Taylor Larrechea

Dr. Gustafson

Math 362 Fourier Analysis

October 19, 2017

Ch. 4.3 HW

Section 4.3

4.3.9

For the value below, for a given signal sampled at nodes, as indicated, determine the largest value of for which we can accurately compute and .

Since all we have to do to find the useful DFT terms is compute . The computed value for this is 32. So therefore, the value is 32.

M=32

4.3.18

For the function on [0,1] given below adapt the MATLAB commands found in this section for to do the following. Show MATLAB commands used.

1. Plot the function on [0,1].
2. Compute .
3. Compute the Nyquist frequency index and comment on whether there is conjugate symmetry about the entry in **c**.
4. Compute and
5. Organize the results for and in a table.

|  |  |
| --- | --- |
| Input Commands | Output (Plot if applicable) |
| >> N=8;  >> t=[0:1/N:(N-1)/N]';  >> f=4\*t-1;  >> plot(f) |  |

b.)

|  |  |
| --- | --- |
| Input Commands | Output (Plot if applicable) |
| >> N=8;  >> t=[0:1/N:(N-1)/N]';  >> f=4\*t-1;  >> plot(f);  >> c=(1/N)\*fft(f) | c =  0.7500 + 0.0000i  -0.2500 + 0.6036i  -0.2500 + 0.2500i  -0.2500 + 0.1036i  -0.2500 + 0.0000i  -0.2500 - 0.1036i  -0.2500 - 0.2500i  -0.2500 - 0.6036i |

c.)

The Nyquist frequency is calculated by the following formula, . For this case the Nyquist frequency would be .

For this example, there is a conjugate symmetry about the entry of **c**.

d.)

|  |  |
| --- | --- |
| Input Commands | Output (Plot if applicable) |
| i.)  >> a0=real(c(1))  ii.)  >> a=2\*real(c(2:4))  iii.)  >> b=-2\*imag(c(2:4)) | i.)  a0 =  0.7500  ii.)  a =  -0.5000  -0.5000  -0.5000  iii.)  b =  -1.2071  -0.5000  -0.2071 |

e.)

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| 0 | 0.75 | - | - |
| 1 | -0.50 | -1.2071 | -0.2500 + 0.6036i |
| 2 | -0.50 | -0.50 | -0.2500 + 0.2500i |
| 3 | -0.50 | -0.2071 | -0.2500 + 0.1036i |

4.3.27

Refer to the function on [0,1] from the Exercise specified. Adapt either of the MATLAB methods described in this section for to plot the frequency domain plot of using only the non-aliased DFT values. Show all MATLAB commands used.

|  |  |
| --- | --- |
| Input Commands | Output (Plot if Applicable) |
| >> N=64;  >> t=[0:1/N:(N-1)/N]';  >> f=4\*t-1;  >> c=(1/N)\*fft(f);  >> stem(abs(c(1:N/2))) |  |

4.3.33

For the following sound wave, adapt the MATLAB commands used in this section to do the following. Show all MATLAB commands used.

1. Find the length of the sound wave vector **x**.
2. Compute the Nyquist frequency index
3. Plot the full sound wave vector **x**.
4. Plot a stable zoomed-in portion of the sound wave.
5. Compute **c** and list the values of for

“Hello World”

a.)

|  |  |
| --- | --- |
| Input Commands | Output (Plot if Applicable) |
| >> x=audioread('helloworld.wav');  >> N=length(x) | N =  81920 |

Length

b.)

|  |  |
| --- | --- |
| Input Commands | Output (Plot if Applicable) |
| >> N/2 | ans =  40960 |

c.)

|  |  |
| --- | --- |
| Input Commands | Output (Plot if Applicable) |
| >> x=audioread('helloworld.wav');  >> plot(x) |  |

d.)

|  |  |
| --- | --- |
| Input Commands | Output (Plot if Applicable) |
| >> x=audioread('helloworld.wav');  >> plot(x(8000:9000)) |  |

e.)

|  |  |
| --- | --- |
| Input Commands | Output (Plot if Applicable) |
| >> x=audioread('helloworld.wav');  >> N=length(x);  >> c=(1/N)\*fft(x);  >> c(1:5) | ans =  -0.0466 + 0.0000i  0.0050 + 0.0006i  0.0007 - 0.0010i  0.0253 - 0.0449i  -0.0106 + 0.1288i |