(j). Create a MATLAB function that takes as input a sequence `x` of arbitrary length and calculates N equally spaced frequency samples of its DTFT. The first line of your function should be

```
function X=dtft(x,N)
```

Be sure that your function works properly for both the case when $N \geq M$, and $N < M$. You should use exactly one call to the MATLAB function `fft` for each input `x`.

5.2 Telephone Touch-Tone

This exercise will teach you how the touch-tone system on the telephone uses signals of different frequencies to indicate which key has been pushed. The DTFT of a sampled telephone signal can be used to identify these frequencies. The sound you hear when you push a key is the sum of two sinusoids. The higher frequency sinusoid indicates the column of the key on the touch-pad and the lower frequency sinusoid indicates the row of the key on the touch-pad. Figure 5.1 shows the layout of a telephone keypad and the two DTFT frequencies corresponding to each digit, assuming the continuous-time waveform is sampled at 8192 kHz. The figure also contains a table listing each digit and the DTFT frequencies for that digit. For example, the digit 5 is represented by the signal

$$d_5[n] = \sin(0.5906\,n) + \sin(1.0247\,n). \tag{5.8}$$

	ω_{column}		
ω_{row}	0.9273	1.0247	1.1328
0.5346	1	2	3
0.5906	4	5	6
0.6535	7	8	9
0.7217		0	

Digit	ω_{row}	ω_{column}
0	0.7217	1.0247
1	0.5346	0.9273
2	0.5346	1.0247
3	0.5346	1.1328
4	0.5906	0.9273
5	0.5906	1.0247
6	0.5906	1.1328
7	0.6535	0.9273
8	0.6535	1.0247
9	0.6535	1.1328

Figure 5.1. DTFT frequencies for touch-tone signals sampled at 8192 Hz.

Basic Problems

In these problems, you will create the appropriate touch-tone for each digit, and examine the DTFT to make sure the signals have the correct frequencies. You will also define a vector containing the touch-tones for your phone number.

(a). Create row vectors d0 through d9 to represent all 10 digits for the interval $0 \leq n \leq 999$. Listen to each signal using sound. For example, sound(d2,8192) should sound like the tone you hear when you push a '2' on the telephone.

(b). The function fft can be used to compute N samples of the DTFT of a finite-length signal at frequencies $\omega_k = 2\pi k/N$. For example, X = fft(x,2048) computes 2048 evenly spaced samples of $X(e^{j\omega})$ at $\omega_k = 2\pi k/2048$ for $0 \leq k \leq 2047$. Use fft to compute samples of $D_2(e^{j\omega})$ and $D_9(e^{j\omega})$ at $\omega_k = 2\pi k/2048$. Define omega to be a vector containing ω_k for $0 \leq k \leq 2047$. Plot the magnitudes of the DTFTs for these signals, and confirm that the peaks fall at the frequencies specified in Figure 5.1. You will find it easier to see which frequencies are present if you use axis to restrict the ω-axis to the region $0.5 \leq \omega \leq 1.25$ while keeping the full vertical range. Generate appropriately labeled plots of the DTFT magnitudes for these two digits.

(c). Define space to be a row vector with 100 samples of silence using zeros. Define phone to be your phone number by appending the appropriate signals and space. For instance, if your phone number was 555-7319, you would type

```
>> phone = [d5 space d5 space d5 space d7 space d3 space d1 space d9];
```

Note that in order to append the signals this way, all of the digits you defined in Part (a) and space must be row vectors. Play your phone number using sound and verify that it sounds the same as when you dial it on a touch-tone phone.

Intermediate Problems

For these problems, you will learn to decode phone numbers from their touch-tones. The phone numbers to be decoded are in a file called touch.mat, which is in the Computer Explorations Toolbox. The Computer Explorations Toolbox can be obtained from The MathWorks at the address listed in the Preface. The data can be loaded into MATLAB by typing load touch as long as the file is in your MATLABPATH. If the file loaded correctly, you should be able to list the names of variables by typing

```
>> who

Your variables are:

hardx1    hardx2    x1    x2
```

The vectors x1 and x2 contain sampled versions of the sequence of touch-tones representing two different phone numbers. As in Part (c), the signals consist of 7 digits of 1000 samples each, separated by 100 samples of silence. The vectors hardx1 and hardx2 are less precisely dialed versions of the same numbers that are used in Part (i).

(d). Using fft, compute 2048 evenly spaced samples of the DTFT for each digit of x1. In order to apply fft to each digit separately, you will need to segment the signal into seven digits using the information you have about the relative lengths of the

digits and spaces, or by plotting the signal and identifying when each digit starts and stops. Apply `fftshift` to the output of `fft`, which will rearrange the samples of the DTFT so they correspond to ascending frequencies in the range $-\pi \leq \omega_k < \pi$. Define `X11` through `X17` to contain these samples of the DTFT you obtained by applying `fftshift` to the output of `fft` for each digit. To determine the digits of the telephone number represented by `x1`, plot the magnitude of the DTFTs and compare the peak frequencies of the signals with those shown in Figure 5.1 As a check for your answer, the sum of the digits should be 41.

(e). Repeat Part (d) for the signal `x2`, and decode the digits of this phone number as well. These digits may not sum to 41.

Advanced Problems

In these problems, you will write a function to decode phone numbers automatically from the touch-tones. To help design your decoder, you will look at the energy of the tones at each of the possible frequencies from Figure 5.1.

(f). Using `fft` to compute 2048 samples of $X(e^{j\omega})$, figure out which value of ω_k, and the corresponding index `k`, is closest to each of the touch-tone frequencies. Remember that MATLAB vectors begin with index `k=1`, so the DTFT sample at $\omega = 0$ is stored in `X(1)`.

(g). The value of $|X(e^{j\omega_k})|^2$ gives the energy in a signal at frequency ω_k. Apply `fft` to `d8`, which was defined in Part (a), and use the output of `fft` to compute $|D_8(e^{j\omega_k})|^2$ for each of the ω_k you determined in Part (f). Is the energy highest for appropriate values of ω_k corresponding to an '8'?

(h). Write a function `ttdecode` which accepts as its input a touch-tone signal in the format used in Part (c), (1000 samples of touch-tone for each digit, separated by 100 samples of silence), and returns as its output a seven element vector containing the phone number. For example, if the vector `phone` contained the touch-tones for the number 555-7319, you should get the following output:

```
>> testout = ttdecode(phone);
>> testout
testout =
5 5 5 7 3 1 9
```

The first line of the M-file you write to implement the function should read

```
function digits = ttdecode(x)
```

Your function should use `fft` to compute 2048 samples of the DTFT of each digit in `x`, and then check the energy at the samples you specified in Part (f) corresponding to the ω_k for the touch-tones. Pick which of the row frequencies and column frequencies has the most energy, then use those frequencies to determine what the digit is. Test

your function by using `x1` and `x2` as inputs and verify that it returns the same phone number as you obtained in Parts (d) and (e). Also, test your function on the signal you created in Part (c) to represent your own phone number.

(i). Most people do not dial their telephone with the ruthless precision assumed in this exercise, with each digit exactly the same length and the space between digits always the same length. Modify your function to work with touch-tone signals where the digits and silences can have varying lengths. To simplify things slightly, assume that all the tones and silences are at least 100 samples long. Verify the new version of your function works with the signals `hardx1` and `hardx2`. These signals contain the same tones as `x1` and `x2`, but they are not regularly spaced.

5.3 Discrete-Time All-Pass Systems

This exercise explores the effect of all-pass systems on discrete-time signals. All-pass systems are defined to have a frequency response with magnitude equal to one, i.e., $|H(e^{j\omega})| = 1$. The magnitude of the DTFT of the output signal of an all-pass system is identical to the magnitude of the DTFT of the input signal. However, as will be demonstrated in this exercise, the phase of the frequency response can lead to significant distortion in the output of the system.

You will work with two different discrete-time all-pass systems in this exercise. Both of these systems are causal LTI systems. Their inputs and outputs satisfy the following linear, constant-coefficient difference equations:

$$\text{System 1:} \qquad y[n] = x[n-3]\,,$$

$$\text{System 2:} \qquad y[n] - (3/4)y[n-1] = -(3/4)x[n] + x[n-1]\,.$$

Basic Problems

(a). Define coefficient vectors `a1` and `b1` for System 1 and use `freqz` to generate appropriately labeled plots of the magnitude and phase of the frequency response $H_1(e^{j\omega})$ of this system. If you are unfamiliar with `freqz`, you may wish to review Tutorial 3.2. Do your plots confirm that the system is an all-pass system?

(b). Define coefficient vectors `a2` and `b2` for System 2 and generate plots of the magnitude and phase of the frequency response of this system. Again, your plots should show that $|H_2(e^{j\omega})| = 1$ for all ω. Is the phase of $H_2(e^{j\omega})$ the same as the phase of $H_1(e^{j\omega})$? When the inputs to both systems are the same, would you expect the outputs to be the same?

Intermediate Problems

(c). Define `x` to be the signal $x[n] = (3/4)^n\, u[n]$ for $0 \le n \le 50$. Create an appropriately labeled plot of `x` using `stem`.

(d). Let $y_1[n]$ and $y_2[n]$ be the outputs of Systems 1 and 2 respectively when the input is $x[n]$. Use the function `filter` described in Tutorial 2.2 to compute the vectors `y1` and