Lab 2

Sampling

0. Abstract

Sampling the continuous signal is a very important technique in the digital signal processing, since we need to first transfer the continuous signal to the discrete ones so that computer can deal with it. In Lab2, we will sample some continuous signals and use Fast Fourier Transfer to turn the signal into frequency domain. We will get to understand how the sample rate and sample number will effect the result of the FFT.

1. Sections

1.1. Read the data(2.1 in Lab)Sample a sinusoidal signal

The a sinusoidal signal is sampled with amplitude 1 and frequency of 128 Hz at a sampling rate of 2, 048Hz. The code is as follows:

```
n= 0:1:31;
xn = sin(2*pi*128*n/2048);
stem([0:31],xn)
```

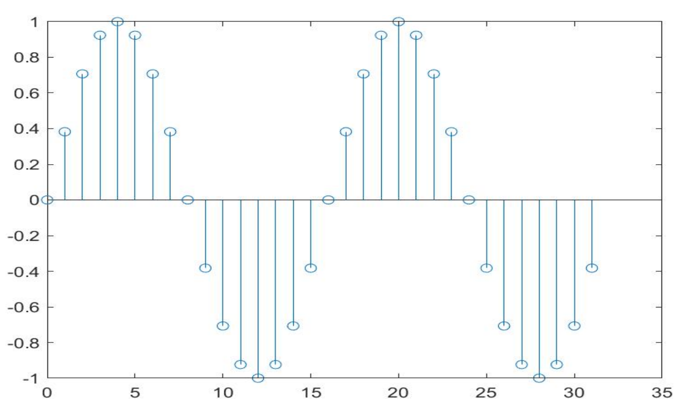

Figure 1: The sample of a sinusoidal signal

1.2. Find the DFT of the sequence

By using the fft and stem in the matlab, a graph having the frequency between 0 and the Nyquist frequency can be plot.

```
fs=2048;
Ndata=32;
N=32;
n=0:Ndata-1;t=n/fs;
for n = 1:32
x(n)=sin(2*pi*128*n/2048);
end
y=fft(x,N);
mag=abs(y);
f=(0:N-1)*fs/N;
stem(f(1:N/2),mag(1:N/2)*2/N);
xlabel('Frequency(Hz)');ylabel('|x|');
title('Ndata=32 Nfft=32');grid on;
```

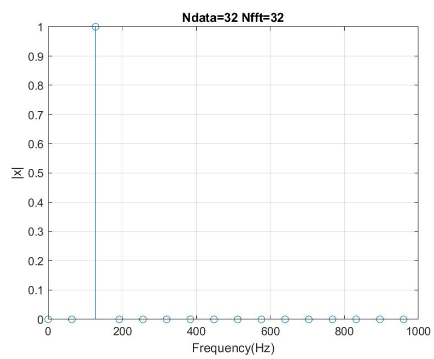

Figure2: The magnitude spectrum of a real 128Hz sinusoidal sampled at 2048Hz

1.3. Answer for questions

The phase in pi radian at the 128 Hz should be zero. This answer makes senses because when we sample a 128Hz at the rate of 2048 and sample a whole period. So the phase at the frequency 128Hz is the same as the original signal.

1.4. Sample a sinusoid at 220 Hz

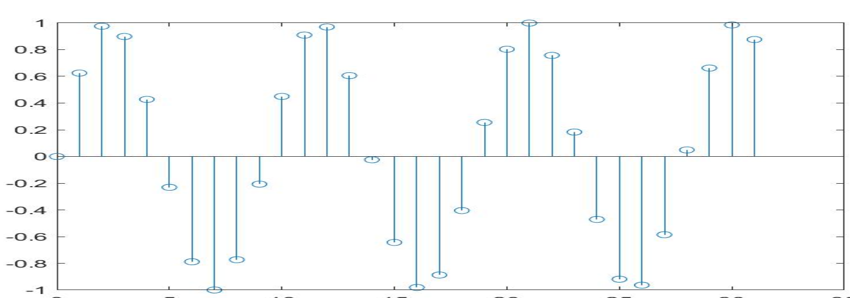

Figure 3: Sample a sinusoid at 220Hz

1.5. Draw the hypothesis

The hypothesis graph is as follows:

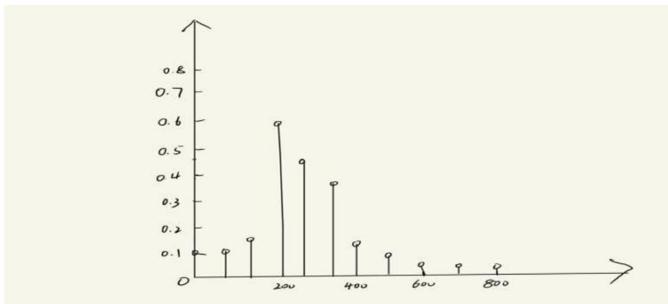

Figure 4: The hypothesis of the magnitude DFT of the signal.

The sampling rate we used at 1.1 is 2048Hz and we also take 32 samples to complete a whole period, so we can restore the frequency of the sinusoid. However, 2048Hz is not a in multiplies of 220Hz, so after sampling and DFT, there will be many other frequencies other than 220Hz.

1.6. The DFT of this signal

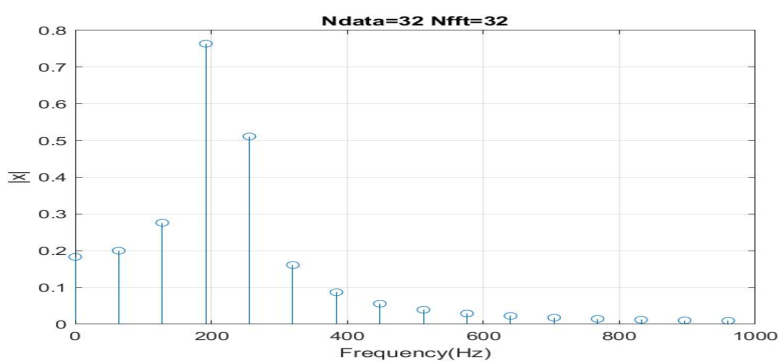

Figure 5: The magnitude of the result versus frequency

In 1.2 the DFT gets only frequency that the sinusoid is(220Hz). But in 1.6, it shows that all the frequencies possible with 32 uniformly spaced sample are present. And this is consistent with my hypothesis.

1.7. Plot the time domain sequence

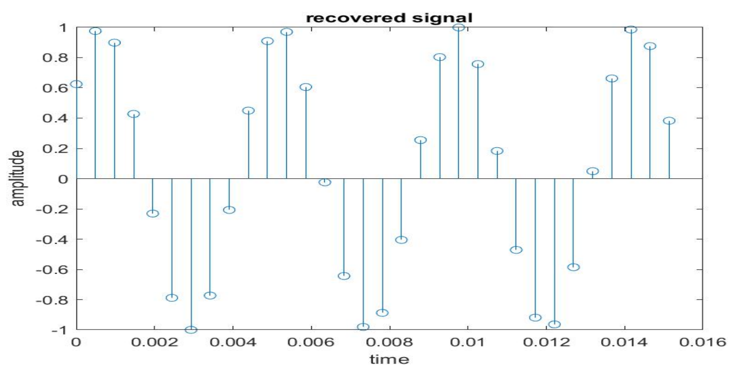

Figure 6:The resulting time-domain sequence

1.8. Create a 512-length sequence

Create a 512-length sequence of the sinusoid in equation (3) at a frequency of 220 Hz, amplitude of 1, and a sampling rate of 2,048 Hz.

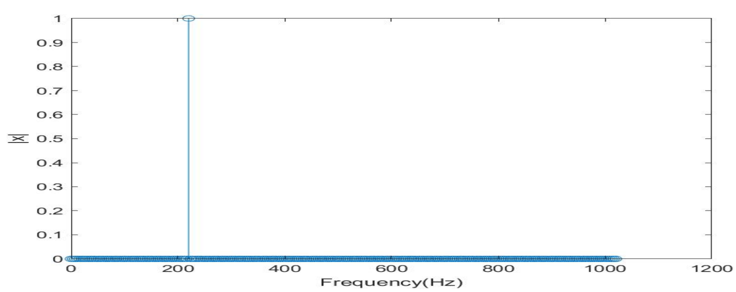

Figure 7: The magnitude of the result versus frequency(512-length sequence)

2.1.Sample the sinusoid in different frequencies

The code is as follows:

```
function exe2_1()
N = [100,200,300,400,500,600,700,800,900,1000];
fs=1000;
```

```
for i = 1:10
for n = 1 : 128
x(n) = sin(2 * pi * n * N(i)./1000);
end
y = x;
subplot(4,3,i), stem([0:127],y);
title(i)
end
end
```

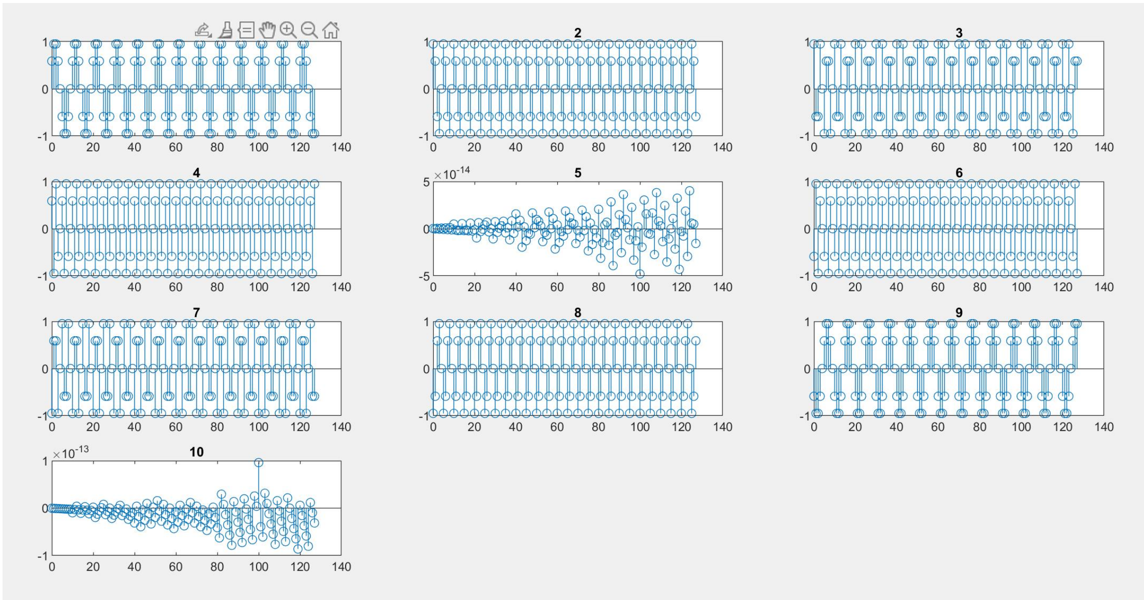

Figure 8: The sampling in different f

2.2 Observation from the plot

After observation, it seems that most dots in each plot will fall on four x-parallel lines, and it is symmetric(except 500Hz and 1000Hz).It is hard to tell the difference between the 400 and 600Hz. When the f=1000Hz, the plot tends to be divergent and can hardly show show what the original frequency is.

2.3Evalute the DFT in 2.1

The code of the normalized magnitude spectrum is as follows:

```
function exe2_3()
N = [100,200,300,400,500,600,700,800,900,1000];
fs=1000;
for i = 1:10
```

```
for n = 1 : 128
x(n) = sin(2 * pi * n * N(i)./1000);
end
y = fft(x,N(i));
mag=abs(y);
a = 0;
for j = 1 : length(mag)
if a < mag(j)
a = mag(j);
end
end
mag = mag./a;
f=(0:N(i)-1)*fs/N(i);
subplot(4,3,i), stem(f(1:N(i)/2),mag(1:N(i)/2)*2/N(i));
title(i);xlabel('Frequency(Hz)');ylabel('|x|');
end
end
```

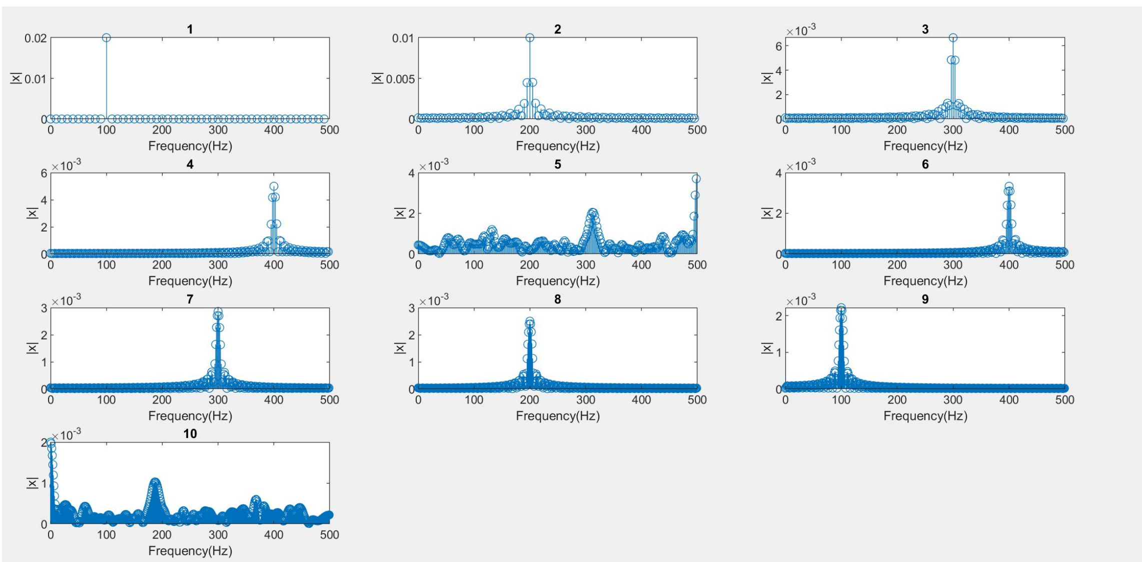

Figure 9: The normalized magnitude spectrum in different frequencies

2.4 Answer for questions

After observing these plots, I notice that when the frequency is small(100Hz or 200Hz), it is easy to tell what is the original frequency from the plot. And the higher frequencies is, the higher density of the dots spread and harder to tell the original frequency. When zoom out the plots, the difference between 400 and 600 Hz can be detected, but the difference is not so much. 300Hz is most of energy located for the sampled sinusoid have f = 700.

2.5 Run the program given

The code and comments are as follow:

```
function exe2_5()
clear all
sndfile = 'speech_female.wav';%load the wav file
[x,Fs] = audioread(sndfile);%reads data from the wav, and returns sampled data, x, and a sample rate for that data, Fs.
N = 512;%sample numbers
[S,F,T] = spectrogram(x(1:Fs*1.4),N,3*N/4,N*4,Fs);%Spectrogram using short-time Fourier transform.It returns the spectrogram at the cyclical frequencies specified in f
f = figure('Position',[500 300 700 500],'MenuBar','none', ...%create a figure window to show image
'Units','Normalized');
set(f,'PaperPosition',[0.25 1.5 8 5]);
axes('FontSize',14);%specifies the limits for the current axes.
imagesc(T,F./1000,20*log10(abs(S)));% Display the data in array as an image that uses the range of colors in the colormap.
axis xy;
set(gca,'YTick',[0:2000:Fs/2]./1000, ...
'YTickLabel',[0:2000:Fs/2]./1000);%Set graphics object properties
ylabel('Frequency (kHz)');% add y axis label
xlabel('Time (s)');%add x axis label
end
```

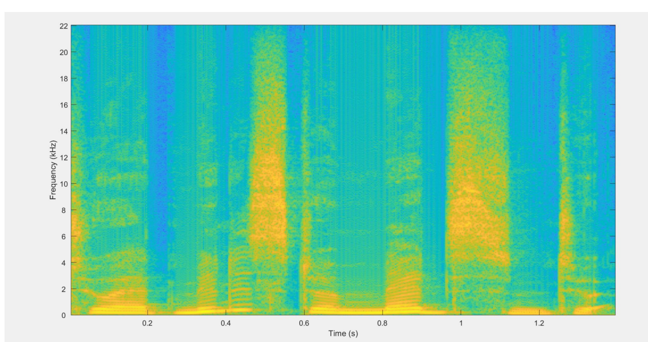

Figure 10: The log magnitude of the STFT

2.5 Answer for the questions(2.6 and 2.7 in Lab)

From the spectrum in 2.5, the frequency from 0Hz to 18kHz. Based on the the Nyquist-Shannon Sampling Theorem, if the signal contain no frequency higher than 18Hz, we can set the sample fate at 40 kHz making 18 <40/2 to help perfect reconstruction.

After observing the distribution of energy across frequencies in the spectrogram, the 'minister' in administer and the 'dicine' in medicine represent the first and second sound that correspond to the energy between 4 and 18kHz respectively.

2. Conclusion

The moving average can be not only used in 1-D data processing. But we need to be careful when the M becomes too big since some details that we want may also me averaged. The

moving average can also be extended to 2-D image processing as well.

In Lab2, we know that we should be really careful when choosing the sample rate, or it will cause having many unwanted frequencies after FFT. In order to fix this problem, taking more sample is also a way to help get the real frequency after FFT. In the last section of the lab, we also learned how the STFT works in processing the sound file.

3. Acknowledgments

Thanks for the TA gives us the illustration on the basic principle of how sampling works and the use of the code in 2.5 at the section.