EE 102A - Assignment 7

Nicholas Dwork

Problem 1. *The DFT Matrix*
a) Find the matrix W such that DFT$\{x\} = W\,x$.
b) Find the matrix W^{-1} such that DFT$^{-1}\{y\} = W^{-1}\,y$.
c) Show that $W^{-1} = \frac{1}{N}\,W^*$.

Problem 2. *Find the Transfer Function*
You are given a black box with input terminals and output terminals. Through some testing, the box seems to behave like a linear shift invariant system. You make the assumption that the box is linear and shift invariant and you want to find its transfer function. How do you do this?
(You might think that you can apply an impulse to the input and see what comes out. But it's impossible to generate a true impulse. Keep in mind that division by 0 is undefined.)

Problem 3. *Interpolation as Convolution*
Suppose you are supplied evenly spaced samples of a signal

$$(x(-N), x(-N+1), \ldots x(-1), x(0), x(1), \ldots x(N-1), x(N))$$

a) Show that zero order hold interpolation can be expressed as convolution with a shifted rect function.
b) Show that linear interpolation can be expressed as convolution with a tri function.
c) For each of the above, how does the spectrum of the interpolated signal relate to the spectrum of the original signal? What does interpolation do to the function? (Assume that the original function is band limited and that the sampling frequency is greater than twice the bandwidth.)

Problem 4. *The Frequencies of the DFT*
a) Suppose you compute the DFT of a 8 element in vector using the following Matlab commands.

```
dftVector = fft( inputVector );
```

What frequency does each element of `dftVector` correspond to?
b) Suppose instead, you using the following Matlab command.

```
dftVector = fftshift( fft( inputVector ) );
```

What frequency does each element of `dftVector` correspond to?
c) Repeat parts (a) and (b) for a 7 element vector.

Problem 5. Prove the following theorem:

$$\mathcal{F}\{\text{Ш}_\Delta\} = \frac{1}{\Delta}\text{Ш}_{1/\Delta},$$

where $\text{III}_\Delta(x) = \sum_{n=-\infty}^{\infty} \delta(x - n\Delta)$.

Problem 6. Let $f : \mathbb{R} \to \mathbb{C}$ such that $f(t) = 0$ for all $t < 0$. Show how to recover f only using the real part of its Fourier Transform.

Problem 7. The Nyquist Theorem states that we can perfectly reconstruct a band-limited function from its samples. In this theorem, the samples are evaluations of the function at specific points in its domain:

$$\{f(n\Delta) : n \in \mathbb{Z}\}.$$

Suppose, instead, the samples were captured using a more realistic Analog-to-Digital converter. Rather than evaluating the function at individual points, each sample was the result of a short integration over the domain:

$$\left\{ \int_{n\Delta-\epsilon}^{n\Delta+\epsilon} f(x)\, dx : n \in \mathbb{Z} \right\}.$$

If the signal is still reconstructed using sinc interpolation, how does the spectrum of the reconstruction relate to the true spectrum of the function? (Assume that the sampling frequency is greater than twice the bandwidth.)

Problem 8. *Amplitude Modulation Signals - by John Pauly*

The data for this problem is a segment of the AM-spectrum; it was captured using a Universal Software Radio Peripheral (USRP). This is an opensource project to develop radio transmitters and receivers that are USB peripherals, and that can be configured to handle almost any type of signal. It uses a 66 MHz A/D, and a programmable gate array to capture the signal, which can then be processed in software on your computer. The data comes from Michael Gray, and is published on his web site `http://www.kd7lmo.net`. He has examples of quite a few different signals he's captured, including GPS, FM, HF, and others.

The data we'll be looking at is the AM band. The capture is of about 15 seconds of a large part of the band sampled at 256 kHz. The AM spectrum in the US goes from 520 kHz to 1610 kHz, with channels separated by 10 kHz. You can download the data from the following link:

`https://web.stanford.edu/~ndwork/teaching/1706ee102a/hmwk/data/am_data.mat`

The signal has already been demodulated by 600 kHz once using a quadrature receiver. If we just low-pass filter and detect the envelope, we'd hear the station at 600 kHz. Unfortunately, the strong signals are elsewhere! We first need to find the signals, then demodulate to the right frequency, and then use an envelope detector to extract the audio waveform.

Some additional notes:

- The data is sampled at 256 kHz, which is much faster than your sound card can handle. After demodulating and filtering the signal, reduce the sampling rate to 8 kHz by taking every 256/8 = 32 samples. Also, scale it to ± 1.
- You might expect that since we have used a quadrature demodulator, we could simply listen to the real part of the signal using `sound(real(mySignal))`. But, unless you are extremely lucky, this will sound odd. This is due to the fact that you almost certainly haven't gotten the carrier frequency exactly right. An envelope detector does much better: `sound(abs(mySignal))`. Explain why in your answer to this problem.
- You can rely on the fact that your speaker or headphones don't respond to any constant voltage to eliminate any constant offset in your signal. However, this is not power efficient. Improve your receiver by explicitly suppressing the constant offset and scaling the offset-suppressed signal to ± 1.

- You can use an ideal low-pass filter, and that will work ok. Upgrade your receiver by using a smoother window as your low-pass filter. Did that do any better?

Write a Matlab function that accepts a frequency as input and outputs the sound of the station. Use this function to listen to two different stations. Write what you hear in your answer to this problem.