ELC 433-L1

Lab 2 - Discrete-Time Fourier Transform

Brian Worts and Chris Jenson

10/09/20



**Introduction:**

This lab built upon the fundamentals reviewed in the previous lab to analyze Discrete-Time Fourier Transforms (DTFT). The Fourier Transform is a key tool in understanding how we perceive time, send modulated info, the behavior of linear circuits, and even the Heisenberg Uncertainty Principle. It was also important to understand the relationship between the DTFT and the Discrete Fourier Transform (DFT). The DFT is the DTFT for a finite duration but sampled at uniformly spaced frequencies. From there, the relationship between the DFT and the Fast Fourier Transform (FFT) was analyzed. FFT is for a finite sequence of N points, and it is used to identify the number and frequency of sinusoidal components and or to identify the general spectral shape of a noise-like signal. An understanding of how to use the FFT to analyze discrete-time signals in the frequency domain was acquired.

**Procedure:**

This lab consisted of eleven different steps. Step 1 was to use the given signal information to first compute the DTFT of the sequence using paper and pen. Then to compute the complex magnitude of the DTFT. Step 2 was using MATLAB to plot the sequence for the n=[0,199], the resulting graph can be seen in results. Step 3 was plotting the function obtained in Step 1 for the interval of [0,2pi] and at 1024 angles.

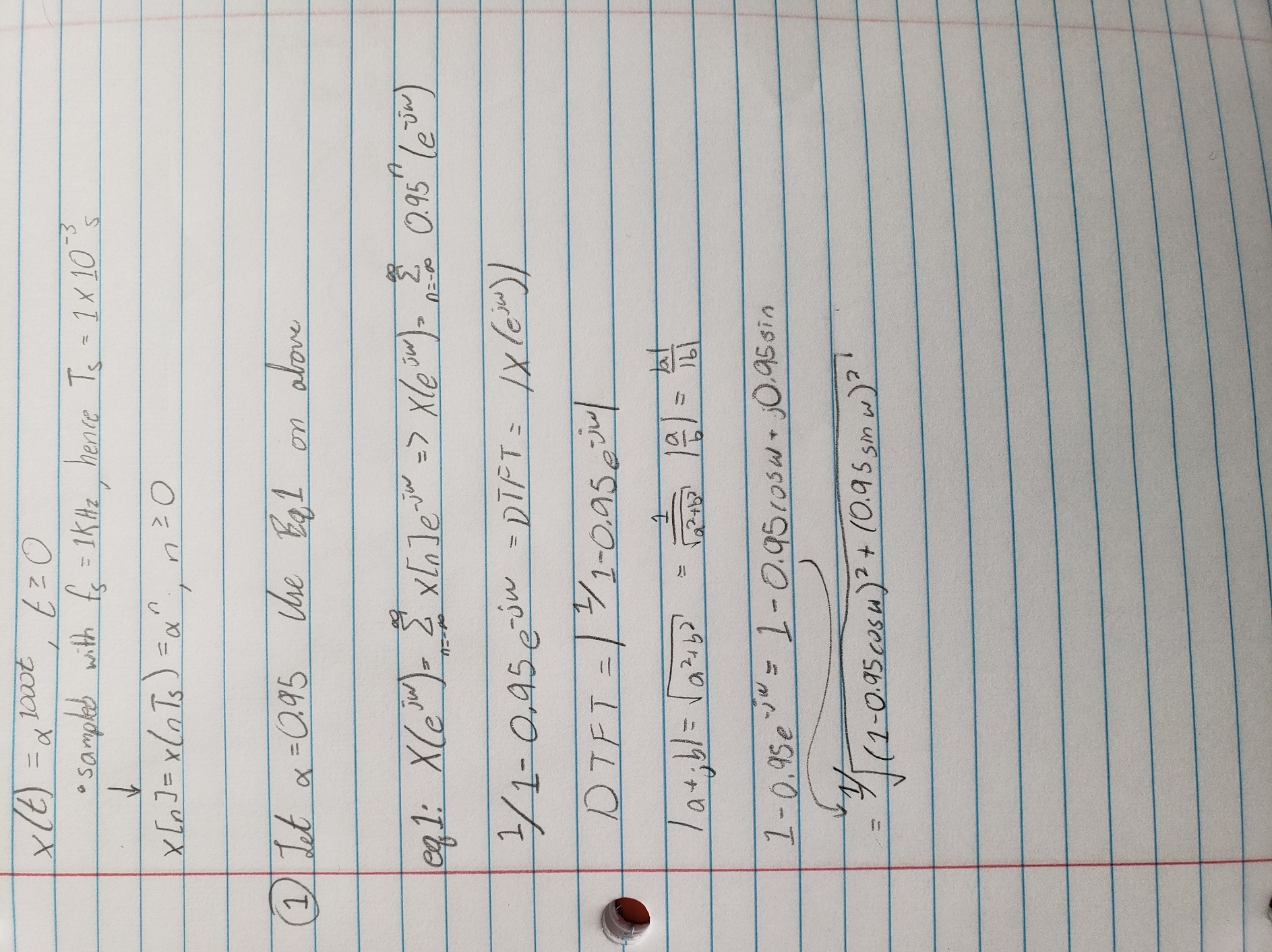
Step 4 used the MATLAB fft() function. It internally pads the sequence with zeroes before doing the operation. The results were plotted and were compared to the results from Steps 3 & 4. Step 5 is answered in the Results Questions section.

Step 6 was to do the FFT with N=256, a sequence was created. The resulting sum of vectors was plotted. Step 7 took the FFT of the length 256 sequence. The stem() and abs() function were applied and then it was plotted again with only the first 128 samples. The plot was then separated into real and imaginary components.

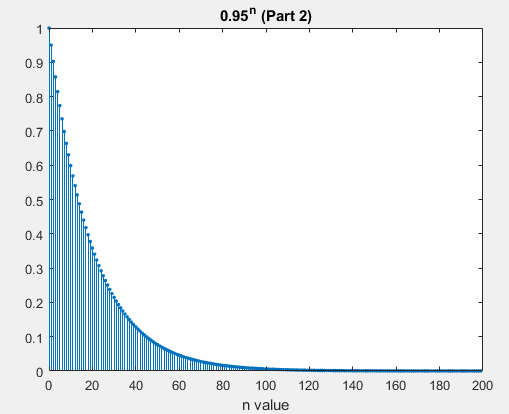
Step 8 computed and plotted another sinusoidal based on the given criteria. Step 9 was to analyze the plot. It is visible that the FFT is a sampled version of the DTFT. It is a DTFT signal which is the sine wave multiplied by a rectangular window function. Step 11 was hand drawing the given sketch.

**Results:**

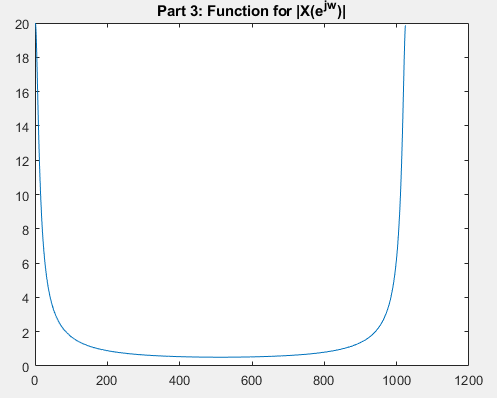
Part 1



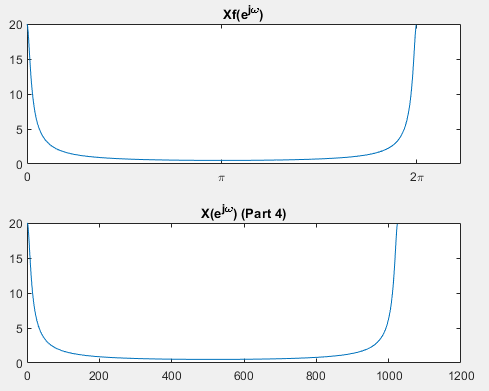
Part 2



Part 3



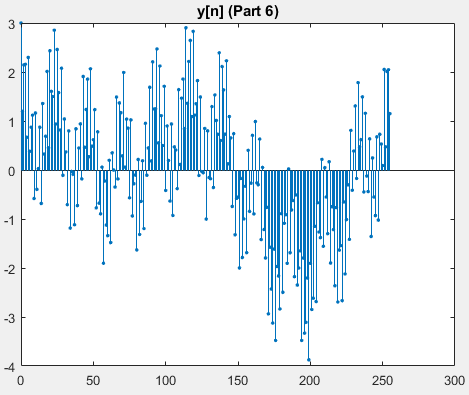
Part 4



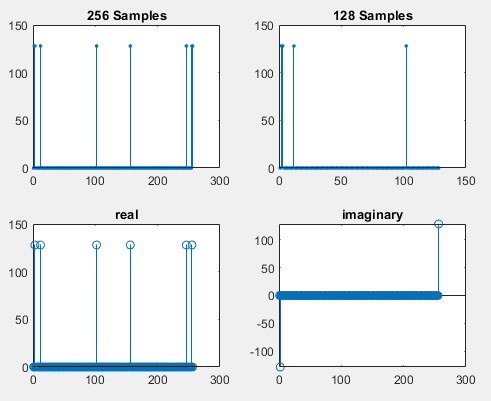
Part 5

* Questions
  + Note that the FTT of the truncated sequence seems to match the DTFT of the infinite length sequence quite well. Why do you think that is?
    - The truncated sequence seems to match the DTFT because the frequency response repeats with every interval of 2pi. This means that the truncated portion removed a repeat of the sequence.
    - No, not at all! The reason is that, in the discrete time domain, the truncated part of the sequence represents a tiny amount of missing energy.

Part 6

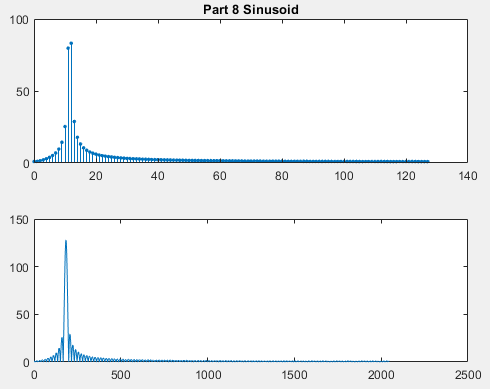


Part 7



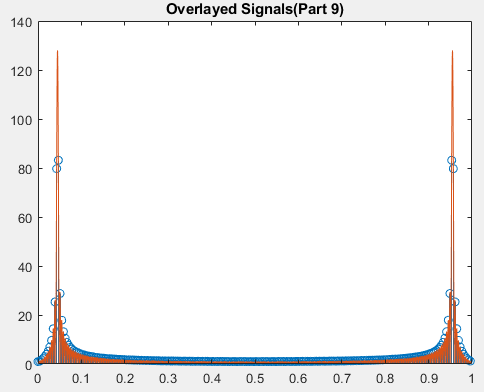
* Question
  + Explain all the results. Explain how you might interpret the FFT to determine the number of frequencies and sinusoids
    - For 256 samples, each spike represents a sinusoid added to the signal.
    - No – there is mirror symmetry. In the 256-sample block each pair of spikes represents one sinusoid.
    - The plot is reflected which accounts for the additional spikes. With 128 samples, the reflected portion is removed. Both graphs can be used to find the frequency of the sinusoids by finding the index of the corresponding spike. The frequency can then be calculated using equation 3 from the lab handout. The imaginary component is shown to only have 2 points while the real has 6 points, combined, these are the points that make up the 8 spikes shown in the 256-sample graph.

Part 8



Part 9

* Questions
  + How many sinusoids would you guess are represented?
    - We would guess that there are many sinusoids summed together but there is only one.

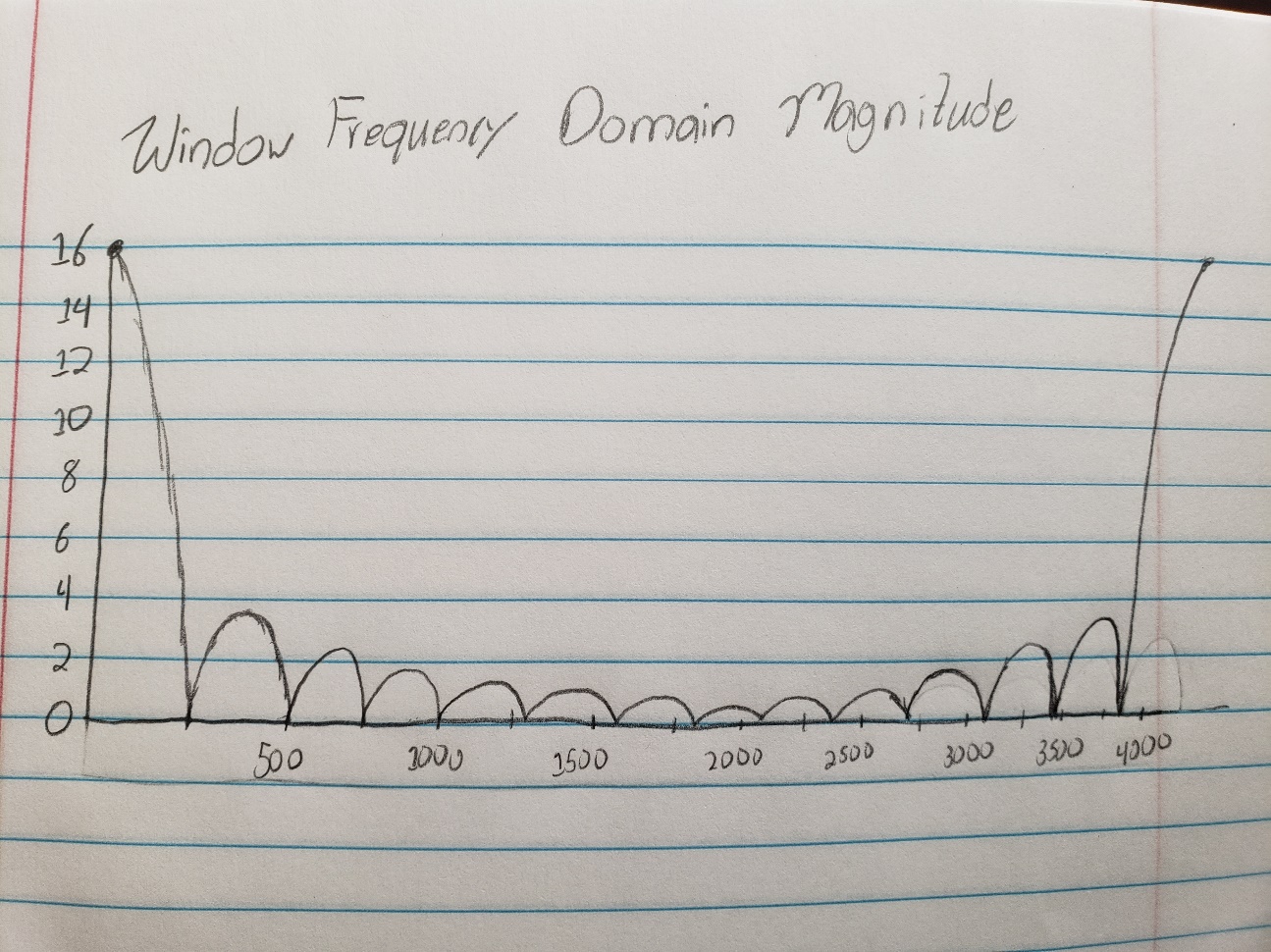


Part 10

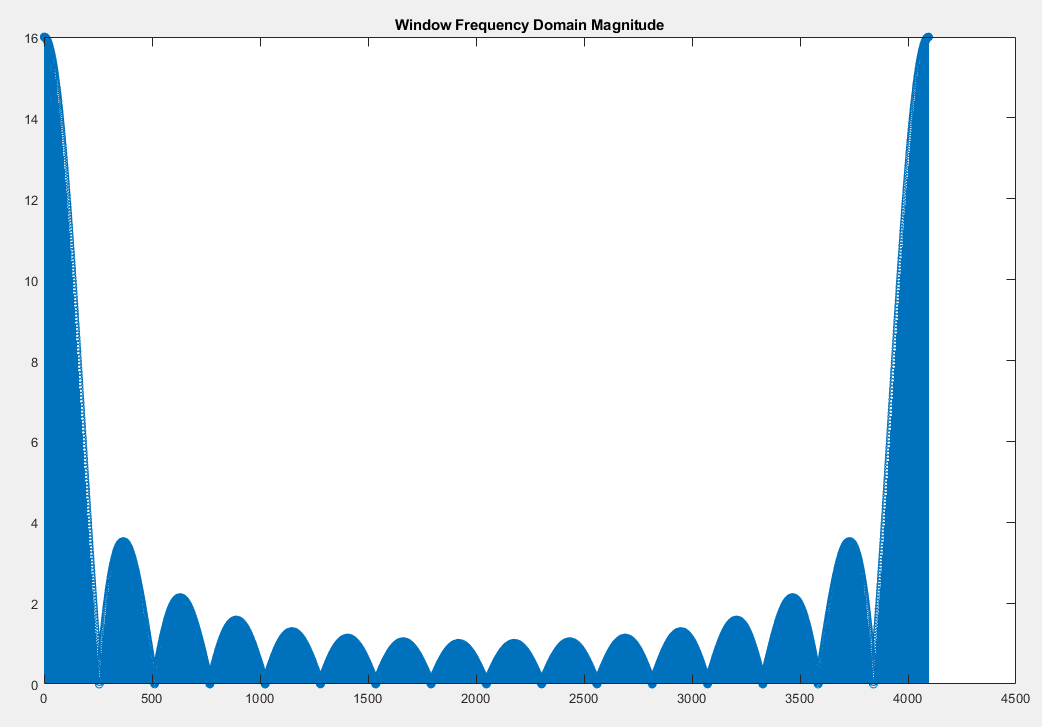
* Questions
  + Why did the results in Step 7 look so clean and perfect, but the results for Step 8 & 9 and so gnarly?
    - Step 7 looks clean and perfect because the FFT is based on a finite amount of data from a continuous signal. Exact intervals of the sinusoid period are used (this is not very precise). More specifically, there are an integer number of periods of each of the sinusoids in the specific finite sized sample. The repeating sampled signals results in clean and perfect sine waves. Step 8 used the FFT of non-integer number of periods of the sinusoids. Thus, when repeating, the signal no longer came together as clean and perfect waves.

Part 11

* By Hand



* By MATLAB … but really x-axis is 0 to 2. Also, you don’t have enough wiggles!



**Engineering Work:**

%BRIAN WORTS AND CHRIS JENSON

%ELC 433-L1

%LAB 2

close all;

e = 2.7182818284590452353602874713527;

%PART 2

n = 0:199;

alpha = 0.95;

x = alpha.^n; // more comments?

figure

stem (n,x,'filled','MarkerSize',2)

xlabel('n value')

title(strcat(num2str(alpha),'^n (Part 2)'))

%PART 3

sampling = 0:1023;

w = (2\*pi/1024)\*sampling;

// Comments!

for i = 1:1024

f(i) = 1/sqrt(((1-0.95\*cos(w(i))).^2)+(0.95\*sin(w(i))).^2);

end

figure

plot(f)

title('Part 3: Function for |X(e^{jw})|')

%PART 4

N = 1024;

Xf = fft(x,N);

figure

subplot(2,1,1)

plot(w,abs(Xf))

title('Xf(e^{j\omega})')

xticks([-3\*pi -2\*pi -pi 0 pi 2\*pi 3\*pi]) % set ticks to be pi-ish

xticklabels({'-3\pi','-2\pi','-\pi','0','\pi','2\pi','3\pi'}) % pi-ish labels

subplot(2,1,2)

plot(f)

title('X(e^{j\omega}) (Part 4)')

%Part 5

%Note that the FFT of the truncated sequence seems to match the DTFT of the

%infinite length sequence quite well. Why do you think that is?

%Part 6

N = 256;

for n = 0:(N-1)

y1(n+1) = sin(2\*pi\*n/N);

y2(n+1) = cos(4\*pi\*n/N);

y3(n+1) = cos(22\*pi\*n/N);

y4(n+1) = cos(202\*pi\*n/N);

end

y = y1+y2+y3+y4;

figure;

stem(0:255,y,'filled','MarkerSize',2)

title('y[n] (Part 6)')

%Part 7

ffty = fft(y);

ffty128 = ffty(1:128);

figure()

subplot(2,2,1)

stem(abs(ffty),'filled','MarkerSize',2)

title('256 Samples')

subplot(2,2,2)

stem(abs(ffty128),'filled','MarkerSize',2)

title('128 Samples')

subplot(2,2,3)

stem(real(ffty)) %Real has symmetry

title('real')

subplot(2,2,4)

stem(imag(ffty)) %imag has conjugate symmetry

title('imaginary')

%Part 8

t = (0:255)/256; %0<t<1 t=1/n, 2/n... 255/n

y5 = cos(23\*pi\*t); % exactly 11.5 cycles

Fy = fft(y5); % take the FFT

figure

subplot(2,1,1) % stack 2 vertical, 1

Fy\_16 = fft(y5,N\*16); % FFT with 4096 samples

stem(0:(N/2-1),abs(Fy(1:N/2)),'filled','MarkerSize',2)

title('Part 8 Sinusoid')

subplot(2,1,2) % stack 2 vertical, 2nd

plot(0:(16\*N/2-1),abs(Fy\_16(1:16\*N/2))) % line plot for det

%Part 9

%How many Sinusoids are represented?

%Superimpose both plots while scaling the x axis

figure()

stem((0:255)/256,abs(Fy))

hold on

plot((0:4095)/4096,abs(Fy\_16))

hold off

title('Overlayed Signals(Part 9)')

%2 peaks may make it seemm like there are 2 sine waves, but there is

%actually

figure()

w = 2\*pi\*(0:4095)/4096;

y9 = abs(sin(w.\*16./2)./sin(w./2));

stem(0:4095,y9) %Hand draw this with N = 16

title('Window Frequency Domain Magnitude')

|  |  |  |  |
| --- | --- | --- | --- |
| **Description** | **Expectation** | **Max Pts.** | **Pts. Deducted** |
| Introduction | Brief overview | 1 |  |
| Procedure | Brief description of procedures | 1 |  |
| Design/Engineering Work |  |  |  |
|  | Mathematical analysis and code are correct. | 2 | 0.2 |
|  | Paste text of code, not images | 0.5 |  |
|  | Use SPACE not TAB characters. Strict indentation. Consistent placement of IF, ELSE, FOR, END | 0.5 |  |
|  | Full commenting | 1 | 0.2 |
| Results | Plots with informative captions. Careful hand sketches. Do all steps, present all requested results, answer all questions. |  |  |
|  | Why do you think that the DTFT of infinite length exponential matches FFT of finite length sequence? | 0.5 | 0.2 |
|  | Based on results of Part 7, explain how you might interpret the FFT to determine the number and frequencies of sinusoids. | 0.5 | 0.2 |
|  | Looking just at the top plot, how many sinusoids would you guess are represented? | 0.5 |  |
|  | Why did the results in Step (7) look so clean and perfect, but the results for Steps (8) & (9) are so gnarly? | 0.5 | 0.2 |
| Knowledge Gained |  | 1 |  |
| Who Did What |  | 1 |  |
| **Total** |  | 9 |  |

**Knowledge Gained:**

The students both gained knowledge in the uses of DTFTs and how they can be used to map a potentially infinite sequence of numbers. The DFT is defined as a finite sequence and is equivalent to the DTFT of the same length but sampled at a different rate. The DFT of a sample of finite size and highly zero-padded can approximate the DTFT with precise sampling in the frequency domain. For signals that are a sum of sinusoidal, the FFT can be used to discover the exact frequencies and amplitudes of the components. When signals containing sinusoids do not have an integer number of cycles, then there is a smearing effect. This can cause the signal to look like multiple frequencies. In addition to this, the students' existing understanding of various MATLAB functionalities was increased.

**Who Did What:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Student** | **Analysis** | **Development** | **Coding** | **Results** | **Writing** |
| Brian | 50 | 50 | 75 | 50 | 25 |
| Chris | 50 | 50 | 25 | 50 | 75 |