf_num, tf_denom);
```

■

Example D.32 (Bessel lowpass filter design)**.** Suppose that we want to design a tenth-order Bessel lowpass filter with a cutoff frequency of 100 rad/s. We can design such a filter as well as plot its frequency response using the code given in Listing D.8 below. The frequency response of the filter obtained from this code is shown in Figure D.9.

Listing D.8: Bessel lowpass filter design

```
% Calculate the transfer function coefficients for a tenth-order Bessel
% lowpass filter with a cutoff frequency of 100 rad/s.
[tf_num, tf_denom] = besself(10, 100);

% Plot the frequency response of the filter.
freqs(tf_num, tf_denom);
```

■

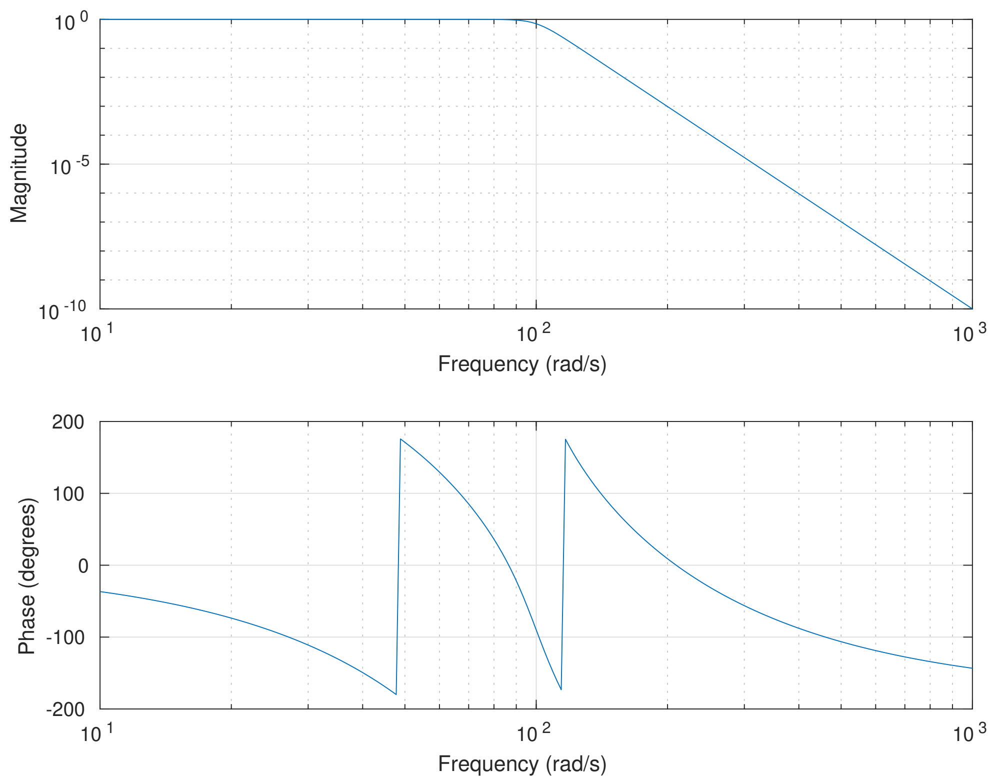


Figure D.8: The frequency response of the Butterworth lowpass filter obtained from the code example.

D.15.2 Discrete-Time Signal Processing

In the sections that follow, we introduce some of the functionality in MATLAB that is useful for discrete-time signals and systems.

Most discrete-time LTI systems of practical interest are causal with rational transfer functions. For this reason, MATLAB has considerable functionality for working with such systems. Consider the rational transfer function

$$H(z) = \frac{\sum_{k=1}^{n} b_k z^{-(k-1)}}{\sum_{k=1}^{m} a_k z^{-(k-1)}} = \frac{b_1 + b_2 z^{-1} + \ldots + b_n z^{-(n-1)}}{a_1 + a_2 z^{-1} + \ldots + a_m z^{-(m-1)}}. \tag{D.2}$$

Typically, MATLAB represents such a transfer function using two vectors of coefficients, one for the $\{b_k\}$ and one for the $\{a_k\}$.

D.15.2.1 Frequency Responses

For a LTI system with a transfer function of the form of (D.2), the `freqz` function can be used to compute and optionally plot the frequency response of the system.

Example D.33 (Computing and plotting frequency responses with freqz)**.** Consider the LTI system with transfer function

$$H(z) = \frac{-0.2037z^4 + 0.5925z^2 - 0.2037}{z^4} = -0.2037 + 0.5925z^{-2} - 0.2037z^{-4}.$$

Suppose that we would like to evaluate the frequency response of this system. This is equivalent to evaluating the transfer function H at points on the unit circle. To this end, we can employ the `freqz` function in MATLAB. More specifically, we can calculate and plot the magnitude and phase responses of the above system with the following code:

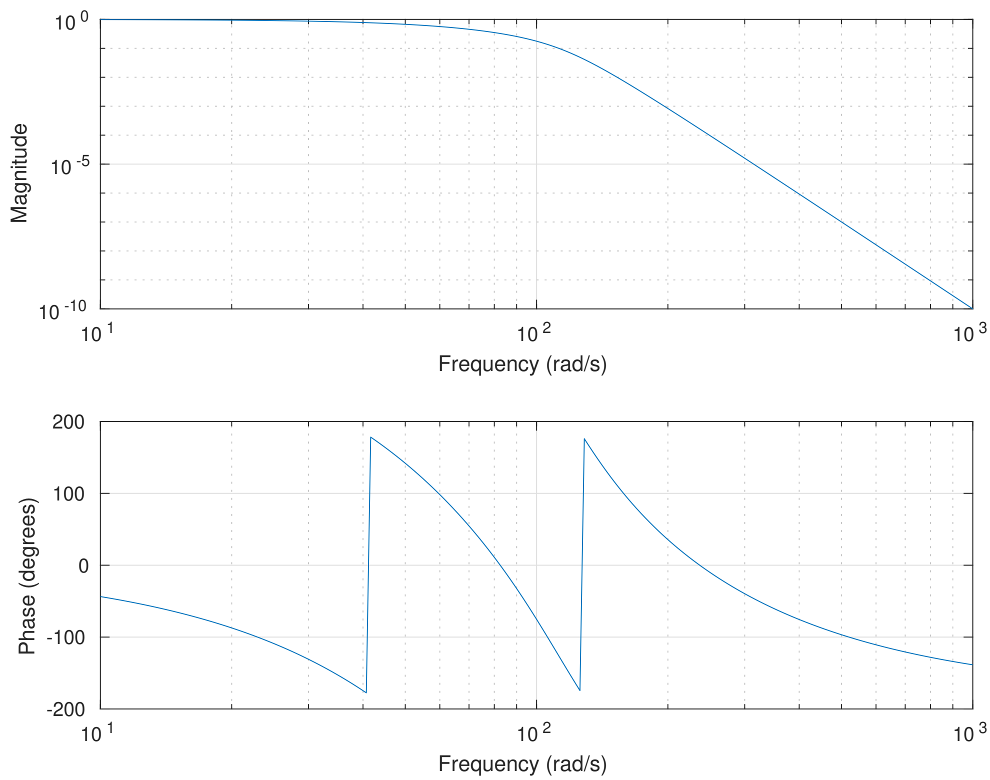


Figure D.9: The frequency response of the Bessel lowpass filter obtained from the code example.

```
% Initialize the numerator and denominator coefficients of the transfer
% function.
tf_num = [
  -0.2037
   0.0000
   0.5925
   0.0000
  -0.2037
];
tf_denom = [1];

% Plot the magnitude and phase responses.
freqz(tf_num, tf_denom);
```

The plot produced by the `freqz` function is shown in Figure D.10. ■

Example D.34 (Plotting frequency responses)**.** Suppose that we would like to have a function that behaves in a similar way to the MATLAB `freqz` function, but with a few differences in how plotting is performed. In particular, we would like the plots generated with the magnitude response as a unitless quantity (not in decibels) and the phase response in unwrapped form. This can be accomplished with the code given in Listing D.9.

Listing D.9: `myfreqz.m`

```
function [freq_resp, omega] = myfreqz(tf_num, tf_denom, omega)
    % The myfreqz function has essentially the same interface as the
    % MATLAB freqz function, but performs plotting slightly differently.
    % The magnitude response is plotted as a unitless quantity (not in
```

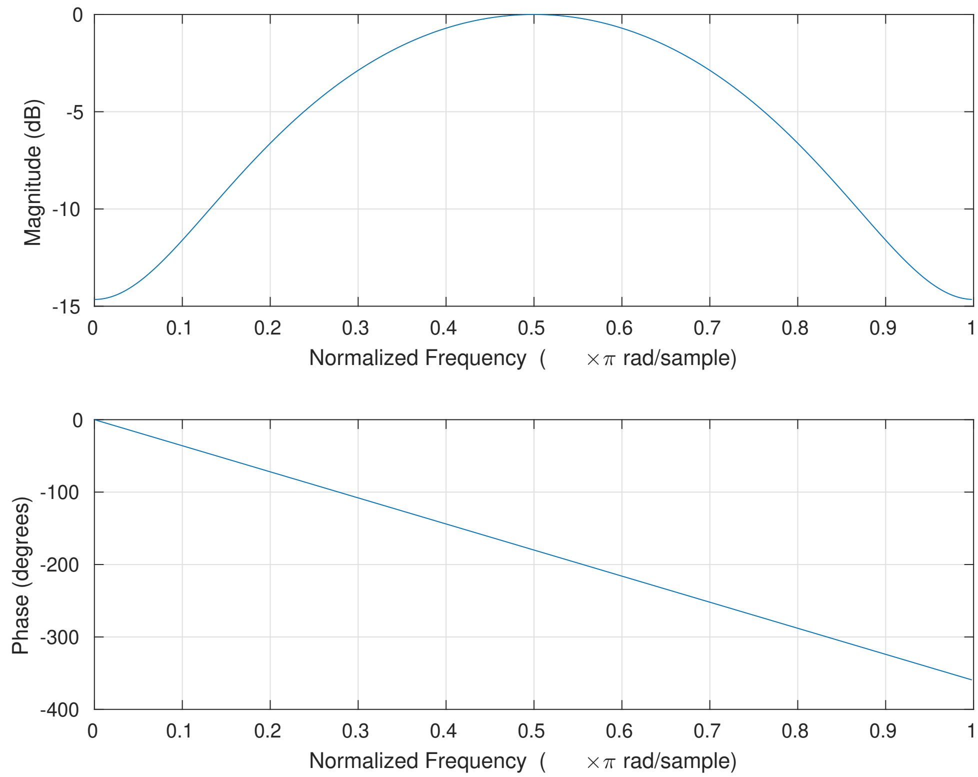


Figure D.10: The frequency response of the filter as produced by the `freqz` function.

```
% decibels).
% The phase response is plotted with the phase unwrapped.

% If the frequencies have been specified as an input argument, then simply
% pass them through to the real freqz function.
if nargin >= 3
    [freq_resp, omega] = freqz(tf_num, tf_denom, omega);
else
    [freq_resp, omega] = freqz(tf_num, tf_denom);
end

% If no output arguments were specified, plot the frequency response.
if nargout == 0

    % Compute the magnitude response as a unitless quantity.
    mag_resp = abs(freq_resp);

    % Compute the phase response with the phase unwrapped.
    phase_resp = unwrap(angle(freq_resp)) * 180 / pi;

    % On the first of two graphs, plot the magnitude response.
    subplot(2, 1, 1);
    plot(omega / pi, mag_resp);
    title('Magnitude Response');
```

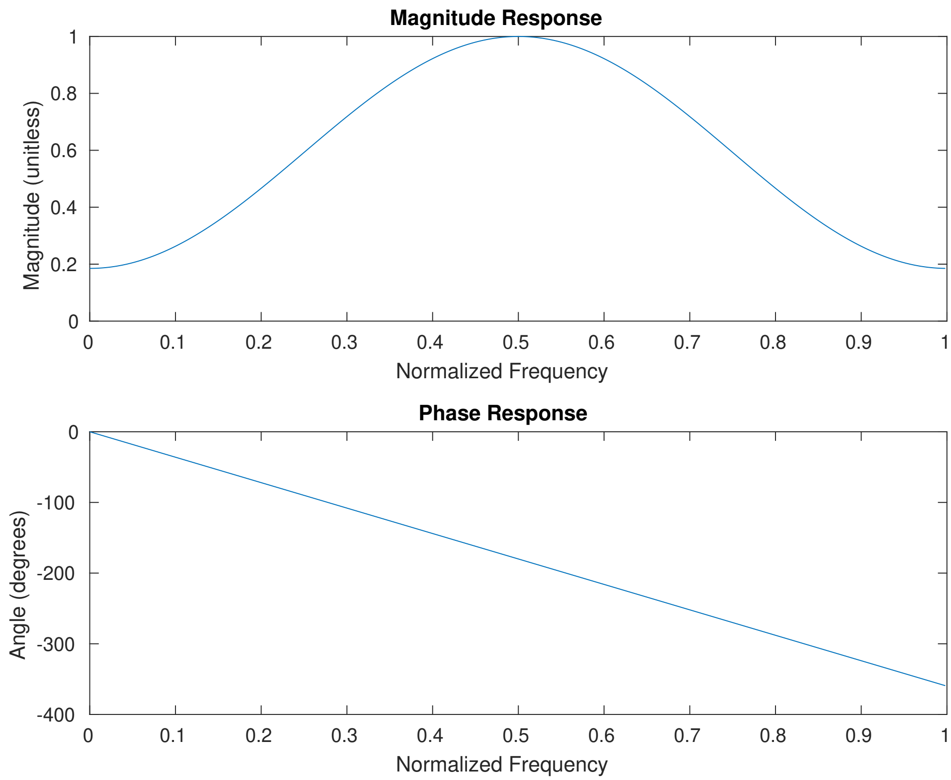


Figure D.11: The frequency response of the filter as produced by the `myfreqz` function.

```
        xlabel('Normalized Frequency');
        ylabel('Magnitude (unitless)');

        % On the second of two graphs, plot the phase response.
        subplot(2, 1, 2);
        plot(omega / pi, phase_resp);
        title('Phase Response');
        xlabel('Normalized Frequency');
        ylabel('Angle (degrees)');

    end
end
```

For the filter in Example D.33, the `myfreqz` function produces the frequency-response plots shown in Figure D.11.
■

D.15.2.2 Impulse and Step Responses

Sometimes, we need to determine the response of a LTI system to a specific input. Two inputs of particular interest are the unit-impulse sequence δ and unit-step sequence u. Fortunately, it is quite easy to compute impulse and step responses using the `impz` and `stepz` functions in MATLAB, as illustrated by the example below.

Example D.35 (Computing impulse and step responses)**.** Consider the LTI system with the transfer function

$$H(z) = \frac{0.571 - 0.591z^{-1} + 0.503z^{-2} + 0.503z^{-3} - 0.591z^{-4} + 0.571z^{-5}}{1.000 - 3.588z^{-1} + 5.303z^{-2} - 4.003z^{-3} + 1.538z^{-4} - 0.239z^{-5}}.$$

Suppose that we wish to calculate and plot the impulse and step responses of this system. This can be accomplished with the code given in Listing D.10. Executing this code produces the plots shown in Figure D.12.

Listing D.10: Computing and plotting the impulse and step responses

```
% Initialize the numerator and denominator coefficients of the transfer
% function.
tf_num = [
   0.571
  -0.591
   0.503
   0.503
  -0.591
   0.571
];
tf_denom = [
   1.000
  -3.588
   5.303
  -4.003
   1.538
  -0.239
];

% Plot the impulse response.
subplot(2, 1, 1);
impz(tf_num, tf_denom);

% Plot the step response.
subplot(2, 1, 2);
stepz(tf_num, tf_denom);
```

■

D.15.2.3 Filter Design

A number of functions are provided in MATLAB to assist in the design of various types of discrete-time filters. In what follows, we consider a few examples of using such functions.

Example D.36 (Chebyshev type-II lowpass filter design)**.** Suppose that the sampling rate is 2000 rad/s, and we want to design a fifth-order Chebyshev type-II lowpass filter with a cutoff frequency of 250 rad/s and stopband ripple of 20 dB. The cutoff frequency in normalized units is $250/(\frac{2000}{2}) = 0.25$. We can design the desired filter as well as plot its frequency response using the code given in Listing D.11 below. The frequency response of the filter obtained from this code is shown in Figure D.13.

Listing D.11: Chebyshev type-II lowpass filter design

```
% Calculate the transfer function coefficients for a
% fifth-order Chebyshev type-II lowpass filter with a
% (normalized) cutoff frequency of 0.25 and a
% stopband ripple of 20 dB.
[tf_num, tf_denom] = cheby2(5, 20, 0.25, 'low');

% Plot the frequency response of the filter.
freqz(tf_num, tf_denom);
```

■

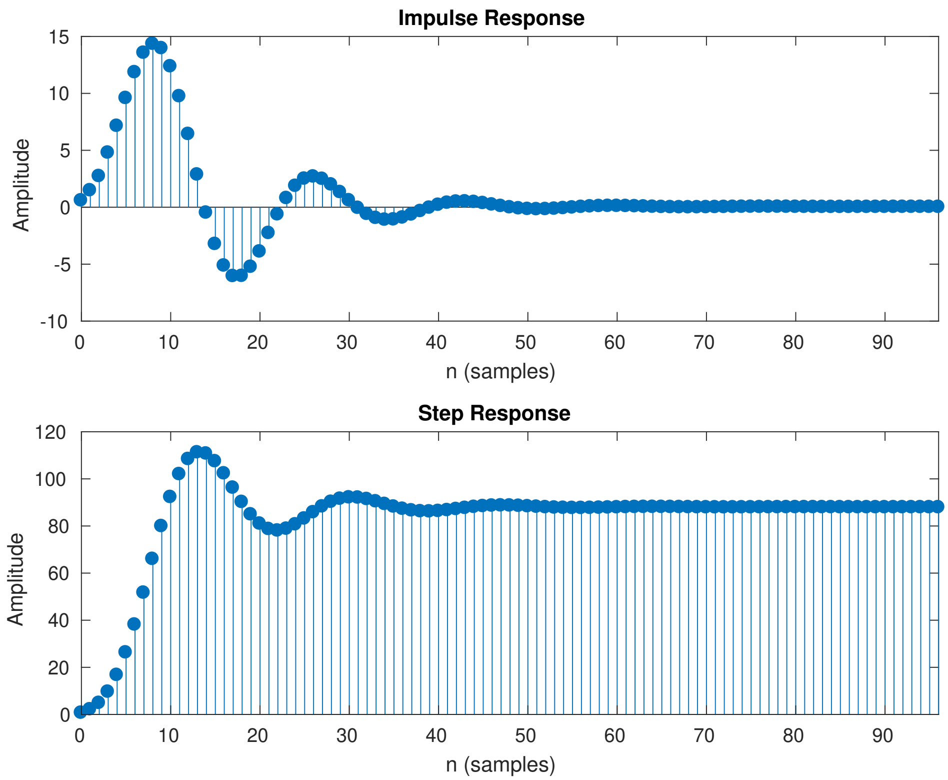


Figure D.12: The impulse and step responses of the system obtained from the code example.

Example D.37 (Linear-phase FIR bandpass filter design)**.** Suppose that the sampling rate is 2000 rad/s, and we want to design an order-64 linear-phase FIR bandpass filter with cutoff frequencies of 250 and 750 rad/s using the `fir1` function in MATLAB. The cutoff frequencies in normalized units are $250/(\frac{2000}{2}) = 0.25$ and $750/(\frac{2000}{2}) = 0.75$. We can design the desired filter as well as plot its frequency response using the code given in Listing D.12 below. The frequency response of the filter obtained from this code is shown in Figure D.14.

Listing D.12: Linear-phase FIR bandpass filter design

```
% Calculate the transfer function coefficients for a
% 64th-order FIR bandpass filter with
% (normalized) cutoff frequencies of 0.25 and 0.75.
tf_num = fir1(64, [0.25 0.75], 'bandpass');
tf_denom = [1];

% Plot the frequency response of the filter.
freqz(tf_num, tf_denom);
```

■

D.16 Miscellany

Some other functions that might be useful are listed in Table D.29.

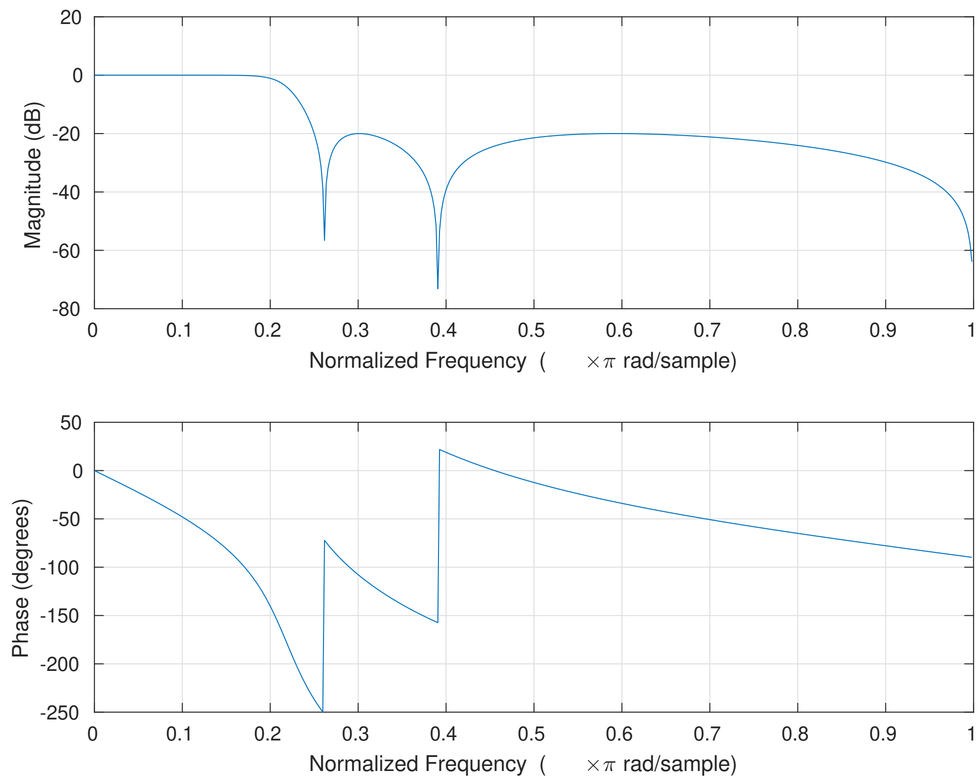


Figure D.13: The frequency response of the Chebyshev type-II lowpass filter obtained from the code example.

Table D.29: Miscellaneous functions/commands

Name	Description
`roots`	find roots of polynomial
`clear`	clear a variable
`diary`	log MATLAB session
`echo`	echo commands on execution (for debugging)
`quit`	quit MATLAB
`format`	output format for numbers

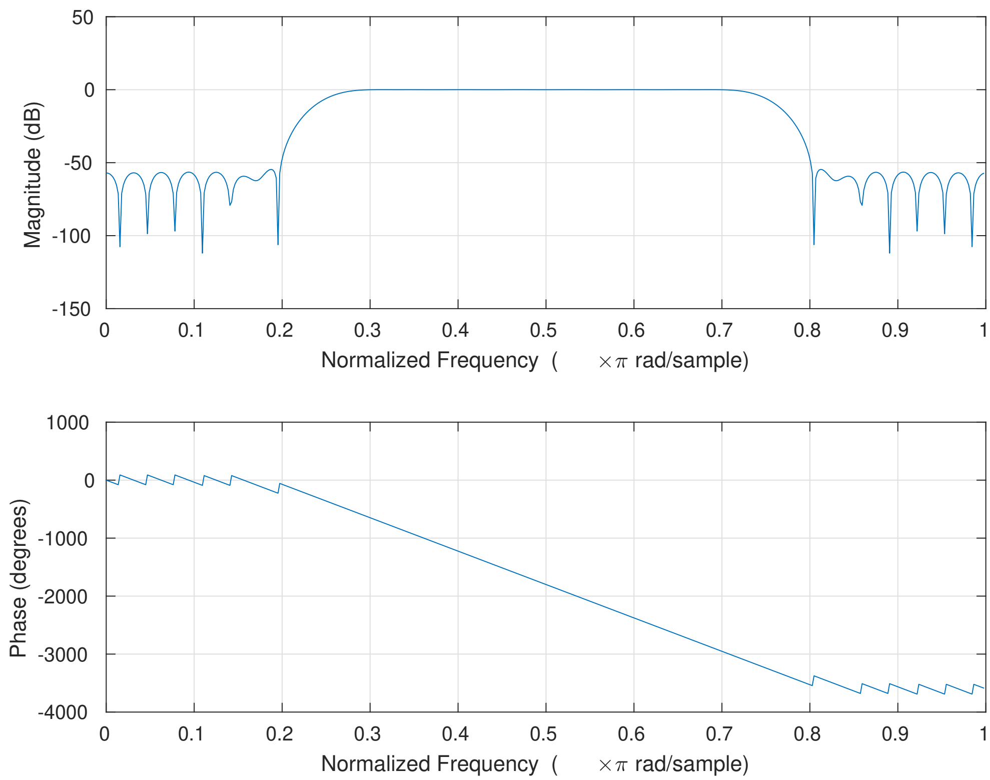


Figure D.14: The frequency response of the linear-phase FIR bandpass filter obtained from the code example.

D.17 Exercises

D.1 Indicate whether each of the following is a valid MATLAB identifier (i.e., variable/function name):
(a) `4ever`
(b) `$rich$`
(c) `foobar`
(d) `foo_bar`
(e) `_foobar`

D.2 Consider the vector `v` defined by the following line of code:

```
v = [0 1 2 3 4 5]
```

Write an expression in terms of `v` that yields a new vector of the same dimensions as `v`, where each element t of the original vector `v` has been replaced by the given quantity below. In each case, the expression should be as short as possible.
(a) $2t-3$;
(b) $1/(t+1)$;
(c) t^5-3; and
(d) $|t|+t^4$.

D.3 Let T_C, T_F, and T_K denote the temperature measured in units of Celsius, Fahrenheit, and Kelvin, respectively. Then, these quantities are related by

$$T_F = \tfrac{9}{5}T_C + 32 \quad \text{and}$$
$$T_K = T_C + 273.15.$$

Write a program that generates a temperature conversion table. The first column of the table should contain the temperature in Celsius. The second and third columns should contain the corresponding temperatures in units of Fahrenheit and Kelvin, respectively. The table should have entries for temperatures in Celsius from -50 to 50 in steps of 10.

D.4 (a) Write a function called `unitstep` that takes a single real argument t and returns $u(t)$, where

$$u(t) = \begin{cases} 1 & t \geq 0 \\ 0 & \text{otherwise.} \end{cases}$$

(b) Modify the function from part (a) so that it takes a single vector argument $t = \begin{bmatrix} t_1 & t_2 & \cdots & t_n \end{bmatrix}^T$ (where $n \geq 1$ and $t_1, t_2, \ldots, t_n$ are real) and returns the vector $\begin{bmatrix} u(t_1) & u(t_2) & \cdots & u(t_n) \end{bmatrix}^T$. Your solution must employ a looping construct (e.g., a for loop).
(c) With some ingenuity, part (b) of this exercise can be solved using only two lines of code, without the need for any looping construct. Find such a solution. [Hint: In MATLAB, to what value does an expression like "`[-2 -1 0 1 2] >= 0`" evaluate?]

D.5 Let F denote the complex-valued function of a real variable given by

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

Write a program to plot $|F(\omega)|$ and $\arg F(\omega)$ for ω in the interval $[-10, 10]$. Use `subplot` to place both plots on the same figure.

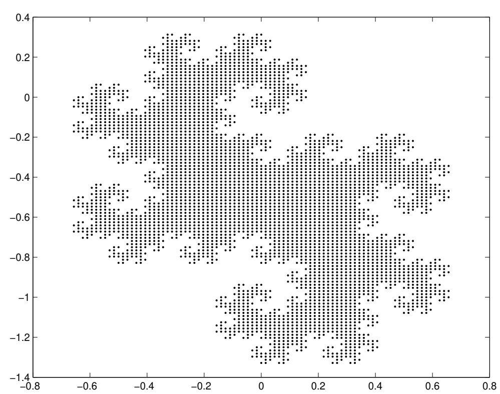
D.6 In what follows, we consider a simple algorithm for generating a set of 2^d points in the complex plane, where d is a positive integer. For $n = 0, 1, \ldots, 2^d - 1$, this algorithm computes the nth point p_n as

$$p_n = \sum_{k=0}^{d-1} a_k \beta^{k+1},$$

where $\beta = \frac{1}{\sqrt{2}} e^{-j\pi/4}$, $a_k \in \{0, 1\}$, and $n = \sum_{k=0}^{d-1} a_k 2^k$. Note that the binary sequence $a_{d-1}, \ldots, a_1, a_0$ is simply the d-bit binary representation of the integer n, where a_0 corresponds to the least-significant bit. For example, if $d = 3$, then the relationship between n and a_2, a_1, a_0 is as shown in the following table:

n	a_2	a_1	a_0
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

(a) Write a function called `twindragon` that calculates and plots the point set obtained by using the above algorithm for a specified value of the parameter d. The value of d should be passed as an input argument to the function. [Hint: It is possible to implement this function in about 15 to 20 lines of code. The `polyval` and `dec2bin` functions should be quite helpful.]
(b) Using the function developed in part (a), plot the set of points obtained with the parameter d equal to 12. In the limit as d approaches infinity, the resulting set of points converges to the well-known twin-dragon fractal set. Although choosing $d = 12$ should be sufficient to see the approximate shape of the point set in the limiting case, larger values of d will yield better approximations, albeit at the expense of significantly more computation. You should obtain a plot resembling that shown below.



D.7 In what follows, let $\min(a, b)$ denote the minimum of a and b. For a complex number z, we define an iterative process that generates the complex-number sequence $v_z(0), v_z(1), v_z(2), \ldots$, where

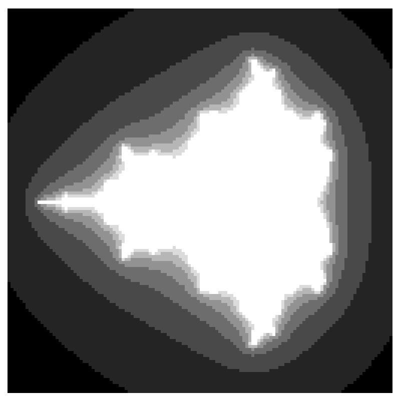
$$v_z(n) = \begin{cases} v_z^2(n-1) + z & n \geq 1 \\ 0 & n = 0. \end{cases}$$

Let $g(z)$ denote the smallest value of n for which $|v_z(n)| > 10$. In the case that $|v_z(n)| > 10$ is not satisfied for any n, $g(z)$ is simply defined to be ∞. Let $f(z) = \min(g(z), 10)$. For example, we have the following:

z	First few elements of v_z	g	f
0	$0,0,0,0$	∞	10
j	$0, j, -1+j, -j, -1+j, -j$	∞	10
2	$0,2,6,38$	3	3
$2j$	$0, 2j, -4+2j, 12-14j$	3	3

Write a function called `mandelbrotfunc` to compute the value $f(z)$ for any (complex) value of z. Using the function `mandelbrotfunc`, evaluate f on an evenly-spaced rectangular grid (in the complex plane) having width 128, height 128, and with the bottom-left and top-right corners of the grid corresponding to the points $-2.25 - 1.5j$ and $1 + 1.5j$, respectively. Store these computed values into a 128×128 matrix. Then, using the `pcolor` function in MATLAB, plot the contents of this matrix. After the call to `pcolor`, use the command "`shading interp`" to eliminate the grid lines from the plot (which causes the plot to look somewhat less attractive).

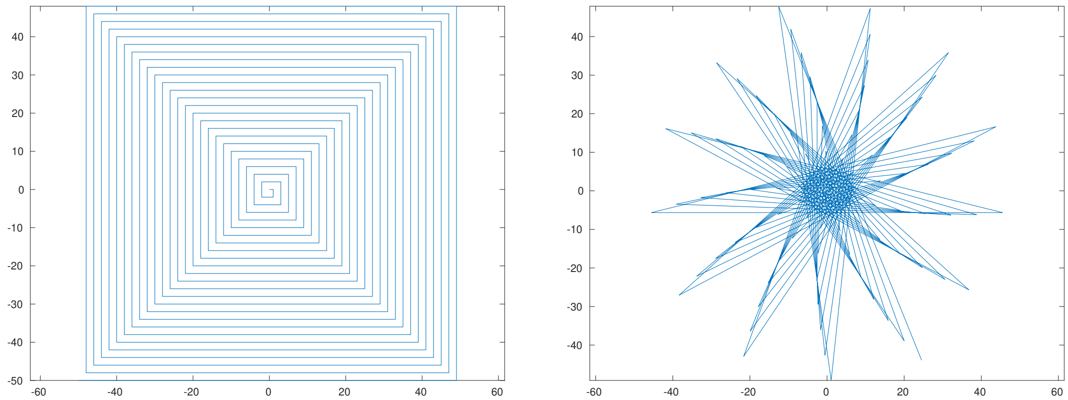
A complete solution to this exercise requires less than 25 lines of code (excluding comments). A correct solution should yield a plot resembling the one shown below. Incidentally, the innermost region in the plot is an approximation to the famous Mandelbrot (fractal) set.



D.8 In this exercise, we consider an algorithm for generating a sequence p of n points in the plane (i.e., $p_0, p_1, \ldots, p_{n-1}$). The first point p_0 is chosen as the origin (i.e., $p_0 = [\,0\ 0\,]^T$), with the remaining points being given by the formula

$$p_i = p_{i-1} + \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}^{i-1} \begin{bmatrix} i \\ 0 \end{bmatrix}.$$

(a) Using MATLAB, write a function called `drawpattern` that takes n and θ as input arguments (in that order) with θ being specified in degrees, and then computes and plots the points $p_0, p_1, \ldots, p_{n-1}$ connected by straight lines (i.e., draw a line from p_0 to p_1, p_1 to p_2, p_2 to p_3, and so on). When performing the plotting, be sure to use `axis('equal')` in order to maintain the correct aspect ratio for the plot. For illustrative purposes, the plots produced for two sets of θ and n values are shown in Figures (a) and (b) below.
(b) Generate the plots obtained by invoking `drawpattern` with $n = 100$ and θ set to each of the following values: 89°, 144°, and 154°. [Note: In MATLAB, the `sin` and `cos` functions take values in radians, not degrees.]



(a) $\theta = 90^\circ$ and $n = 100$ (b) $\theta = 166^\circ$ and $n = 100$

Appendix E

Additional Exercises

E.1 Overview

This appendix contains numerous additional exercises, which may be useful for practice purposes. These exercises have been deliberately left uncategorized, so that the reader can gain practice in identifying the general approach required to solve a problem. In the case of each exercise for which a short answer is possible, an answer key is provided.

E.2 Continuous-Time Signals and Systems

E.1 A communication channel heavily distorts high frequencies but does not significantly affect very low frequencies. Determine which of the functions x_1, x_2, x_3, and x_4 would be least distorted by the communication channel, where

$$x_1(t) = \delta(t), \quad x_2(t) = 5, \quad x_3(t) = 10e^{j1000t}, \quad \text{and} \quad x_4(t) = 1/t.$$

Short Answer. x_2

E.2 Consider a system consisting of a communication channel with input x and output y. Since the channel is not ideal, y is typically a distorted version of x. Suppose that the channel can be modelled as a causal LTI system with impulse response $h(t) = e^{-t}u(t) + \delta(t)$. Determine whether we can devise a physically-realizable BIBO-stable system that recovers x from y. If such a system exists, find its impulse response g.

Short Answer. $g(t) = \delta(t) - e^{-2t}u(t)$

E.3 A causal LTI system has impulse response h, system function H, and the following characteristics: 1) H is rational with one pole at -2 and no zeros; and 2) $h(0^+) = 4$. Find h.

Short Answer. $h(t) = 4e^{-2t}u(t)$

E.4 A causal LTI system with input x and output y is characterized by the differential equation

$$y'(t) + 3y(t) = 2x(t),$$

where the prime symbol denotes derivative. Find the impulse response h of the system.

Short Answer. $h(t) = 2e^{-3t}u(t)$

E.5 A causal LTI system $\mathcal{H}$ is such that $y = \mathcal{H}x$, where $x(t) = e^{-2t}u(t)$ and $y(t) = e^{-3t}u(t)$. Find the unit-step response s of the system.

Short Answer. $s(t) = \frac{2}{3}u(t) + \frac{1}{3}e^{-3t}u(t)$

E.6 A causal BIBO-stable LTI system $\mathcal{H}$ has impulse response h and system function H. The function H is rational, contains a pole at -2 and does not have a zero at the origin. The other poles and zeros of H are unknown. Determine whether each of the statements below is true, false, or uncertain (i.e., insufficient information to determine).
(a) $\mathcal{F}\{e^{3t}h(t)\}$ converges;
(b) $\int_{-\infty}^{\infty} h(t)dt = 0$;
(c) $th(t)$ is the impulse response of a causal and BIBO-stable system;
(d) $\mathcal{D}h$ has at least one pole in its Laplace transform expression, where $\mathcal{D}$ denotes the derivative operator.

Short Answer. (a) false; (b) false; (c) true; (d) true

E.7 A communication channel can be well approximated by a LTI system with impulse response $h(t) = \frac{1000}{\pi}\operatorname{sinc}(1000t)$. Determine which of the functions x_1, x_2, and x_3 would be least distorted by this channel, where

$$x_1(t) = \delta(t), \quad x_2(t) = u(t), \quad \text{and} \quad x_3(t) = \cos(100t).$$

Short Answer. x_3

E.8 A common problem in real-world instrumentation systems is electromagnetic interference caused by 60 Hz power lines. In particular, 60 Hz power lines can often introduce a significant amount of interference (i.e., noise) at 60 Hz and its higher harmonics (i.e., 120 Hz, 180 Hz, 240 Hz, and so on). Consider a causal system with an impulse response of the form $h(t) = a[u(t) - u(t-b)]$, where a and b are nonzero real constants. With an appropriate choice of a and b, such a system can be made to reject interference at 60 Hz and all of its higher harmonics. Find a and b.

Short Answer. $a \neq 0$ and $b = \frac{1}{60}$

E.9 For the causal LTI system with input x, output y, and impulse response h that is characterized by each differential equation given below, find h. [Note: The prime symbol denotes derivative.]
(a) $4y''(t) = 2x(t) - x'(t)$.

Short Answer. (a) $(\frac{1}{2}t - \frac{1}{4})u(t)$

E.3 Discrete-Time Signals and Systems

No additional exercises for discrete-time signals and systems are currently available.

Appendix F

Miscellaneous Information

F.1 Overview

This appendix contains numerous mathematical formulas and tables that are likely to be helpful in relation to the material covered in this book.

F.2 Combinatorial Formulas

For a nonnegative integer n, the **factorial** of n, denoted $n!$, is defined as

$$n! = \begin{cases} n(n-1)(n-2)\cdots(1) & n \geq 1 \\ 1 & n = 0. \end{cases}$$

For two integers n and k such that $0 \leq k \leq n$, the $\binom{n}{k}$ **binomial coefficient** is defined as

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}.$$

F.3 Derivatives

Some basic derivatives include:

$$\begin{aligned} \tfrac{d}{dx}x^n &= nx^{n-1} \quad \text{for } n \in \mathbb{Z}; \\ \tfrac{d}{dx}\cos x &= -\sin x; \\ \tfrac{d}{dx}\sin x &= \cos x; \quad \text{and} \\ \tfrac{d}{dx}e^x &= e^x. \end{aligned}$$

Some formulas for the derivatives of products and quotients are as follows:

$$\begin{aligned} \tfrac{d}{dx}(uv) &= u\tfrac{dv}{dx} + v\tfrac{du}{dx}; \quad \text{and} \\ \tfrac{d}{dx}\left(\frac{u}{v}\right) &= \frac{v\frac{du}{dx} - u\frac{dv}{dx}}{v^2}. \end{aligned}$$

Some additional derivatives include:

$$\tfrac{d}{dx}\operatorname{sinc}(x) = \frac{\cos(x) - \operatorname{sinc}(x)}{x}.$$

F.4 Integrals

Some basic integrals include:

$$\int x^n dx = \begin{cases} \frac{1}{n+1}x^{n+1} + C & n \neq -1 \\ \ln|x| + C & n = -1; \end{cases}$$
$$\int \cos x dx = \sin x + C;$$
$$\int \sin x dx = -\cos x + C; \quad \text{and}$$
$$\int e^x dx = e^x + C.$$

Some additional integrals include:

$$\int x e^{ax} dx = \tfrac{1}{a^2} e^{ax}(ax-1) + C \quad \text{for } a \neq 0; \tag{F.1}$$
$$\int x^2 e^{ax} dx = \tfrac{1}{a^3} e^{ax}(a^2x^2 - 2ax + 2) + C \quad \text{for } a \neq 0; \tag{F.2}$$
$$\int e^{ax}\cos(bx)dx = \frac{e^{ax}[a\cos(bx) + b\sin(bx)]}{a^2+b^2} + C \quad \text{for } a \neq \pm jb; \tag{F.3}$$
$$\int e^{ax}\sin(bx)dx = \frac{e^{ax}[a\sin(bx) - b\cos(bx)]}{a^2+b^2} + C \quad \text{for } a \neq \pm jb; \tag{F.4}$$
$$\int x\cos(ax)dx = \frac{ax\sin(ax) + \cos(ax)}{a^2} + C \quad \text{for } a \neq 0; \quad \text{and} \tag{F.5}$$
$$\int x\sin(ax)dx = \frac{\sin(ax) - ax\cos(ax)}{a^2} + C \quad \text{for } a \neq 0. \tag{F.6}$$

The formula for integration by parts is as follows:

$$\int u dv = uv - \int v du.$$

F.5 Arithmetic and Geometric Sequences

The sum of the arithmetic sequence $a, a+d, a+2d, \ldots, a+(n-1)d$ is given by

$$\sum_{k=0}^{n-1}(a+kd) = \frac{n[2a+d(n-1)]}{2}. \tag{F.7}$$

The sum of the geometric sequence $a, ra, r^2a, \ldots, r^{n-1}a$ is given by

$$\sum_{k=0}^{n-1} r^k a = a\frac{r^n - 1}{r-1} \quad \text{for } r \neq 1. \tag{F.8}$$

The sum of the infinite geometric sequence $a, ra, r^2a, \ldots$ is given by

$$\sum_{k=0}^{\infty} r^k a = \frac{a}{1-r} \quad \text{for } |r| < 1. \tag{F.9}$$

F.6 Taylor/Maclaurin Series

Some series for trigonometric functions are as follows:

$$\sin x = \sum_{n=0}^{\infty} \frac{(-1)^n}{(2n+1)!} x^{2n+1} \quad \text{for all } x \in \mathbb{C}; \quad \text{and} \tag{F.10}$$

$$\cos x = \sum_{n=0}^{\infty} \frac{(-1)^n}{(2n)!} x^{2n} \quad \text{for all } x \in \mathbb{C}. \tag{F.11}$$

Some series for exponential and logarithmic functions are as follows:

$$e^x = \sum_{n=0}^{\infty} \frac{x^n}{n!} \quad \text{for all } x \in \mathbb{C}; \tag{F.12}$$

$$\ln(1-x) = -\sum_{n=1}^{\infty} \frac{x^n}{n} \quad \text{for all } x \in \mathbb{C} \text{ satisfying } |x| < 1 \text{ or } x = -1; \quad \text{and} \tag{F.13}$$

$$\ln(1+x) = \sum_{n=1}^{\infty} (-1)^{n+1} \frac{x^n}{n} \quad \text{for all } x \in \mathbb{C} \text{ satisfying } |x| < 1 \text{ or } x = 1.$$

Some series for hyperbolic functions are as follows:

$$\sinh x = \sum_{n=0}^{\infty} \frac{x^{2n+1}}{(2n+1)!} \quad \text{for all } x \in \mathbb{C}; \quad \text{and}$$

$$\cosh x = \sum_{n=0}^{\infty} \frac{x^{2n}}{(2n)!} \quad \text{for all } x \in \mathbb{C}.$$

F.7 Other Formulas for Sums

Formulas for various sums are as follows:

$$\sum_{k=1}^{n} kx^k = \begin{cases} \dfrac{nx^{n+2} - (n+1)x^{n+1} + x}{(x-1)^2} & x \in \mathbb{C}, x \neq 1 \\ \dfrac{n(n+1)}{2} & x = 1; \end{cases} \tag{F.14}$$

$$\sum_{k=1}^{\infty} kx^k = \frac{x}{(x-1)^2} \quad \text{for all } x \in \mathbb{C} \text{ satisfying } |x| < 1; \tag{F.15}$$

$$\sum_{k=1}^{n} k^2 x^k = \begin{cases} \dfrac{n^2 x^{n+3} - (2n^2 + 2n - 1)x^{n+2} + (n^2 + 2n + 1)x^{n+1} - x(x+1)}{(x-1)^3} & x \in \mathbb{C}, x \neq 1 \\ \dfrac{n\,(2n+1)\,(n+1)}{6} & x = 1; \end{cases} \quad \text{and}$$

$$\sum_{k=1}^{\infty} k^2 x^k = \frac{x(1+x)}{(1-x)^3} \quad \text{for all } x \in \mathbb{C} \text{ satisfying } |x| < 1.$$

F.8 Trigonometric Identities

A key Pythagorean identity is as follows:

$$\sin^2(\theta) + \cos^2(\theta) = 1.$$

Some angle-sum and angle-difference identities are as follows:

$$\begin{aligned}
\sin(a+b) &= \sin(a)\cos(b)+\cos(a)\sin(b); \\
\sin(a-b) &= \sin(a)\cos(b)-\cos(a)\sin(b); \\
\cos(a+b) &= \cos(a)\cos(b)-\sin(a)\sin(b); \\
\cos(a-b) &= \cos(a)\cos(b)+\sin(a)\sin(b); \\
\tan(a+b) &= \frac{\tan(a)+\tan(b)}{1-\tan(a)\tan(b)}; \quad \text{and} \\
\tan(a-b) &= \frac{\tan(a)-\tan(b)}{1+\tan(a)\tan(b)}.
\end{aligned}$$

Some double-angle identities are as follows:

$$\begin{aligned}
\sin(2\theta) &= 2\sin(\theta)\cos(\theta); \\
\cos(2\theta) &= \cos^2(\theta)-\sin^2(\theta); \quad \text{and} \\
\tan(2\theta) &= \frac{2\tan(\theta)}{1-\tan^2(\theta)}.
\end{aligned}$$

Some product-to-sum identities are as follows:

$$\begin{aligned}
\sin(a)\sin(b) &= \tfrac{1}{2}[-\cos(a+b)+\cos(a-b)]; \\
\sin(a)\cos(b) &= \tfrac{1}{2}[\sin(a+b)+\sin(a-b)]; \quad \text{and} \\
\cos(a)\cos(b) &= \tfrac{1}{2}[\cos(a+b)+\cos(a-b)].
\end{aligned}$$

Some sum-to-product identities are as follows:

$$\begin{aligned}
\sin(a)+\sin(b) &= 2\sin\left[\tfrac{1}{2}(a+b)\right]\cos\left[\tfrac{1}{2}(a-b)\right]; \\
\sin(a)-\sin(b) &= 2\cos\left[\tfrac{1}{2}(a+b)\right]\sin\left[\tfrac{1}{2}(a-b)\right]; \\
\cos(a)+\cos(b) &= 2\cos\left[\tfrac{1}{2}(a+b)\right]\cos\left[\tfrac{1}{2}(a-b)\right]; \quad \text{and} \\
\cos(a)-\cos(b) &= -2\sin\left[\tfrac{1}{2}(a+b)\right]\sin\left[\tfrac{1}{2}(a-b)\right].
\end{aligned}$$

F.9 Exact Trigonometric Function Values

The exact values of various trigonometric functions for certain special angles can be found in Table F.1.

F.10 Miscellany

An important property that holds for both real and complex numbers is the **triangle inequality**:

$$|a+b| \leq |a|+|b| \quad \text{for all } a,b \in \mathbb{C}. \tag{F.16}$$

Table F.1: Exact values of various trigonometric functions for certain special angles

θ				
Degrees	Radians	$\sin\theta$	$\cos\theta$	$\tan\theta$
0	0	0	1	0
15	$\frac{\pi}{12}$	$\frac{\sqrt{6}-\sqrt{2}}{4}$	$\frac{\sqrt{6}+\sqrt{2}}{4}$	$2-\sqrt{3}$
30	$\frac{\pi}{6}$	$\frac{1}{2}$	$\frac{\sqrt{3}}{2}$	$\frac{\sqrt{3}}{3}$
45	$\frac{\pi}{4}$	$\frac{\sqrt{2}}{2}$	$\frac{\sqrt{2}}{2}$	1
60	$\frac{\pi}{3}$	$\frac{\sqrt{3}}{2}$	$\frac{1}{2}$	$\sqrt{3}$
75	$\frac{3\pi}{4}$	$\frac{\sqrt{6}+\sqrt{2}}{4}$	$\frac{\sqrt{6}-\sqrt{2}}{4}$	$2+\sqrt{3}$
90	$\frac{\pi}{2}$	1	0	undefined
120	$\frac{2\pi}{3}$	$\frac{\sqrt{3}}{2}$	$-\frac{1}{2}$	$-\sqrt{3}$
135	$\frac{3\pi}{4}$	$\frac{\sqrt{2}}{2}$	$-\frac{\sqrt{2}}{2}$	-1
150	$\frac{5\pi}{6}$	$\frac{1}{2}$	$-\frac{\sqrt{3}}{2}$	$-\frac{\sqrt{3}}{3}$
180	π	0	-1	0
210 (-150)	$\frac{7\pi}{6}$ $\left(-\frac{5\pi}{6}\right)$	$-\frac{1}{2}$	$-\frac{\sqrt{3}}{2}$	$\frac{\sqrt{3}}{3}$
225 (-135)	$\frac{5\pi}{4}$ $\left(-\frac{3\pi}{4}\right)$	$-\frac{\sqrt{2}}{2}$	$-\frac{\sqrt{2}}{2}$	1
240 (-120)	$\frac{4\pi}{3}$ $\left(-\frac{2\pi}{3}\right)$	$-\frac{\sqrt{3}}{2}$	$-\frac{1}{2}$	$\sqrt{3}$
270 (-90)	$\frac{3\pi}{2}$ $\left(-\frac{\pi}{2}\right)$	-1	0	undefined
300 (-60)	$\frac{5\pi}{3}$ $\left(-\frac{\pi}{3}\right)$	$-\frac{\sqrt{3}}{2}$	$\frac{1}{2}$	$-\sqrt{3}$
315 (-45)	$\frac{7\pi}{4}$ $\left(-\frac{\pi}{4}\right)$	$-\frac{\sqrt{2}}{2}$	$\frac{\sqrt{2}}{2}$	-1
330 (-30)	$\frac{11\pi}{6}$ $\left(-\frac{\pi}{6}\right)$	$-\frac{1}{2}$	$\frac{\sqrt{3}}{2}$	$-\frac{\sqrt{3}}{3}$

Appendix G

Video Lectures

G.1 Introduction

The author has prepared video lectures for some of the material covered in this textbook. All of the videos are hosted by YouTube and available through the author's YouTube channel:

- https://www.youtube.com/iamcanadian1867

The most up-to-date information about this video-lecture content can be found at:

- https://www.ece.uvic.ca/~mdadams/sigsysbook/#video_lectures

For the convenience of the reader, some information on the video-lecture content available at the time of this writing is provided in the remainder of this appendix.

G.2 2020-05 ECE 260 Video Lectures

The author prepared video lectures for all of the continuous-time material covered in this textbook in order to teach the 2020-05 offering of the course ECE 260 (titled "Continuous-Time Signals and Systems") in the Department of Electrical and Computer Engineering at the University of Victoria, Victoria, Canada. All of these videos are available from the author's YouTube channel. Although these video lectures are based on a modified version of Edition 2.0 of the textbook and lecture slides, these video lectures were prepared so as to be independent of the textbook edition used. For this reason, these video lectures are likely to be an extremely valuable resource to the reader, regardless of the particular edition of the textbook that they are using. The video lectures for the above course can be found in the following YouTube playlist:

- https://www.youtube.com/playlist?list=PLbHYdvrWBMxYGMvQ3QG6paNu7CuIRL5dX

An information package for the video lectures is available that includes:

- a copy of the edition of the lecture slides used in the video lectures (in PDF format);
- a copy of all of the fully-annotated worked-through examples used in the video lectures (in PDF format); and
- a fully-cataloged list of the slides covered in the video lectures, where each slide in the list has a link to the corresponding time offset in the YouTube video where the slide is covered.

This information package is available from the video lecture section of the web site for the textbook:

- https://www.ece.uvic.ca/~mdadams/sigsysbook/#video_lectures

For the convenience of the reader, the catalog of the video lectures is also included in what follows.

