Task1-2: Signal Processing II

Name: Steven Chuo **Degree:** MA **ID:** 20522068

Problem 1: Convolution

$$f(t) = g(t) = \begin{Bmatrix} A, & a > |t| \\ 0, & otherwise \end{Bmatrix}$$

(a) Compute an analytic y(t) which is the convolution of f(t) and g(t):

$$f(\tau) = \begin{cases} A, & -a < \tau < a \\ 0, & \text{otherwise} \end{cases}$$

$$g(t-\tau) = \begin{cases} A, & t-a < t-\tau < t+a \\ 0, & \text{otherwise} \end{cases}$$

$$y(t) = f(t) * g(t) = \int_{-\infty}^{\infty} f(\tau)g(t-\tau)d\tau$$

$$\text{if t} < 0: \int_{-\infty}^{t-a} f(\tau)g(t-\tau)d\tau + \int_{t-a}^{-a} f(\tau)g(t-\tau)d\tau + \int_{-a}^{t+a} f(\tau)g(t-\tau)d\tau + \int_{t+a}^{a} f(\tau)g(t-\tau)d\tau + \int_{a}^{\infty} f(\tau)g(t-\tau)d\tau$$

$$\text{if t} < 0: \int_{-a}^{t+a} A \cdot A d\tau = A^2 \tau|_{-a}^{t+a} = A(t+a-(-a)) = A(t+2a) = At + 2aA$$

$$\text{if t} > 0: \int_{-\infty}^{-a} f(\tau)g(t-\tau)d\tau + \int_{-a}^{t-a} f(\tau)g(t-\tau)d\tau + \int_{t-a}^{a} f(\tau)g(t-\tau)d\tau + \int_{a}^{t+a} f(\tau)g(t-\tau)d\tau + \int_{t+a}^{\infty} f(\tau)g(t-\tau)d\tau$$

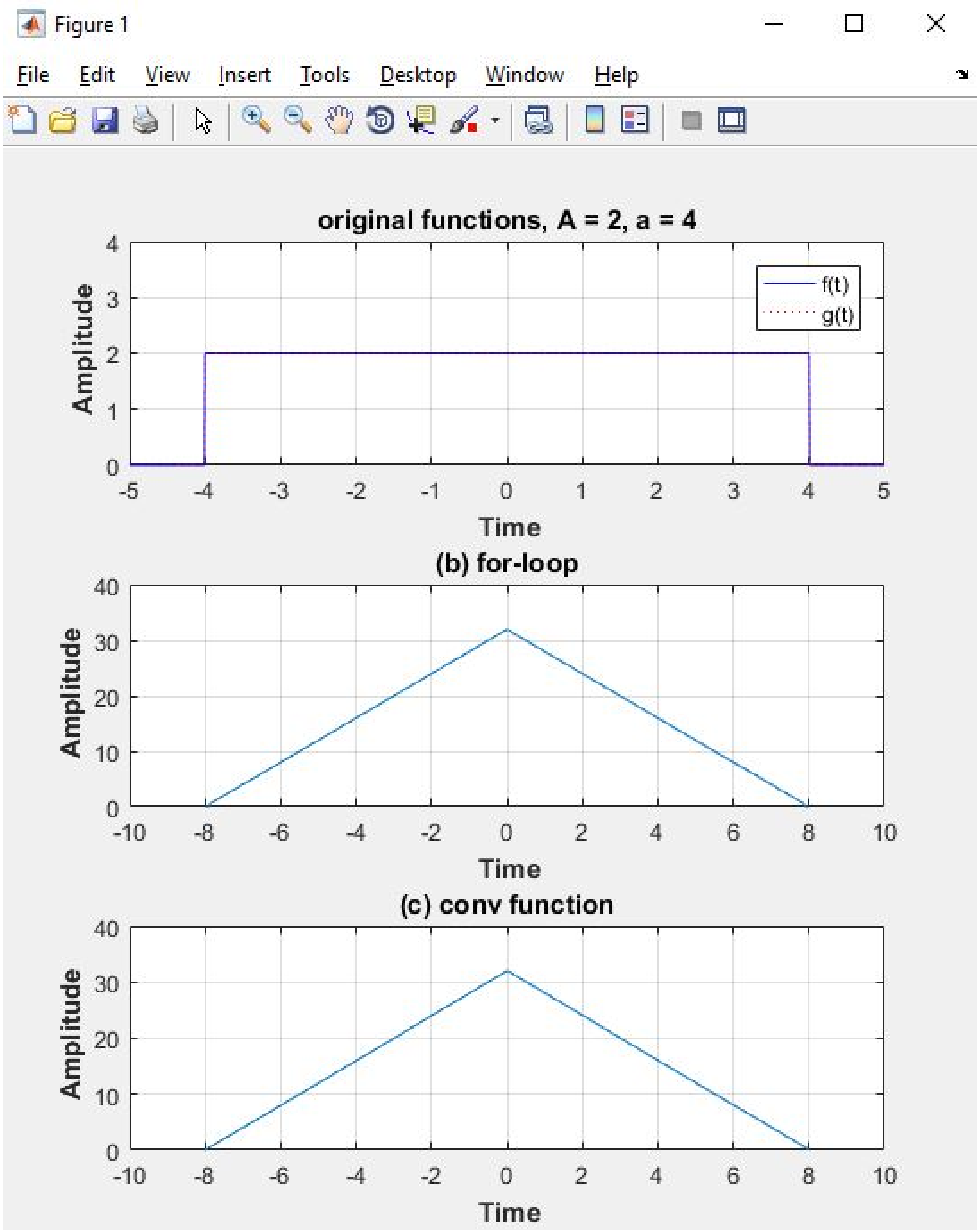
$$\text{if t} > 0: \int_{t-a}^{a} A \cdot A d\tau = A^2 \tau|_{t-a}^{a} = A(a-(t-a)) = A(-t+2a) = -At + 2aA$$

$$y(t) = f(t) * g(t) = \begin{cases} At + 2aA, & -2a < t < 0 \\ -At + 2aA, & 0 > t > 2a \\ 0, & \text{otherwise} \end{cases}$$

(b) Write a code to numerically compute y(t) and plot y(t). Please use for-loop and do not use `conv`. You can randomly assign the values of A and a.

```
clc, clear, format short, format compact, close all

syms t tau

A = 2; a = 4;
f = @(tau) A*(abs(tau)<=a);
```

```
g = @(t, tau) A*(abs(t-tau)<=a);

t = (-a*2):0.01:(a*2);

% (b) numerical convolution with for-loop
y_for = zeros(1, numel(t));
y_fun = @(t) integral(@(tau) f(tau)*g(t, tau), -inf, inf,...
        'ArrayValued', true);
count = 1;
for i=1:numel(t)
    y_for(count) = y_fun(t(i));
    count = count+1;
end

% (c) numerical convolution with conv function
A = 2; a = 4;
f = @(t) A*(abs(t)<=a);
g = @(t) A*(abs(t)<=a);
y_conv = conv(f(t), g(t))*0.01;

figure
subplot(311); plot(t,f(t),'-b'); hold on;
plot(t,g(t),':r'); title('original functions');
legend('f(t)', 'g(t)'); axis tight; grid on;
ylabel('\bf Amplitude'); ylim([0 4]);
xlabel('\bf Time'); xlim([-5 5])

subplot(312); plot(t,y_for)
title('for-loop'); axis tight; grid on;
ylabel('\bf Amplitude'); ylim([0 40]);
xlabel('\bf Time'); xlim([-10 10])

subplot(313); plot(t,y_conv((floor(numel(y_conv)/4)):floor(numel(y_conv)/4)*3))
title('conv function'); axis tight; grid on;
ylabel('\bf Amplitude'); ylim([0 40]);
xlabel('\bf Time'); xlim([-10 10])
```



(c) Write a code to numerically compute y(t) and plot y(t). Please use `conv`. You can randomly assign the values of A and a.

Refer to (b) to the code and plot.

Problem 2: Convolution Theorem

(a) Proof the convolution theorem and explain the meaning of these relationships in your words.

$$F\{x(t) * h(t)\} = X(f) \cdot H(f)$$
$$F\{x(t) \cdot h(t)\} = X(f) * H(f)$$

$$F(x(t) * h(t)) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x(\tau)h(t-\tau) \exp^{-i2\pi ft} \, d\tau dt = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x(\tau)h(v) \exp^{-i2\pi f(\tau+v)} \, d\tau dv$$

$$= \int_{-\infty}^{\infty} x(\tau) \exp^{-i2\pi f(\tau)} \, d\tau \int_{-\infty}^{\infty} h(v) \exp^{-i2\pi f(v)} \, dv = X(f)H(f)$$

The Fourier transformation for convolution of two functions is equal to the product of the Fourier transforms of those two functions.

$$F(x(t) \cdot h(t)) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} X(f_1) \exp^{i2\pi f_1 t} H(f_2) \exp^{i2\pi f_2 t} \exp^{-i2\pi ft} \, df_1 df_2 dt$$

$$= \int_{-\infty}^{\infty} X(f_1) \int_{-\infty}^{\infty} H(f_2) \int_{-\infty}^{\infty} \exp^{-i2\pi(f-f_1-f_2)t} \, dt df_2 df_1 = \int_{-\infty}^{\infty} X(f_1) \int_{-\infty}^{\infty} H(f_2)\delta(f-f_1-f_2)t df_2 df_1$$

$$= \int_{-\infty}^{\infty} X(f_1)H(f-f_1)df_1 = X(f) * H(f)$$

The Fourier transformation for the product of two functions is equal to the convolution of the Fourier transforms of those two functions.

(b) Compute a Fourier transform of the triangular function in both analytic and numeric ways (Note that this function is not a periodic):

$$x(t) = \begin{cases} 1 - |t| & |t| < 1 \\ 0 & |t| > 1 \end{cases}$$

From Problem 1, it was observed that the triangular function appears to be the convolution of two box functions, which could be expressed as:

$$f(t) = \begin{cases} 1, & |t| \leq 0.5 \\ 0, & \text{otherwise} \end{cases}$$

The Fourier transformation for convolution of two functions is equal to the product of the Fourier transforms of those two functions. Therefore the Fourier transform of the triangular function should be the product of the Fourier transforms of the two box functions.

Compute the Fourier transform of the box function:

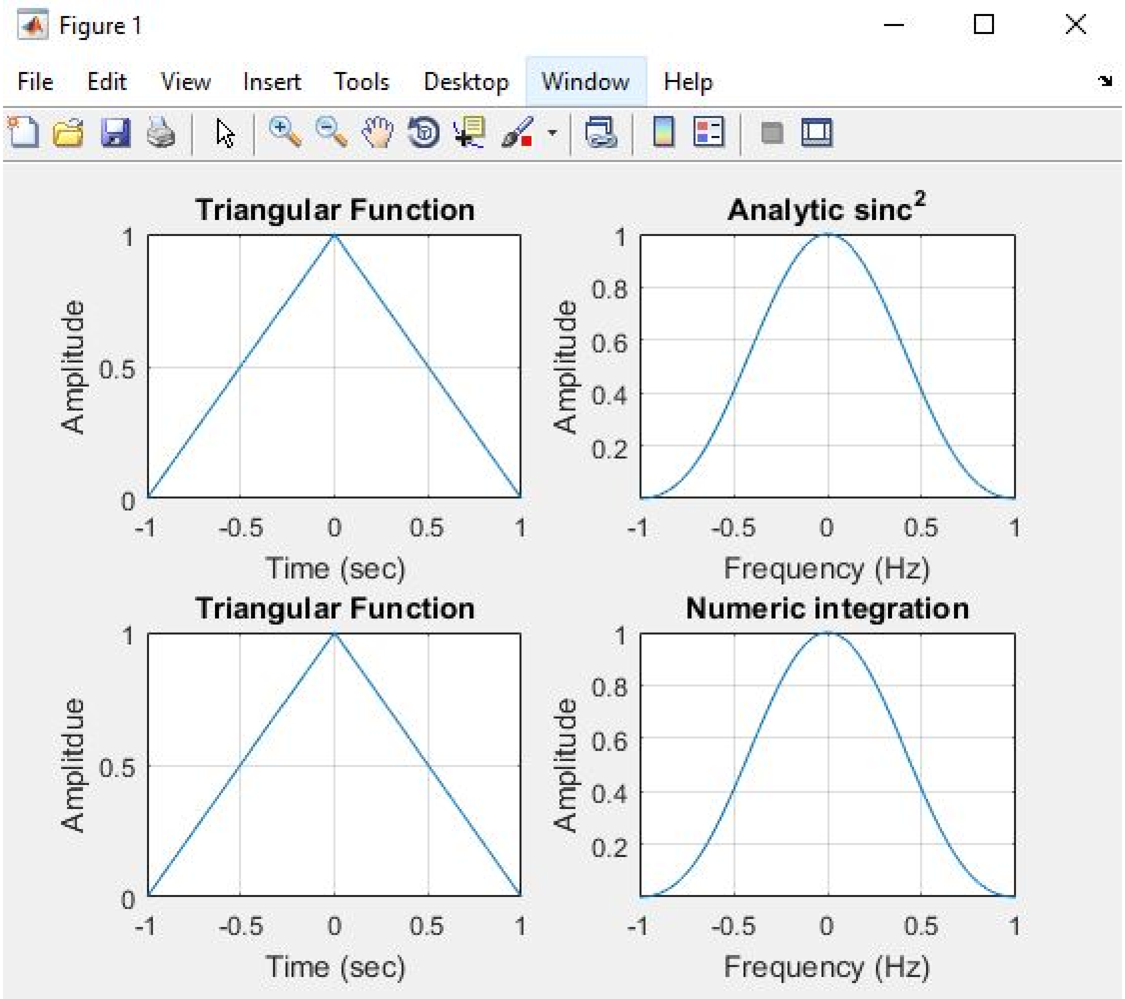
$$F(f) = \int_{-\infty}^{\infty} f(t) \exp^{-2\pi ift} \, dt = \int_{-0.5}^{0.5} 1 \cdot \exp^{-2\pi ift} \, dt = \frac{1}{-2\pi if} \exp^{-2\pi ift} \Big|_{-0.5}^{0.5}$$

$$F(f) = \frac{1}{-2\pi if}(\exp^{-\pi if} - \exp^{\pi if}) = \frac{1}{\pi f} \frac{(\exp^{\pi if} - \exp^{-\pi if})}{2i} = \frac{\sin(\pi f)}{\pi f} = \sin c(f)$$

The Fourier transform of a box function is the sinc function, therefore the Fourier transform of the triangular function can be computed as follows:

$$X(x(t)) = X(f(t) * f(t)) = F(f)F(f) = \sin c(f) \cdot \sin c(f) = \sin c^2(f)$$

```
clc, clear, format short, format compact, close all

% (b) analytic
t = -1:1/1000:1;
f = -1:1/1000:1;
x = @(t) (1-abs(t)).*(abs(t)<1);
X_anal = @(t) sinc(t).*sinc(t);

figure
subplot(221); plot(t,x(t))
axis tight; grid on; ylim([0 1])
ylabel('Amplitude'); xlabel('Time (sec)');
title('Triangular Function')

subplot(222); plot(t,X_anal(t));
axis tight; grid on;
ylabel('Amplitude'); xlabel('Frequency (Hz)');
title('Analytic sinc^2')

% (b) numerical
x_int = @(t,f) x(t)*exp(sqrt(-1)*2*pi*t*f);
X_num = @(f) integral(@(t) x_int(t,f), -inf, inf, 'ArrayValued', true);

subplot(223); plot(t,x(t));
axis tight; grid on;
ylabel('Amplitdue'); xlabel('Time (sec)');
title('Triangular Function')

subplot(224); plot(f, X_num(f));
axis tight; grid on;
ylabel('Amplitude'); xlabel('Frequency (Hz)');
title('Numeric integration')
```



(c) Please explain the result in (b) using your answers for Problem 1.

Refer to (b) for the relationship between Fourier transformation of a function (triangular function) and its convolution (two box functions).

Problem 3: Discrete Fourier Transform 1

(a) What is the meaning of the following relationship in the lecture slide? Please explain it.

$$X_s(f + r/\Delta) = \sum_{n=-\infty}^{\infty} x(n\Delta)e^{-i2\pi(f+r/\Delta)n\Delta} = \sum_{n=-\infty}^{\infty} x(n\Delta)e^{-i2\pi f n\Delta - i2\pi r n} = \sum_{n=-\infty}^{\infty} x(n\Delta)e^{-i2\pi f n\Delta} = X_s(f)$$

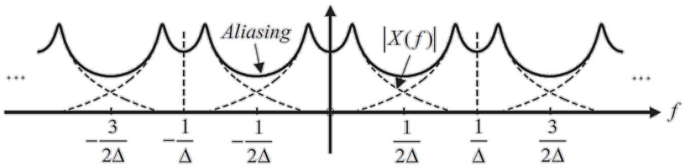
The Fourier transform of a uniformly sampled discrete sequence for a continuous signal is periodic. For any integer value of ***r***, the frequency of the signal is the same and repeated every period of $1/\Delta$.

(b) What is the meaning of the following relationship in the lecture slide? Please explain it.

Fourier Transform of a discrete sequence, $x_s(t)$

$$X_s(f) = I(f) * X(f) = \int_{-\infty}^{\infty} I(g)X(f-g)dg = \int_{-\infty}^{\infty} \frac{1}{\Delta}\sum_{n=-\infty}^{\infty} \delta\left(g-\frac{n}{\Delta}\right)X(f-g)dg = \frac{1}{\Delta}\sum_{n=-\infty}^{\infty}\int_{-\infty}^{\infty} \delta\left(g-\frac{n}{\Delta}\right)X(f-g)dg$$

$$= \frac{1}{\Delta}\sum_{n=-\infty}^{\infty} X\left(f-\frac{n}{\Delta}\right) = \frac{1}{\Delta}\left(\dots + X\left(f-\frac{2}{\Delta}\right) + X\left(f-\frac{1}{\Delta}\right) + X(f) + X\left(f+\frac{1}{\Delta}\right) + \dots\right)$$



The Fourier transform of a discrete periodic sequence can be expressed in terms of the summation of the shifted Fourier transform of a continuous signal. Sampling in the time domain results in periodic and continuous function in the frequency domain. However, aliasing occurs when sampling too slowly, such that the shifted frequency overlap with each other, losing information of the higher frequency components.

(c) What is the difference between these two functions in the lecture slide: $X_s(f)$ and $X(k)$

Even though both are Fourier transform of the signal, the difference between the two is the way the signal is being sampled. While $X_s(f)$ is a continuous function of frequency from negative infinity to positive infinity, $X(k)$ is the sampled version, with finite length sequence and periodic equal spacing in between.

Problem 4: Discrete Fourier Transform 3 - Use FFT

$$y_1(t) = e^{-a|t|}(b \cdot cos2\pi f_1 t + c \cdot cos2\pi f_2 t)$$

where a = 2, b = 2, c = 6, f1 = 3, and f2 = 6

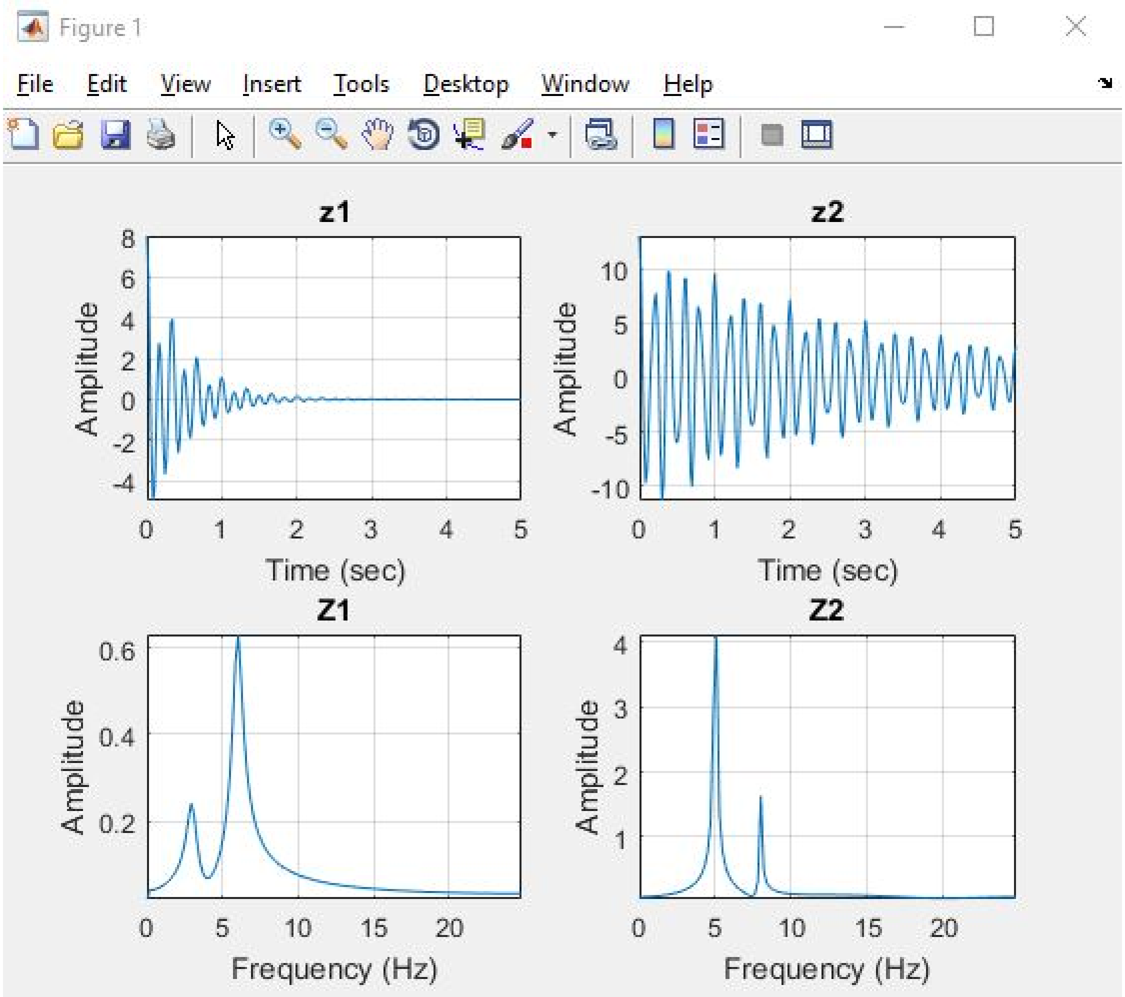
$$y_2(t) = e^{-a|t|}(b \cdot cos2\pi f_1 t + c \cdot cos2\pi f_2 t)$$

where a = 0.3, b = 10, c = 3, f1 = 5, and f2 = 8

(a) z1 and z2 are discrete signals, which are obtained by digitizing y1(t) and y2(t) with a sampling rate of 50 Hz and collecting them for 5 seconds, respectively. Please plot z1 and z2 in the time domain (include a proper time axis).

```
clc, clear, format short, format compact, close all

% (a) time domain
time = 5;
fs = 50;
t = 0:1/fs:time;

a1 = 2;
b1 = 2;
c1 = 6;
```

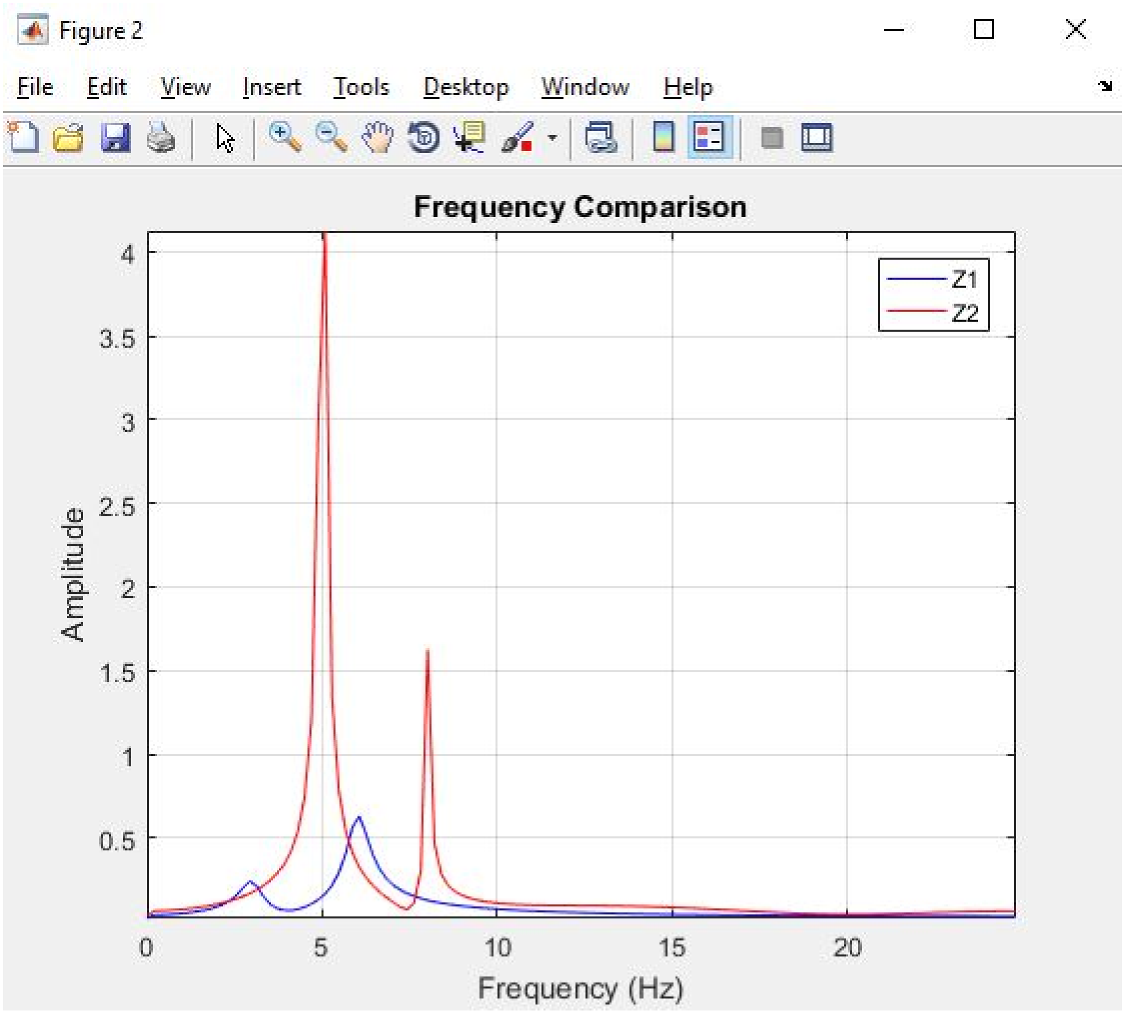
```
f11 = 3;
f21 = 6;
y1 = exp(-a1*abs(t)).*(b1*cos(2*pi*f11*t)+c1*cos(2*pi*f21*t));

a2 = 0.3;
b2 = 10;
c2 = 3;
f12 = 5;
f22 = 8;
y2 = exp(-a2*abs(t)).*(b2*cos(2*pi*f12*t)+c2*cos(2*pi*f22*t));

figure
subplot(221); plot(t,y1)
axis tight; grid on;
ylabel('Amplitude'); xlabel('Time (sec)');
title('z1')

subplot(222); plot(t,y2);
axis tight; grid on;
ylabel('Amplitude'); xlabel('Time (sec)');
title('z2')

% (b) frequency domain
L = length(y1);
nfft = 2^nextpow2(L);

Y1 = fft(y1, nfft);
P2_1 = abs(Y1/L);
P1_1 = P2_1(1:nfft/2+1);
P1_1(2:end-1) = 2*P1_1(2:end-1);

Y2 = fft(y2, nfft);
P2_2 = abs(Y2/L);
P1_2 = P2_2(1:nfft/2+1);
P1_2(2:end-1) = 2*P1_2(2:end-1);

subplot(223), plot(0:(fs/nfft):(fs/2-fs/nfft),P1_1(1:nfft/2));
axis tight, grid on;
ylabel('Amplitude'); xlabel('Frequency (Hz)');
title('Z1')

subplot(224), plot(0:(fs/nfft):(fs/2-fs/nfft),P1_2(1:nfft/2));
axis tight, grid on;
ylabel('Amplitude'); xlabel('Frequency (Hz)');
title('Z2')

% (c) frequency comparison
figure
plot(0:(fs/nfft):(fs/2-fs/nfft),P1_1(1:nfft/2), '-b'); hold on;
plot(0:(fs/nfft):(fs/2-fs/nfft),P1_2(1:nfft/2), '-r');
axis tight, grid on; legend('Z1', 'Z2');
ylabel('Amplitude'); xlabel('Frequency (Hz)');
title('Frequency Comparison')
```



(b) Perform the discrete Fourier transform of z1 and z2, and plot your graphs in the frequency domain (include a proper frequency axis). Plot only positive frequency signals.

Refer to (a) to the code and plot.

(c) Please compare the shape of the frequency curves of z1 and z2. Which frequency curve is thinner (more narrow)? For example, compare the frequency curve at f1 in both graphs. Which one is thinner? Please explain your answer. What makes the difference?



The frequency curve of z2 appears to be thinner. There are two potential reasons: 1) the value of *a* is lower for z2 than z1, which increases the amplitude of the frequency, and 2) the value of *b* is higher for z2 than z1, which also increases the amplitude of the frequency. The result of the combination of both coefficients is what appears to be a narrower frequency curve after Fourier transform.

Problem 5: Discrete Fourier Transform 2 - Use FFT

$$y(t) = A1sin2\pi(25)t + A2sin2\pi(75)t + A3sin2\pi(125)t$$

where A1 = 3, A2 = 10, and A3 = 5

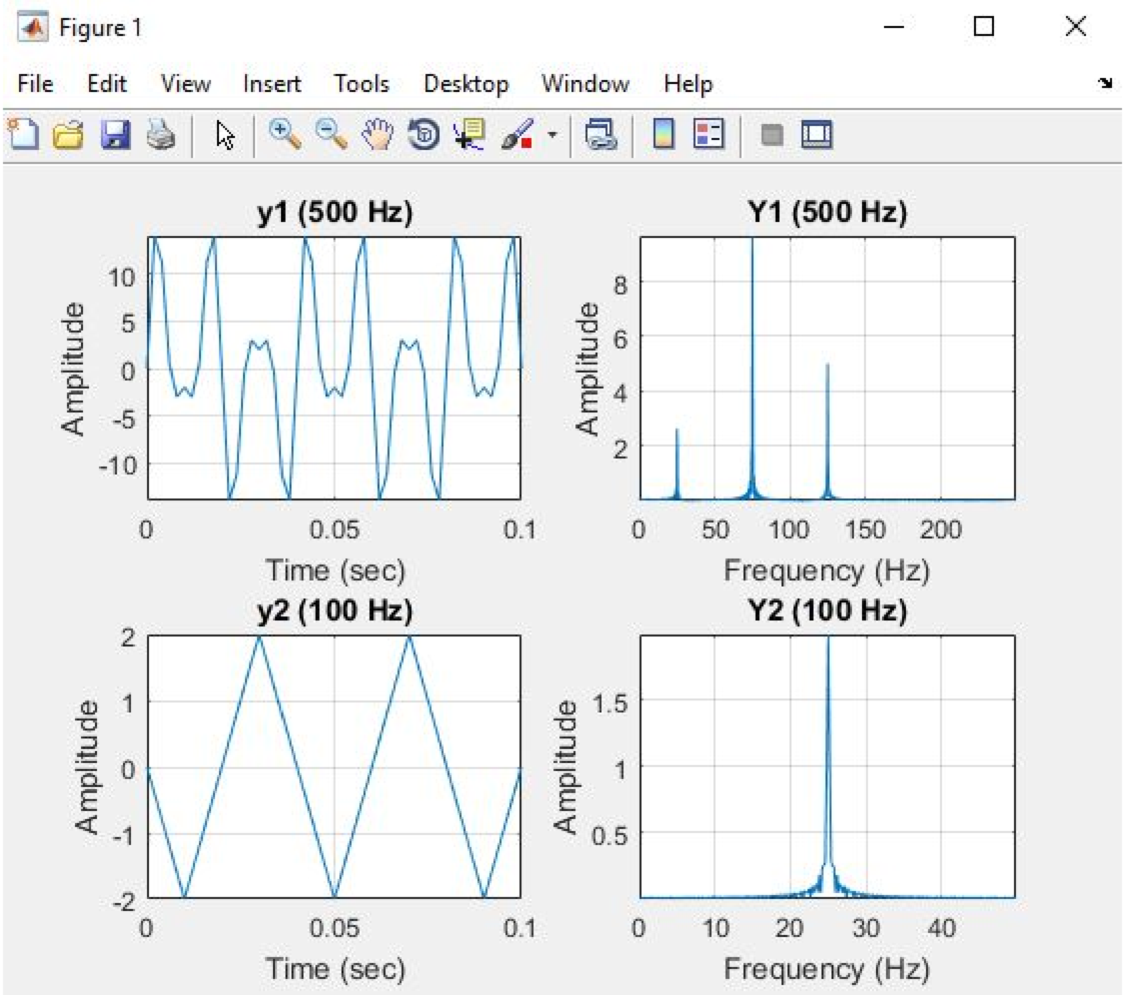
(a) y1 is a discrete signal, which is obtained by digitizing y(t) with a sampling rate of 500 Hz for 3 seconds. Please plot y1 in the time domain (include a proper time axis).

```
clc, clear, format short, format compact, close all

% (a) y1 time domain
time = 3;
fs1 = 500;
t1 = 0:1/fs1:time;
```

```
a1 = 3;
a2 = 10;
a3 = 5;
y1 = a1*sin(2*pi*25*t1) + a2*sin(2*pi*75*t1) + a3*sin(2*pi*125*t1);

figure
subplot(221); plot(t1,y1)
axis tight; grid on;
ylabel('Amplitude'); xlabel('Time (sec)');
xlim([0 0.1]); title('y1 (500 Hz)')

% (b) Y1 frequency domain
L = length(y1);
nfft = 2^nextpow2(L);

Y1 = fft(y1, nfft);
P2_1 = abs(Y1/L);
P1_1 = P2_1(1:nfft/2+1);
P1_1(2:end-1) = 2*P1_1(2:end-1);

subplot(222), plot(0:(fs1/nfft):(fs1/2-fs1/nfft),P1_1(1:nfft/2));
axis tight, grid on;
ylabel('Amplitude'); xlabel('Frequency (Hz)');
title('Y1 (500 Hz)')

% (c) y2 time domain
fs2 = 100;
t2 = 0:1/fs2:3;

y2 = a1*sin(2*pi*25*t2) + a2*sin(2*pi*75*t2) + a3*sin(2*pi*125*t2);

subplot(223); plot(t2,y2)
axis tight; grid on;
ylabel('Amplitude'); xlabel('Time (sec)');
xlim([0 0.1]); title('y2 (100 Hz)')

% (d) Y2 frequency domain
L2 = length(y2);
nfft2 = 2^nextpow2(L2);

Y2 = fft(y2, nfft2);
P2_2 = abs(Y2/L2);
P1_2 = P2_2(1:nfft2/2+1);
P1_2(2:end-1) = 2*P1_2(2:end-1);

subplot(224), plot(0:(fs2/nfft2):(fs2/2-fs2/nfft2),P1_2(1:nfft2/2));
axis tight, grid on;
ylabel('Amplitude'); xlabel('Frequency (Hz)');
title('Y2 (100 Hz)')

% (e) y sampled 100 Hz for 20 seconds
fs3 = 100;
t3 = 0:1/fs2:20;
```

```
y3 = a1*sin(2*pi*25*t3) + a2*sin(2*pi*75*t3) + a3*sin(2*pi*125*t3);

figure
subplot(221); plot(t1,y1)
axis tight; grid on;
ylabel('Amplitude'); xlabel('Time (sec)');
xlim([0 0.1]); title('y1 (500 Hz, 3s)')

subplot(223); plot(t3,y3)
axis tight; grid on;
ylabel('Amplitude'); xlabel('Time (sec)');
xlim([0 0.1]); title('y (100 Hz, 20s)')

L3 = length(y3);
nfft3 = 2^nextpow2(L3);

Y3 = fft(y3, nfft3);
P2_3 = abs(Y3/L3);
P1_3 = P2_3(1:nfft3/2+1);
P1_3(2:end-1) = 2*P1_3(2:end-1);

subplot(222), plot(0:(fs1/nfft):(fs1/2-fs1/nfft),P1_1(1:nfft/2));
axis tight, grid on;
ylabel('Amplitude'); xlabel('Frequency (Hz)');
title('Y1 (500 Hz, 3s)')

subplot(224), plot(0:(fs3/nfft3):(fs3/2-fs3/nfft3),P1_3(1:nfft3/2));
axis tight, grid on;
ylabel('Amplitude'); xlabel('Frequency (Hz)');
title('Y  (100 Hz, 20s)')

% (f) y sampled 105 Hz for 3 seconds
fs4 = 105;
t4 = 0:1/fs4:3;

y4 = a1*sin(2*pi*25*t4) + a2*sin(2*pi*75*t4) + a3*sin(2*pi*125*t4);

figure
subplot(221); plot(t1,y1)
axis tight; grid on;
ylabel('Amplitude'); xlabel('Time (sec)');
xlim([0 0.1]); title('y1 (500 Hz, 3s)')

subplot(223); plot(t4,y4)
axis tight; grid on;
ylabel('Amplitude'); xlabel('Time (sec)');
xlim([0 0.1]); title('y  (105 Hz, 3s)')

L4 = length(y4);
nfft4 = 2^nextpow2(L4);

Y4 = fft(y4, nfft4);
```

```
P2_4 = abs(Y4/L4);
P1_4 = P2_4(1:nfft4/2+1);
P1_4(2:end-1) = 2*P1_4(2:end-1);

subplot(222), plot(0:(fs1/nfft):(fs1/2-fs1/nfft),P1_1(1:nfft/2));
axis tight, grid on;
ylabel('Amplitude'); xlabel('Frequency (Hz)');
title('Y1 (500 Hz, 3s)')

subplot(224), plot(0:(fs4/nfft4):(fs4/2-fs4/nfft4),P1_4(1:nfft4/2));
axis tight, grid on;
ylabel('Amplitude'); xlabel('Frequency (Hz)');
title('Y (105 Hz, 3s)')

% (g) y sampled 251 Hz for 3 seconds
fs5 = 251;
t5 = 0:1/fs5:3;

y5 = a1*sin(2*pi*25*t5) + a2*sin(2*pi*75*t5) + a3*sin(2*pi*125*t5);

figure
subplot(221); plot(t1,y1)
axis tight; grid on;
ylabel('Amplitude'); xlabel('Time (sec)');
xlim([0 0.1]); title('y1 (500 Hz, 3s)')

subplot(223); plot(t5,y5)
axis tight; grid on;
ylabel('Amplitude'); xlabel('Time (sec)');
xlim([0 0.1]); title('y  (251 Hz, 3s)')

L5 = length(y5);
nfft5 = 2^nextpow2(L5);

Y5 = fft(y5, nfft5);
P2_5 = abs(Y5/L5);
P1_5 = P2_5(1:nfft5/2+1);
P1_5(2:end-1) = 2*P1_5(2:end-1);

subplot(222), plot(0:(fs1/nfft):(fs1/2-fs1/nfft),P1_1(1:nfft/2));
axis tight, grid on;
ylabel('Amplitude'); xlabel('Frequency (Hz)');
title('Y1 (500 Hz, 3s)')

subplot(224), plot(0:(fs5/nfft5):(fs5/2-fs5/nfft5),P1_5(1:nfft5/2));
axis tight, grid on;
ylabel('Amplitude'); xlabel('Frequency (Hz)');
title('Y (251 Hz, 3s)')
```



(b) Perform the discrete Fourier transform of y1 and plot your graph in the frequency domain (include a proper frequency axis). Plot only a positive frequency signal.

Refer to (a) for the code and plot.

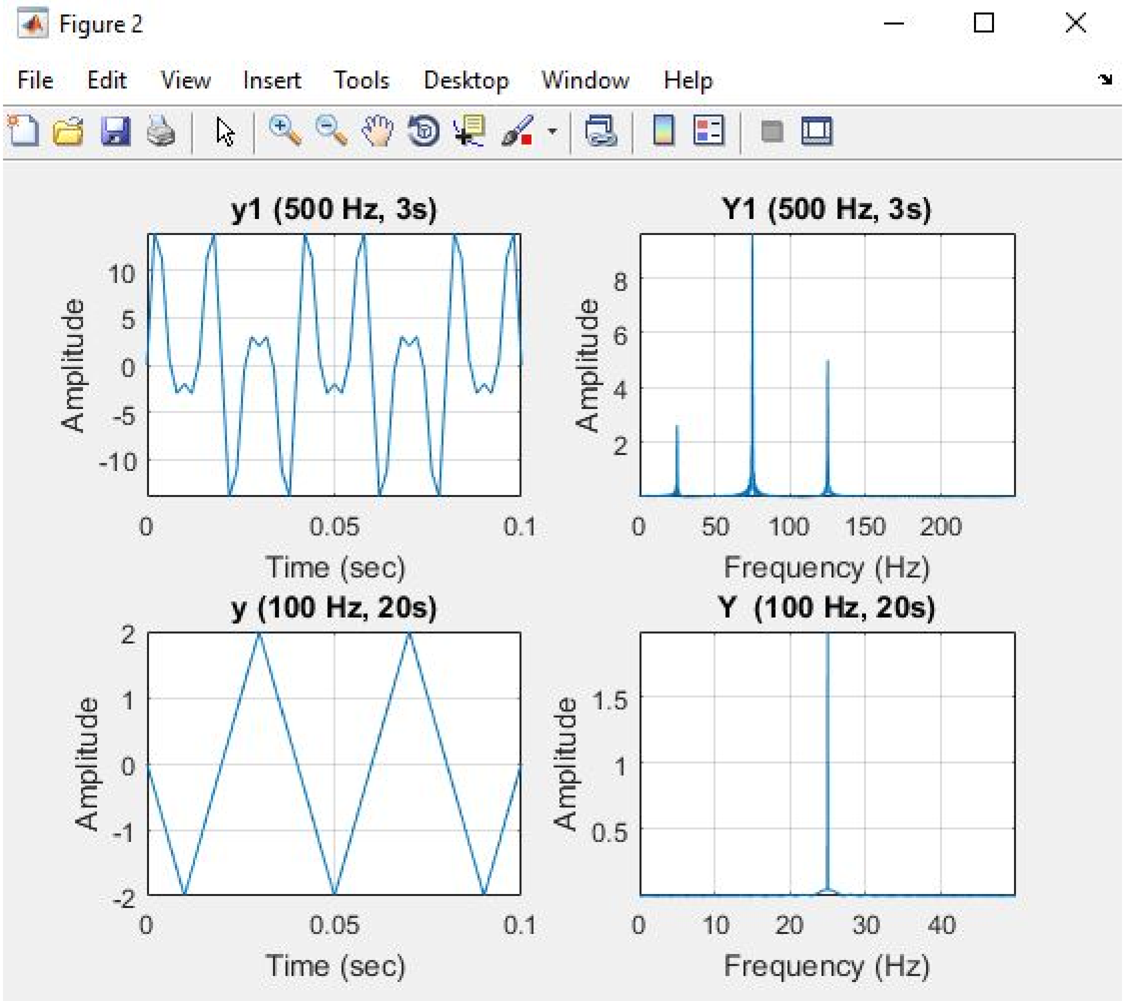
(c) y2 is a discrete signal, which is obtained by digitizing y(t) with a sampling rate of 100 Hz for 3 seconds. Please plot y2 in the time domain (include a proper time axis).

Refer to (a) for the code and plot.

(d) Perform the discrete Fourier transform of y2 and plot your graph in the frequency domain (include a proper frequency axis). Plot only a positive frequency signal.

Refer to (a) for the code and plot.

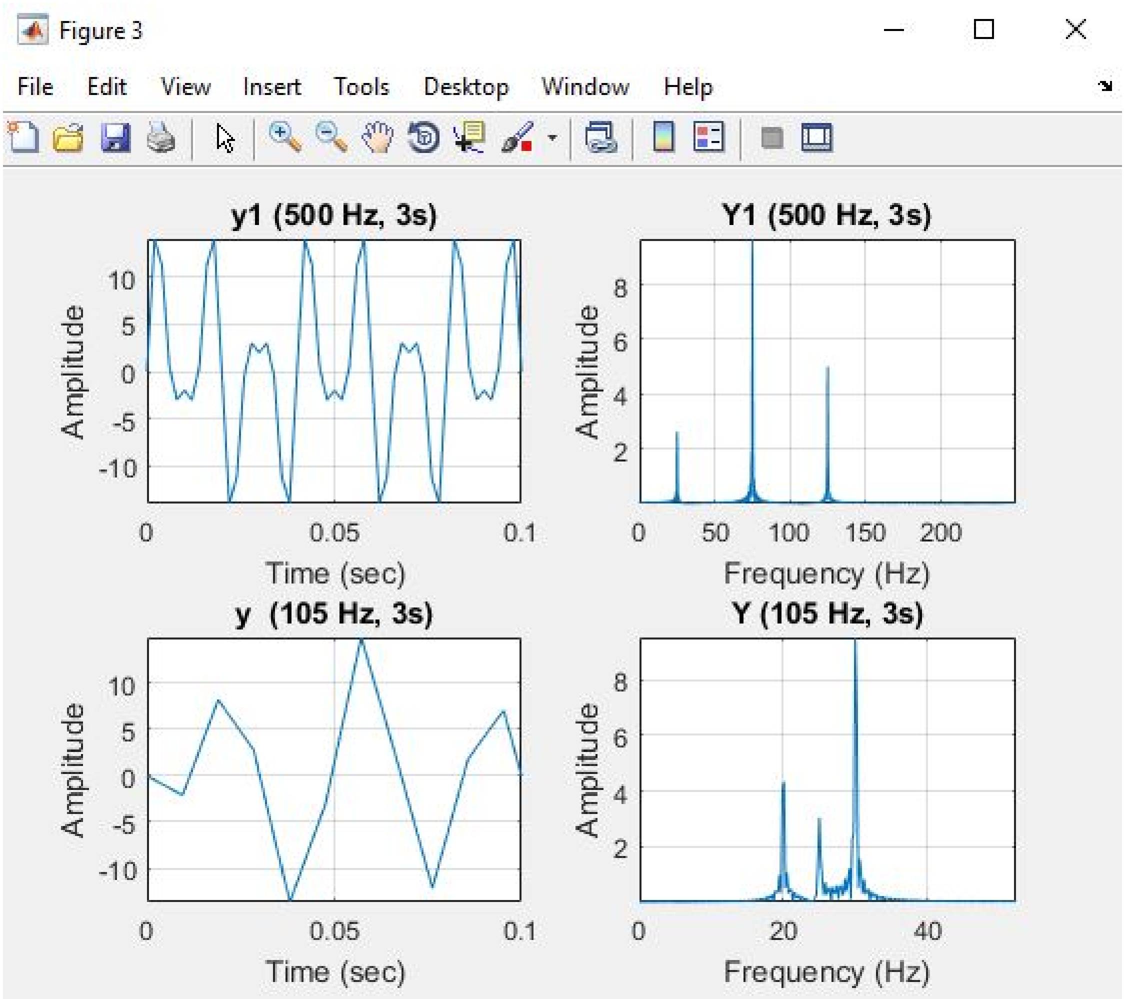
(e) If you digitize a longer-duration signal (let's say 20 seconds) with a sampling rate of 100 Hz, can you measure and extract all frequencies contained in the original signal, y(t)? Please explain your answer.



Refer to (a) for the code.

No, you cannot measure and extract all frequencies contained in the original signal by sampling for a longer period of time. Frequency does not depend on the time duration of sampling, but rather the **sampling rate** compared to the Nyquist frequency. In this case, since the sampling rate is 100 Hz, its frequency content of the resulting discrete response signal is 25 Hz, therefore higher frequency components are not recorded (75 Hz and 125 Hz).

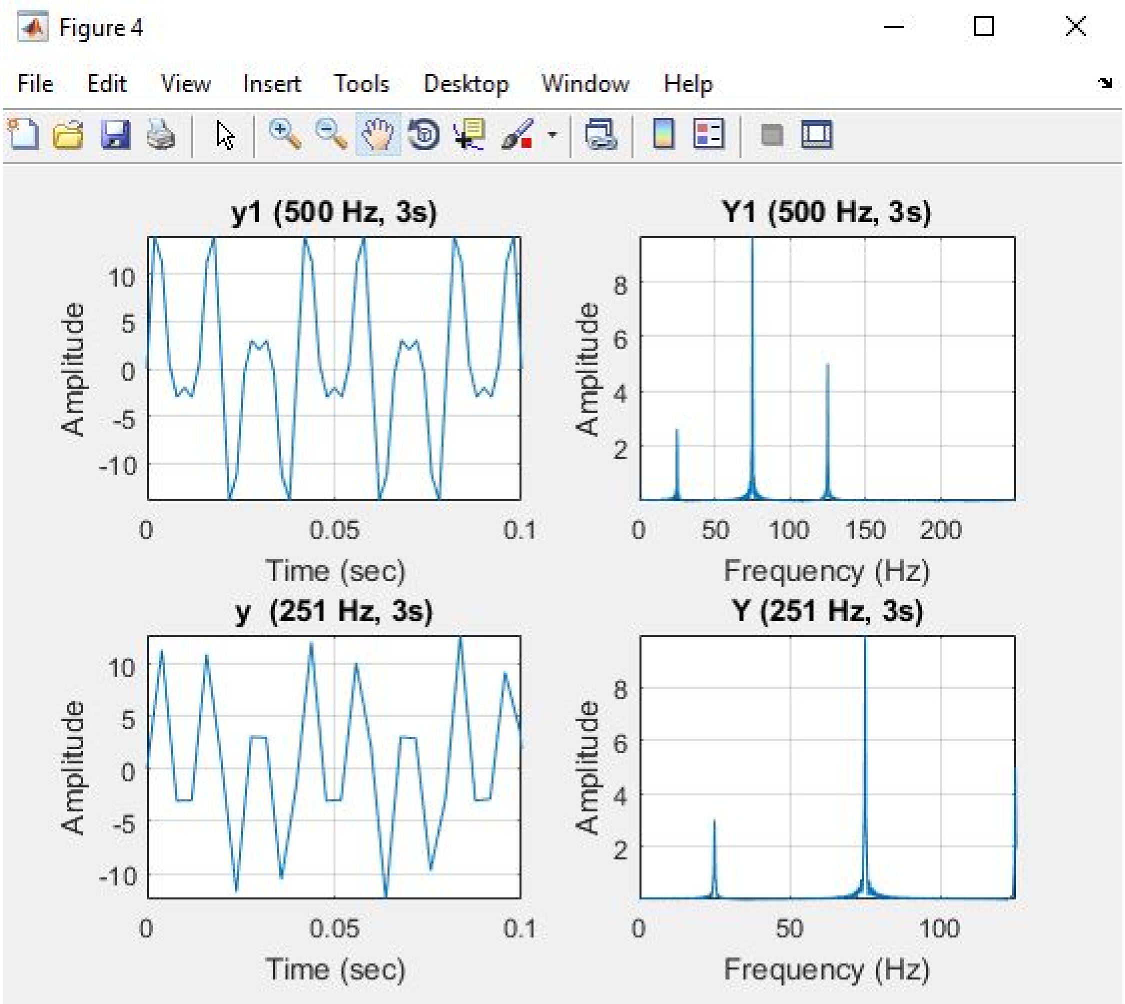
(f) If you digitize the signal with a sampling rate of 105 Hz for 3 seconds, can you measure and extract all frequencies contained in the original signal, y(t)? Please explain your answer.



Refer to (a) for the code.

No, you cannot measure and extract all frequencies contained in the original signal by sampling at the rate of 105 Hz. In this case, the sampling rate is still too slow, therefore higher frequency components are not recorded (75 Hz and 125 Hz).

(g) If you digitize the signal with a sampling rate of 251 Hz for 3 seconds, can you measure and extract all frequencies contained in the original signal, y(t)? Please explain your answer.



Refer to (a) for the code.

No, you cannot measure and extract all frequencies contained in the original signal by sampling at the rate of 251 Hz. In this case, the sampling rate is still too slow, therefore higher frequency components are not recorded (125 Hz).

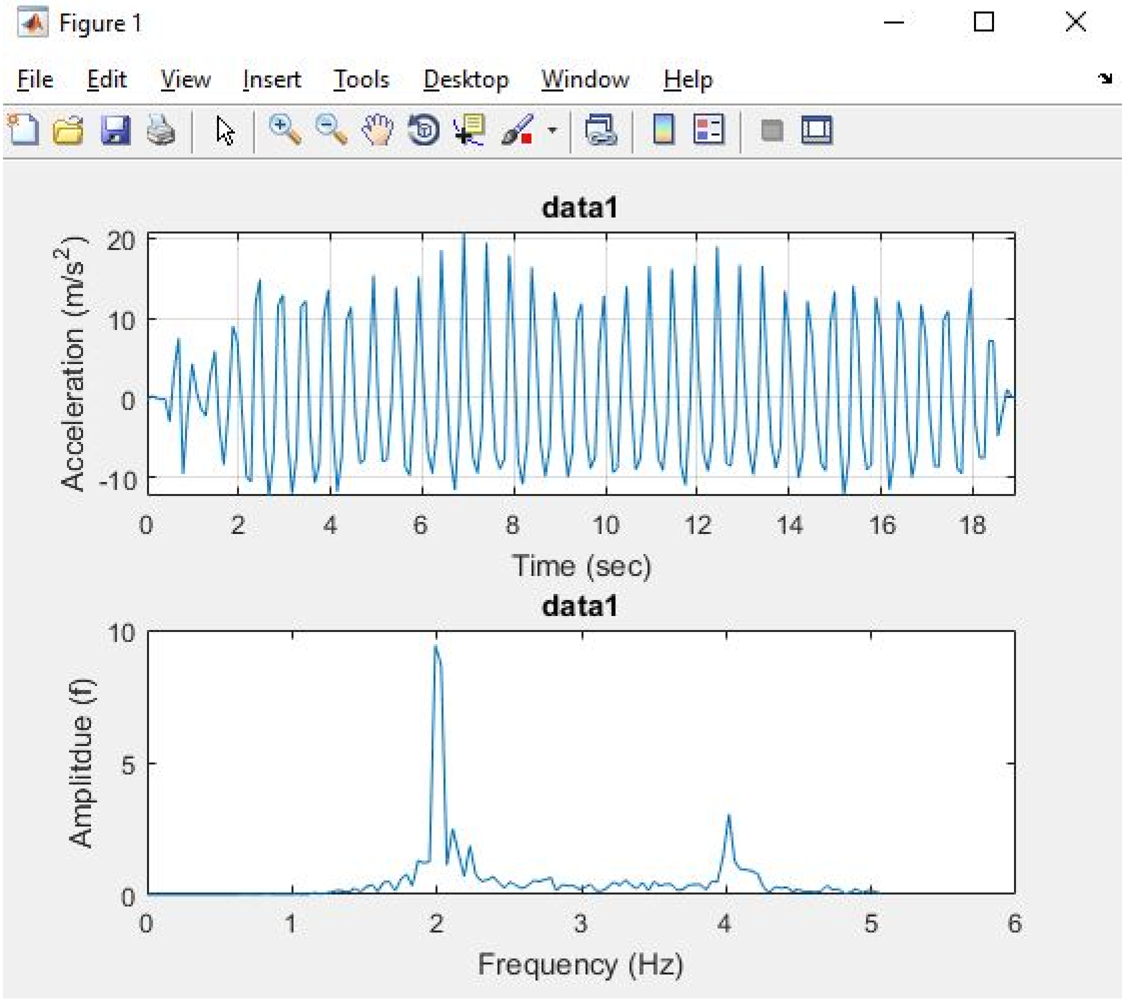
Problem 6: Frequency Analysis

Two sinusoidal accelerations are measured using an accelerometer in a smartphone. Each of the waves is stored at `vib_data1.mat` and `vib_data2.mat`.

(a) Load `vib_data1.mat` and plot the acceleration signal in a z direction (`zvib`) using the corresponding time info (`time`). What is the frequency of this wave?

```
clc, clear, format short, format compact, close all

% (a) vib_data1
```

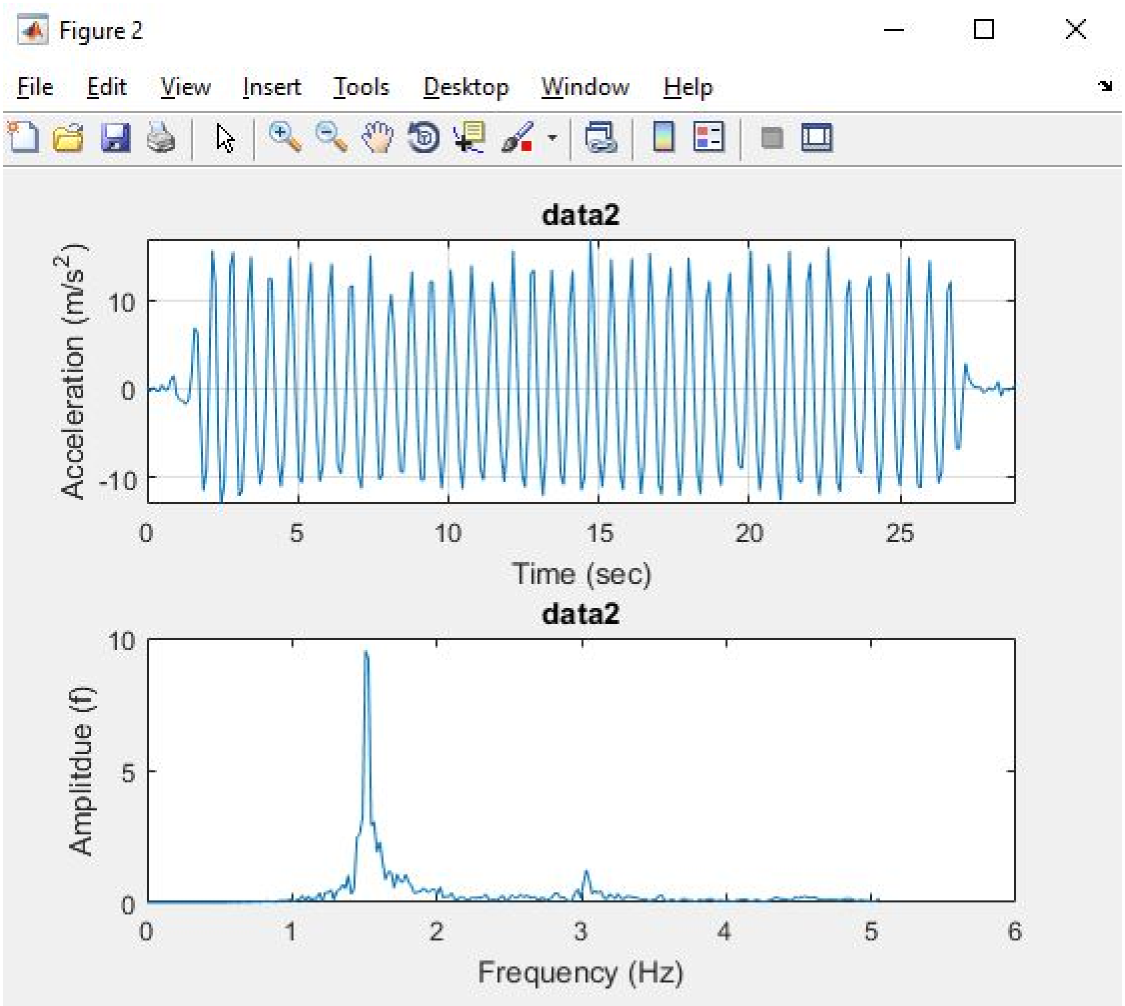
```
load vib_data1
Fs = numel(time)/time(end);

figure
subplot(211); plot(time,zvib);
axis tight; grid on;
ylabel('Acceleration (m/s^2)'); xlabel('Time (sec)');
title('data1')

L = length(zvib);
nfft = 2^nextpow2(L);

Y = fft(zvib,nfft);

P2 = abs(Y/L);
P1 = P2(1:nfft/2+1);
P1(2:end-1) = 2*P1(2:end-1);

subplot(212); plot(0:(Fs/nfft):(Fs/2-Fs/nfft),P1(1:nfft/2));
ylabel('Amplitdue (f)'); xlabel('Frequency (Hz)');
title('data1')

% (b) vib_data1
load vib_data2
Fs = numel(time)/time(end);

figure
subplot(211); plot(time,zvib);
axis tight; grid on;
ylabel('Acceleration (m/s^2)'); xlabel('Time (sec)');
title('data2')

L = length(zvib);
nfft = 2^nextpow2(L);

Y = fft(zvib,nfft);

P2 = abs(Y/L);
P1 = P2(1:nfft/2+1);
P1(2:end-1) = 2*P1(2:end-1);

subplot(212); plot(0:(Fs/nfft):(Fs/2-Fs/nfft),P1(1:nfft/2));
ylabel('Amplitdue (f)'); xlabel('Frequency (Hz)');
title('data2')
```



Based on the frequency plot, the frequency of this wave is 2 Hz.

(b) Load `vib_data2.mat` and plot the acceleration signal in a z direction (`zvib`) using the corresponding time info (`time`). What is the frequency of this wave?

Refer to (a) for the code.



Based on the frequency plot, the frequency of this wave is 1.5 Hz.