Phys 5053 -- Assignment 5 - Solutions

1. Generate, numerically, a sine wave y =10*sin ($2\pi\, t/64$) for 0<t<256. Using a numerical code of your choosing, compute the Fourier transform and show that you can identify the frequency of y(t).

In this simple example, the equation specifies the frequency explicitly: $\omega=2\pi/64$. The period of the signal is therefore T=64, and the frequency $f=1/T=\omega/2\pi=1/64=0.0156$. (If time is in seconds, then f is in Hertz (1/s), and the angular frequency ω is in radians per second.) So the Fourier analysis should produce a peak at f=0.0156.

```
t=0:0.01:256;                 % define the time vector. Notice that dt=0.01.
y=10*sin(2*pi*t/64);          % define the time series
plot(t,y)
```

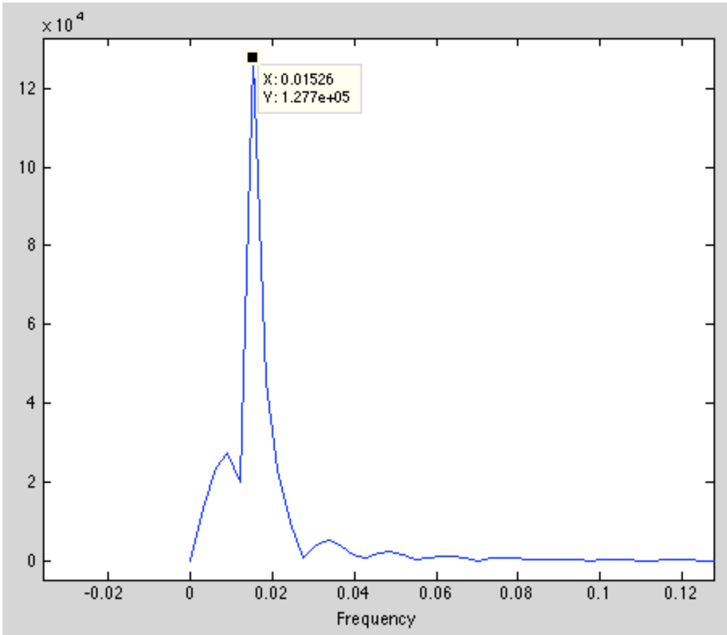
Here is a sample code that performs the Fourier transform (slightly modified from 'SupFourier.m', which is posted online):

```
dt=0.01;                      % define dt to be used for rescaling the time axis
L = max(size(y));             % Length of signal

NFFT = 2^nextpow2(L);         % Next power of 2 from length of y
                              % This is the length of series that fft analyzes
Y = fft(y,NFFT);              % run fft to obtain the Fourier components Y

f = 1/2*linspace(0,1,NFFT/2);
```

The last command defines the frequency vector that corresponds to data in Y. Note that so far Matlab knows nothing about your time units. All it sees is that your series has 256/0.01=25600 points. So the frequency f is expressed in units of 1/(0.01) , and will need to be rescaled on output:

```
plot(f(1:NFFT/2)/dt,abs(Y(1:NFFT/2)))
xlabel('Frequency')
```

Note the dt factor, which brings the frequency axis to the correct units. The plot shows the Fourier amplitudes Y. These are complex numbers, so to plot them you need to take the absolute value (that's what the 'abs' command does). To be precise, the Fourier *power* spectrum is $|Y|^2$, but this affects only the vertical scale and does not affect the frequency. You need to zoom in on the graph to see the peak at around 0.0156.



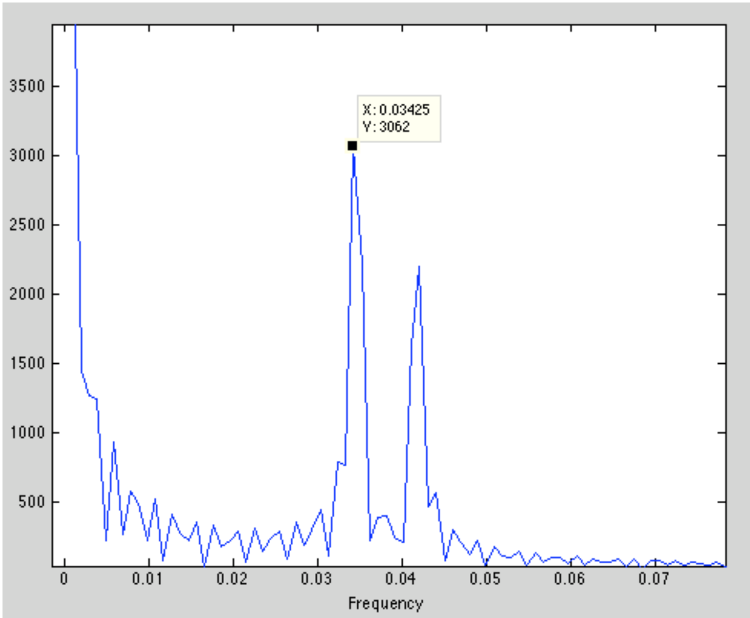
The alternative representation in terms of the characteristic period, rather than frequency, can be obtained by

```
plot(1./f(1:NFFT/2)*dt,abs(Y(1:NFFT/2)))
xlabel('period')
```

Note the way factor dt now enters the equation. The produced graph has a peak at 65.5. It is not exactly at 64 because of the aliasing (there are not enough sampling points to define a peak at 64), and because the original signal has a finite length, which causes the fft to generate extraneous low-frequency components.

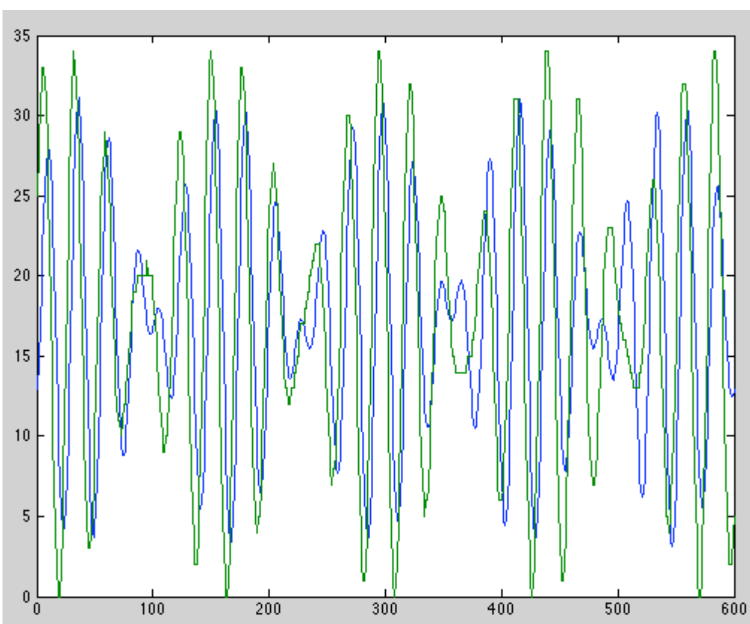
2. The file ‘Star.txt’ contains brightness records for a variable star on 600 successive midnights. Find any characteristic periods (or frequencies) in the signal. Compare the results of autocorrelation analysis with the results of Fourier analysis.

```
L=max(size(Star));
dt=1;
t=1:dt:L;.
y=Star;
NFFT = 2^nextpow2(L);
Y = fft(y,NFFT);                % run fft to obtain the Fourier components Y
f = 1/2*linspace(0,1,NFFT/2);
plot(f(1:NFFT/2)/dt,abs(Y(1:NFFT/2)))
xlabel('Frequency')
```



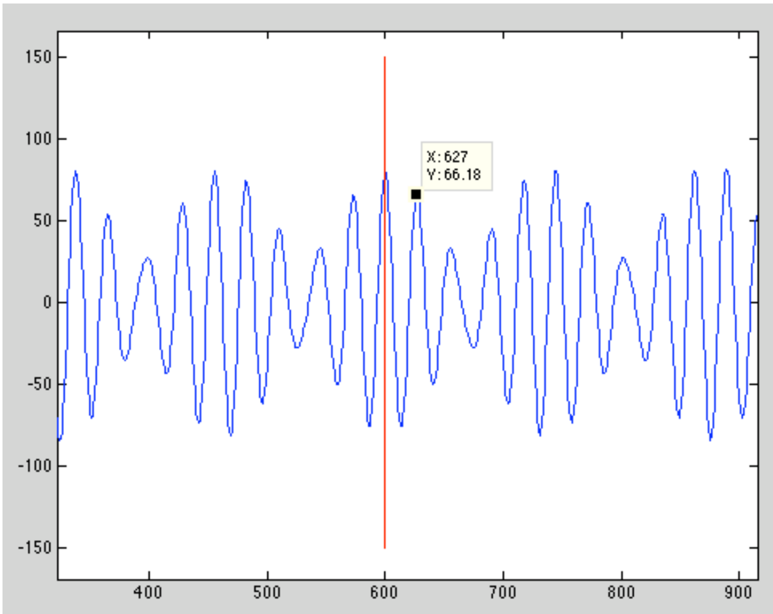
The Fourier analysis indicates two frequencies: 0.03425 and 0.04207. A quick check can help visualize whether these frequencies are correct. Just construct a series g(t) from two sine waves of these frequencies (the amplitudes and phases of these waves would have to be found by trial and error, but the important part are the frequencies):

```
om1=0.03425*2*pi;
om2=0.04207*2*pi;
g=sin(om1*(t+30))+sin(om2*(t-30));
plot(t,7*g+mean(y),t,y)
```



The autocorrelation function can be calculated using either the xcorr or xcov command:

```
c=xcov(y,y,'unbiased'); plot(c)
hold on
mc=150;                  % to plot a center line
plot([0.5*(2*L-1)  0.5*(2*L-1)] , [-mc  mc], 'r')
hold off
```



The period corresponds to 27 units (the center line is at 600), which gives the frequency 1/27=0.037. This is obviously in-between the two frequencies that we determined with Fourier. The autocorrelation function thus does not directly resolve the two frequencies that are close together.

Note: Going beyond the material that was covered in this course, one can see that there are two frequencies from the fact that both the original function and the autocorrelation function exhibit a “beat” pattern. This pattern results from two similar frequencies. The frequencies can be found if one recalls that the period of the beat (about 150 in this case) is related to the frequencies in the signal: $f_{beat} = (f_1-f_2)/2$, and the observed periodicity of the signal $(1/27) = (f_1+f_2)/2$.