Assignment 5: Solutions

1. Assume that a complex multiply takes 1 μs and that the amount of time to compute a DFT or FFT is determined by the amount of time it takes to perform all of the multiplications.

    a) How much time does it take to compute a 1024-point DFT directly?

    b) How much time is required if an FFT is used?

    c) Repeat parts A and B for a 4096-point DFT

Solution:

Directly evaluating the DFT results in N^2 multiplications

Directly evaluating the FFT results in $\frac{N}{2}\log_2 N$ multiplications

A.) How much time does it take to compute a 1024-point DFT directly

$$\text{1024-Point DFT} = N^2 * 1\mu s = 10242 * 1\mu s = \boxed{1.04857 \text{ seconds}}$$

B.) How much time is required if an FFT is used

$$\text{1024 Point FFT time} = \frac{N}{2}\log_2 N * 1\mu s = \frac{1024}{2}\log_2 1024 * 1\mu s = \boxed{0.00512 \text{ seconds}}$$

C.) Repeat parts A and B with 4096

$$\text{4096 – Point DFT Time} = N^2 * 1\mu s = 4096^2 * 1\mu s = \boxed{16.777216 \text{ seconds}}$$

$$\text{4096 – Point FFT Time} = \frac{N}{2}\log_2 N = \frac{4096}{2}\log_2 4096 * 1\mu s = \boxed{0.024576 \text{ seconds}}$$

2. Sampling a continuous-time signal, $x_a(t)$, for 1 second generates a sequence of 4096 samples

a) What is the highest frequency in $x_a(t)$ if it was sampled without aliasing?

b) If the 4096 point FFT of the sampled signal is computed, what is the frequency spacing, in Hertz, between the output points?

Solution:

A.) What is the highest frequency in $x_a(t)$ if it was sampled with aliasing?

Sample Rate = fs = $4096\,\dfrac{samples}{\sec ond}$

$$Max\ Frequency = f_m$$
$$2 * f_m = f_s$$
$$f_m = f_s/2$$
$$f_m = \frac{4096}{2} = \boxed{2048\text{ Hz}}$$

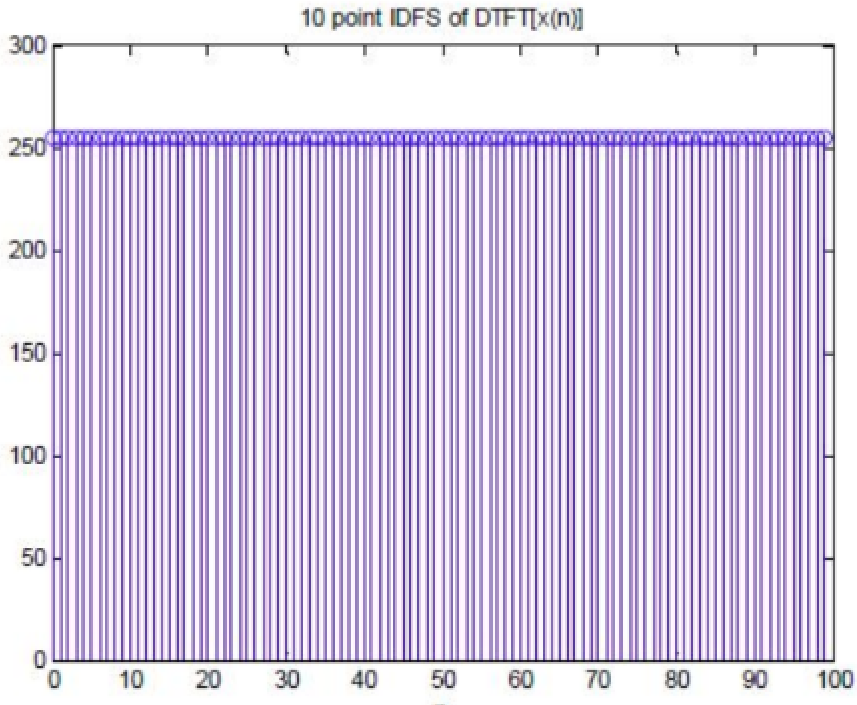
B.) If the 4096 point FFT of the sampled signal is computed, what is the frequency spacing, in Hertz between the output points?

$$\text{FFT bins between 0 and } f_m = \frac{N(i.e.,\ fft\ points)}{2} = \frac{4096}{2} = 2048\text{ bins}$$

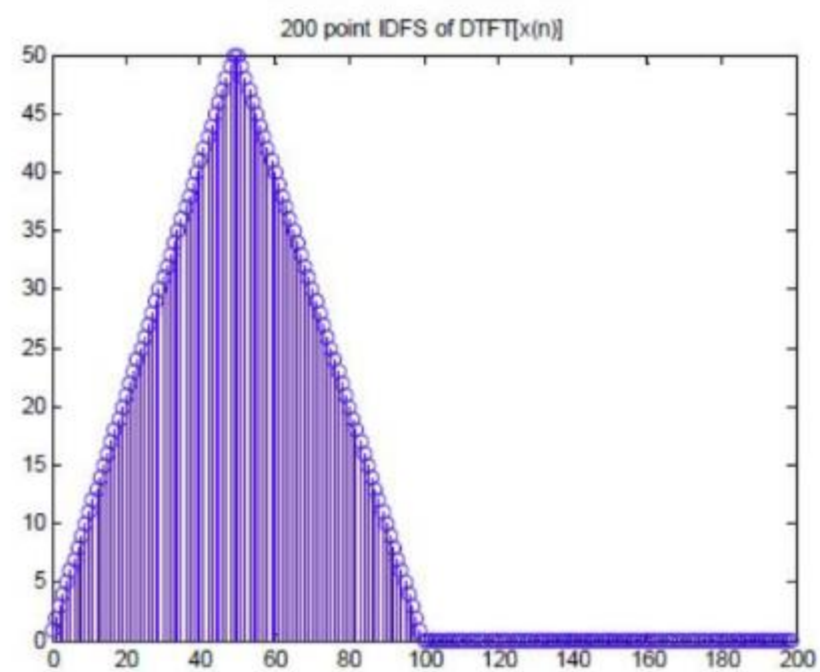
$$\text{Freq Spacing} = \frac{f_m}{fft\ bins} = \frac{2048\,Hz}{2048\,bins} = 1\,\frac{Hz}{bin} = \boxed{1\text{ Hertz}}$$

3. Let $X(\omega)$ be the DTFT of a finite -length sequence

a) Let 10-points (aliasing)



b) Let 200-points (not aliased)

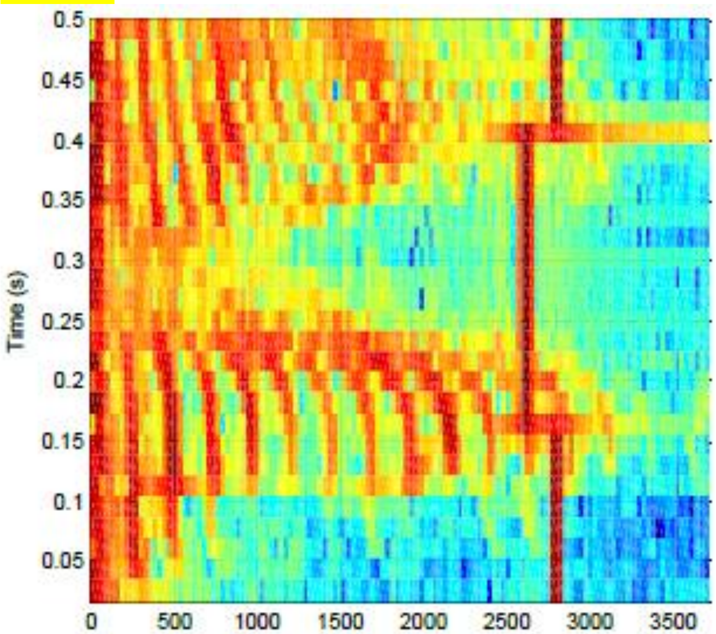


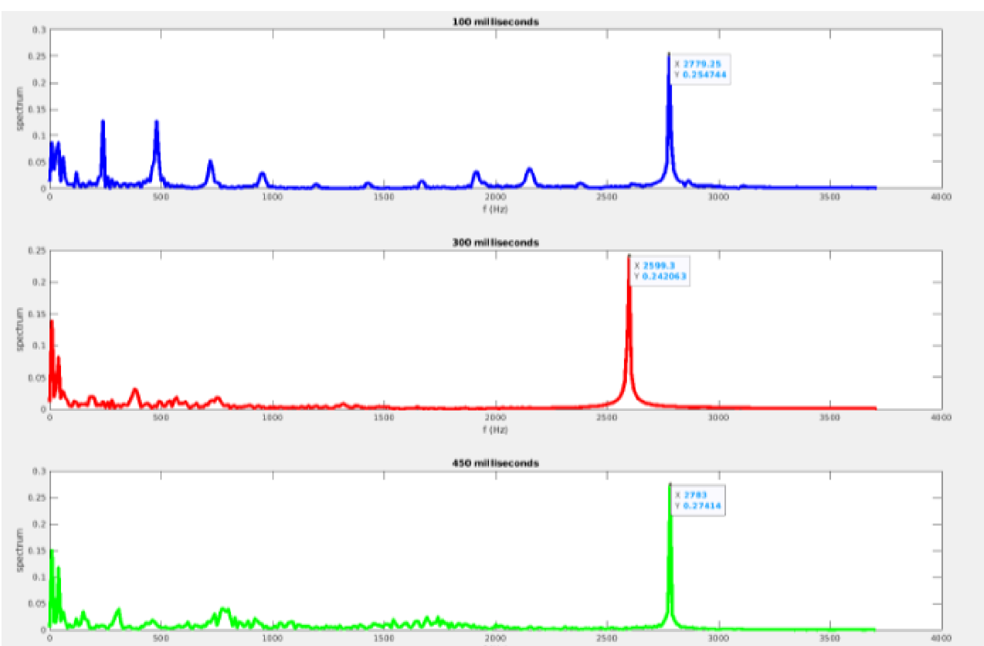
```
n = 0:99;
for i = 1:length(n)
    if n(i) >= 0 && n(i) <= 49
        x_n_3(i) = n(i)+1;
    elseif n(i) >= 50 && n(i) <= 99
        x_n_3(i) = 100 - n(i);
    else
        x_n_3(i) = 0;
    end
end

figure();
stem(n, x_n_3);
xlabel('Sample Number'); ylabel('Amplitude');
title('Question 3 Signal, x_n');
saveas(gcf,'question_3_x_n.png');

N = 10;
k = 0:N-1;
omega = 2*pi*k / N;
X_omega = dtft(x_n_3,n,omega);
```

```
y1 = idfs(X_omega,N);

figure();
stem(k, abs(y1));
xlabel('Sample Number'); ylabel('Amplitude');
title('Question 3 Signal, y1_n (10 samples)');
saveas(gcf,'question_3_y1_n.png');

N = 200;
k = 0:N-1;
omega = 2*pi*k / N;
X_omega = dtft(x_n_3,n,omega);
y1 = idfs(X_omega,N);

figure();
stem(k, abs(y1));
xlabel('Sample Number'); ylabel('Amplitude');
title('Question 3 Signal, y2_n (200 samples)');
saveas(gcf,'question_3_y2_n.png');
```

4. For this problem please review the MATLAB provided in Bb named “HW5_Signal.m.” Running this file will download a signal named “Sig” which is a combination of speech and interference. For the homework you are to perform FFT frequency analysis of Sig at three times: 100 ms, 300 ms, and 450 ms. You should display the FFT magnitude for real positive frequencies only. You are to identify frequency bands for the interference which could be used for later filter design.

Solution:





Example Code:

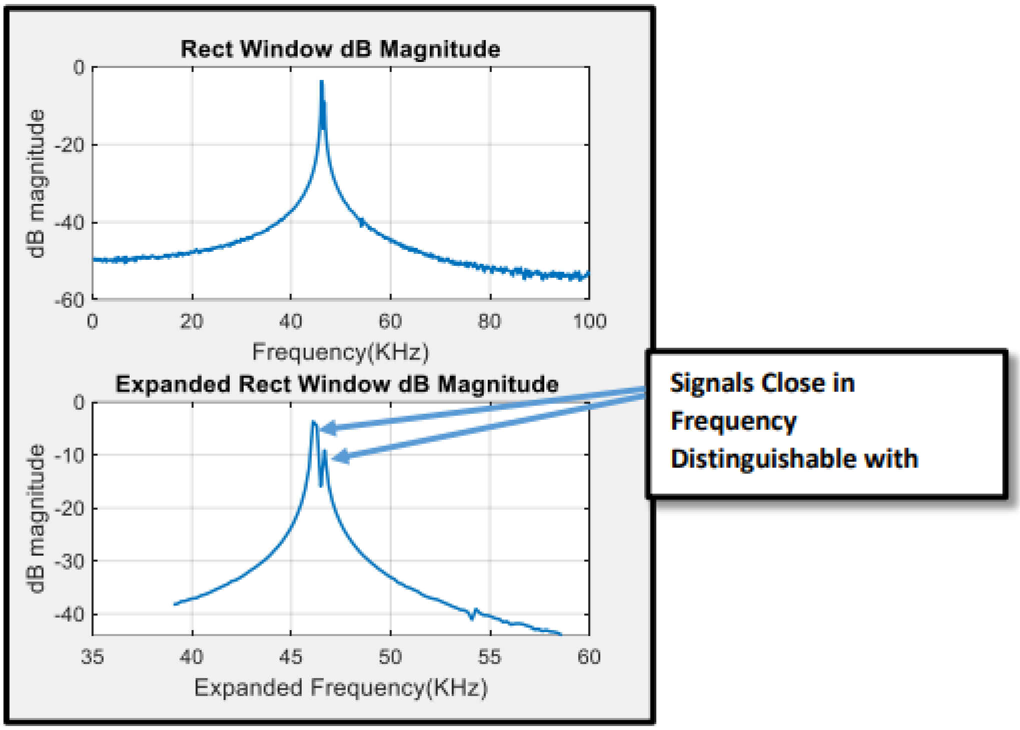
4. We look at subsections of the signal ($50ms$ on either side) for each of center times 100, 300, and $450ms$. For each of these sub-signals, we plot the FFT. The results are shown in Figure 3

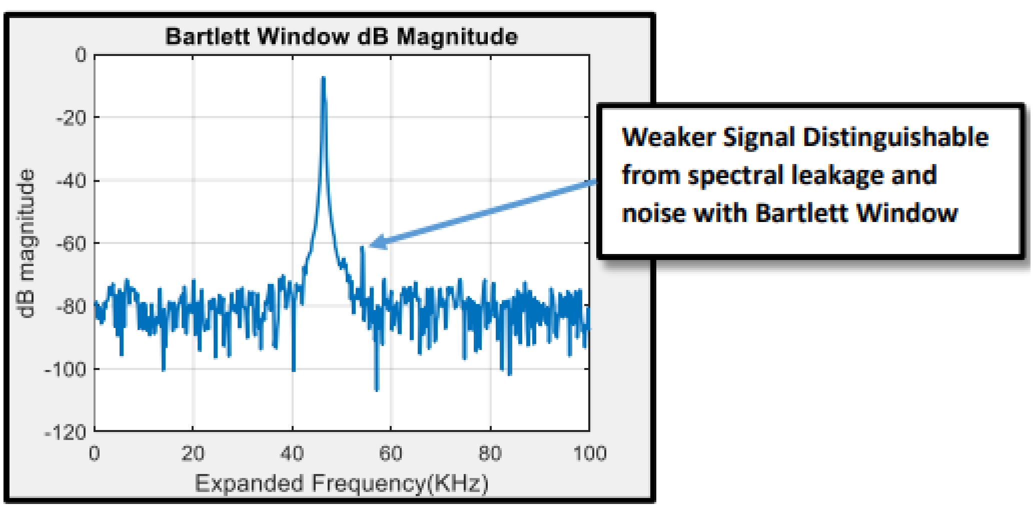
It looks like the frequencies we would want to remove are about $2800Hz$, $2600Hz$, and $2783Hz$. I did not compute the max in **MATLAB**, so the true tonal frequencies may vary by a few hertz. The **MATLAB** code to generate the above plot is given here:

```
load HW5_Sig
t = (0:1/Fs:(length(Sig)-1)/Fs)*1000;
% in milliseconds
figure(1)

subplot(3,1,1)
x1 = Sig(find(t>50 & t<150));
L = length(x1);
f = Fs*(0:(L/2))/L;


Y = fft(x1);
P2 = abs(Y/L);
P1 = P2(1:L/2+1);
P1(2:end-1) = 2*P1(2:end-1);
plot(f, P1,LineWidth=4, Color="blue")
xlabel("f (Hz)")
ylabel("spectrum")
title("100 milliseconds")

subplot(3,1,2)
x2 = Sig(find(t>250 & t<350));
L = length(x2);
f = Fs*(0:(L/2))/L;
Y = fft(x2);
P2 = abs(Y/L);
P1 = P2(1:L/2+1);
P1(2:end-1) = 2*P1(2:end-1);
plot(f, P1,LineWidth=4, Color="red")
xlabel("f (Hz)")
ylabel("spectrum")
title("300 milliseconds")


subplot(3,1,3)
x3 = Sig(find(t>400 & t<500));
L = length(x3);
f = Fs*(0:(L/2))/L;
Y = fft(x3);
P2 = abs(Y/L);
P1 = P2(1:L/2+1);
P1(2:end-1) = 2*P1(2:end-1);
plot(f, P1,LineWidth=4, Color="green")
xlabel("f (Hz)")
ylabel("spectrum")
title("450 milliseconds")
```

In MATLAB construct a signal consisting of four parts: three sinusoid components and one noise component. The signal is given by

You are to calculate and display the magnitude (in dB) of the spectrum as a function frequency for two window functions. The first is to be a rectangular window and the second is to be a Bartlett (Triangular) window. In both cases comment on your observations of the displayed results. The noise is Gaussian or Normal with zero mean.





$$F_1 = 46.00\,KHz,\quad A_1 = 1.0 \quad \textit{Strong Signal}$$
$$F_2 = 46.38\,KHz,\quad A_2 = 0.4 \quad \textit{Signal nearby in frequency}$$
$$F_3 = 54.00\,KHz,\quad A_3 = .002 \quad \textit{Weak Signal atDistant frequency}$$
$$NoiseMean = 0$$
$$Noise\,Variance = 10^{-5}$$
$$Fs = \textit{sample frequency} = 200\,KHz$$
$$N = \textit{FFT Lemgth} = 1024$$

```
%Script File:  FFT_Window_Problem.m

%Initialization
close all   % close all figure windows
clear all   % clear workspace
clc         % Comannd Line Clear

%Discrete signal and noise
KHz=1e3; ms=1/KHz;

N=1024;n=[1:1024]; %(FFT length)
F1=46*KHz; A1=1;
F2=46.38*KHz; A2=0.4;
F3=54*KHz; A3=.002;

NoiseMean=0;NoiseVariance=1e-5;NoiseStd=sqrt(NoiseVariance);
Fs=200*KHz; Ts=1/Fs;

Sig1=A1*cos(2*pi*F1*Ts*n);
Sig2=A2*cos(2*pi*F2*Ts*n+pi/4);
Sig3=A3*cos(2*pi*F3*Ts*n);
Noise=NoiseMean + NoiseStd*randn(1,N);
x=Sig1+Sig2+Sig3+Noise;

Farray=[1:512]*Fs/1024;
wr=rectwin(1024).';  %rectangluar window
xr=wr.*x; Xr=fft(xr)/512; magXr=abs(Xr).^2;
dBXr=10*log10(magXr);

figure(1)
Pos=get(1,'Position');Delta=[0 -200 0 200];set(1,'Position',Pos+Delta)
subplot(2,1,1)
   dBXr=10*log10(magXr);
   plot(Farray/1e3,dBXr([1:512]),'LineWidth',2);grid on
   set(gca,'FontSize',14)
   title('Rect Window dB Magnitude', 'fontsize',16)
   xlabel('Frequency(KHz)','FontSize',16)
   ylabel('dB magnitude','FontSize',16)
subplot(2,1,2)
   plot(Farray([200:300])/1e3,dBXr([200:300]),'LineWidth',2),grid on


   set(gca,'FontSize',14)
   title('Expanded Rect Window dB Magnitude ', 'fontsize',16)
   xlabel('Expanded Frequency(KHz)','FontSize',16)
   ylabel('dB magnitude','FontSize',16)
%-----------------------------------
wb=bartlett(1024).';
xb=x.*wb; Xb=fft(xb)/512; magXb=abs(Xb).^2;
dBXb=10*log10(magXb);
figure(2)
plot(Farray/1e3,dBXb([1:512]),'LineWidth',2),grid on
  set(gca,'FontSize',14)
  title('Bartlett Window dB Magnitude', 'fontsize',16)
  xlabel('Expanded Frequency(KHz)','FontSize',16)
  ylabel('dB magnitude','FontSize',16)
```