# EE321-Lab 6

Sait Sarper Özaslan

22002861

## Part 1)

Part 1

$$y[0] = \underbrace{b[0]\, x[0]}$$

$$y[1] = a[1]\, y[0] + b[0]\, x[1] + b[1]\, x[0]$$

$$= a[1]\, b[0]\, y[0] + b[0]\, x[1] + b[1]\, x[0]$$

Taking the z transform of both sides

$$Y(z) = \sum_{l=1}^{N} a[l]\, z^{-l}\, Y(z) + \sum_{k=0}^{M} b[k]\, z^{-k}\, X(z)$$

$$Y(z)\left(1 - \sum_{l=1}^{N} a[l]\, z^{-l}\right) = \sum_{k=0}^{M} b[k]\, z^{-k}\, X(z)$$

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^{M} b[k]\, z^{-k}}{1 - \sum_{l=1}^{N} a[l]\, z^{-l}}$$

Hence $b[k] = c_n[p]$ , $P = M$

$-a[l] = c_d[q]$ where $c_d[0] = 1$ , $Q = N$

Figure 1: Explanation regarding Z transform

Here is the code for DTLTI

```
function [y]=DTLTI(a,b,x,Ny)
N=length(a)-1;
M=length(b)-1;
Nx=length(x);
y=zeros(1,Ny);%define Ny we can define this from the function
```

```
% The code essentially separates y terms and x terms then adds them up in the end
for n=1:Ny
xtotal=0;
ytotal=0;
for k=0:M
if n-k<=0
g =0;
else
f=n;
g=x(f-k);
end
if n-k>Nx
x(f-k)=0;
end
xtotal=xtotal+g*b(k+1);
end
for l=1:N
if n-l<=0
g=0;
else
f=n;
g=y(f-l);
end
ytotal=ytotal+g*a(l+1);
end
y(n)=xtotal+ytotal;
end
```

## Part 2)

a) Here is the code of impulse response:

```
D=mod(22002861,4);
M=5+D; %M=6
k=1:1:M;
b(1,k)=exp(-(k-1));
a=0;
x=zeros(1,11);
x(1)=1;
Ny=11;
x(1)=1;
y=DTLTI(a,b,x,Ny);
figure
stem(0:length(x)-1,y,"*"),title("Impulse response")
```

Here is the plot of the impulse response:

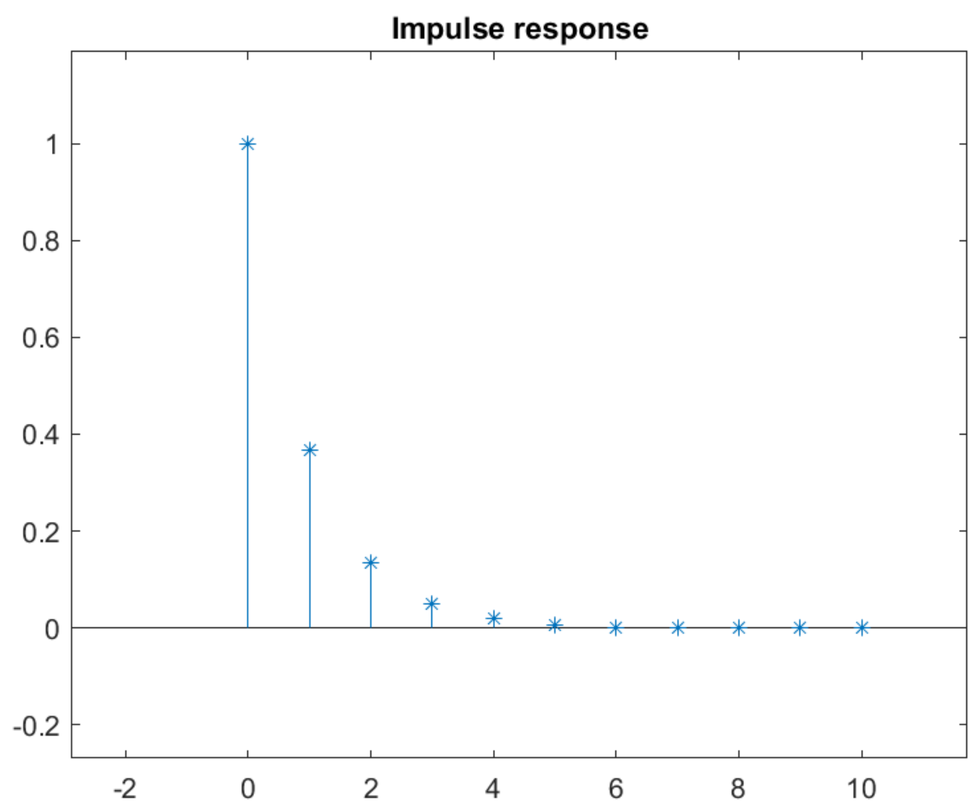



Figure 2: Impulse response

b) The nonzero values of h[n] is same as b[k]s. We can come to this conclusion also from when a[l]=0 and x[n] is an impulse at n we will see the same response for h[n] and b[n]

$$h[n] = \sum_{k=0}^{M} b[k]\,\delta[n-k] = b[n] = e^{-n}, \quad \text{For } 0 < n < M-1$$

Figure 3: Similarity between h[n] and b[k]

c) The impulse response is not a recursive function and it only depends on the input [n]. Moreover, the impulse response h[n] = 0 for n<0 and n>M. As M= 6 we could only see 6 nonzero values.

d) Here is the computation of DTFT:

$$H(z) = \sum_{n=0}^{M-1} e^{-n} z^{-n}$$

$M=6$

$$H(z) = 1 + e^{-1} z^{-1} \cdots + e^{-5} z^{-5}$$

we can turn this to fourier transform by

$$H(z) = H(e^{jw}) = 1 + e^{-1} e^{-jw} + \cdots e^{-5} e^{-jw}$$

Figure 4: Computation of DTFT

e) Here is the code for this part:

We used what we found in part d as a loop in this part's code.

```
freq=-pi:1/1000:pi-1/1000;%freq is the frequency values from -pi to pi

H = 0;

j=sqrt(-1);

for i=1:M-1

H=H+(exp(-i-j*freq*i));

end

H=H+1;

figure(2),plot(freq,abs(H)),title("Filter"),xlabel("Frequency"),ylabel("Magnitude")

%this is a lowpass filter
```

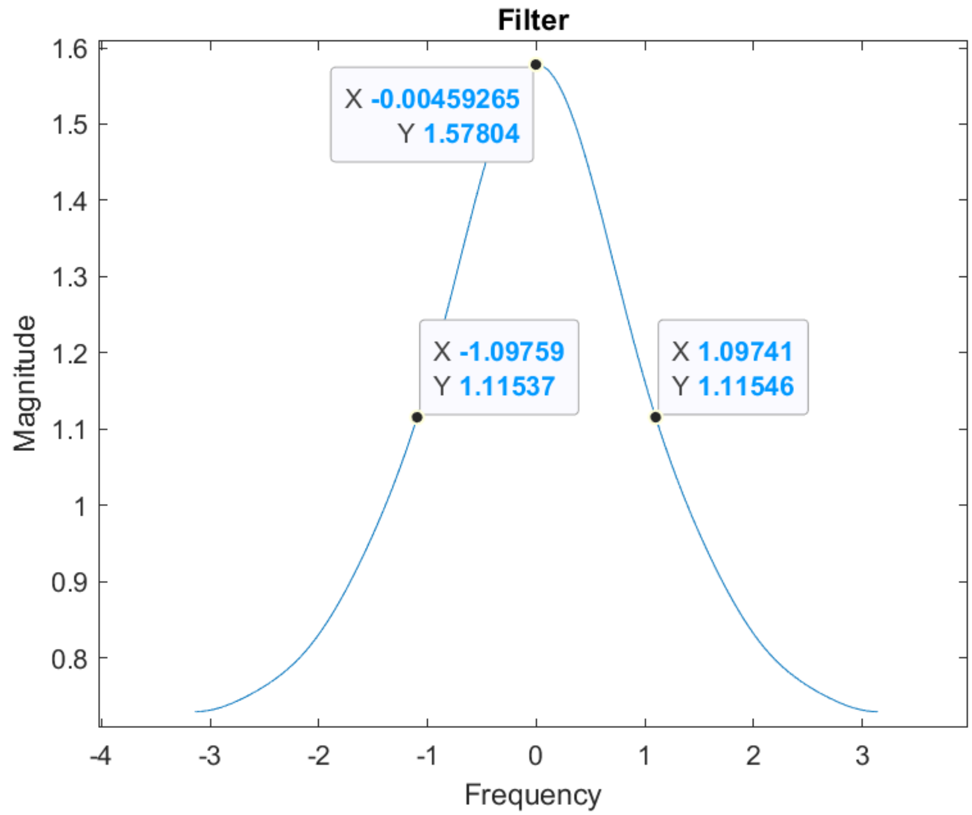



Figure 5: Magnitude Response and Cutoff Frquency

As around 0 pi this filter reaches it’s maximum value and magnitude decreases as we get further away from the center, we can say that this is a lowpass filter. The meaning of 3dB bandwidth for filters refers to range where the endpoints are 3dB less than the maximum output’s gain. In terms of mathematics, this refers to points where the value is $\frac{1}{\sqrt{2}}$ less of the maximum magnitude of the filter. As the maximum magnitude is y= 1.578, the 3dB cutoff frequencies are around y= 1.115. This corresponds to $x = \pm 1.097\ \frac{rad}{sec}$. Hence the bandwidth in this case would be $2.194\ \frac{rad}{sec}$

f) Here is the code for the given part:

```
%lab6 part 2 f)

M=6;

l=1:1:M;

b(1,l)=exp(-(l-1));

a=0;

fsampling=1400;

fend=700;

t=0:1/fsampling:1-1/fsampling;%sampled interval
```

```
%I will take beginningfreq = 0 so I didn't include in the code for k, also

%we need the frequency of the cos to fit into nyquist rate

%we need to calculate k which should allow us to still be in criteria at

%max freq which is 1400/2=700Hz, k decides on the final(instantenous)

%frequency hence,

k = fend; %700 reaches the max rate

phi_t=k*(t.^2)/2;

x=cos(2*pi*phi_t);

y1=DTLTI(a,b,x,length(x));

%transforming the interval to fit into freq response

freqt=0:pi/length(y1):pi-pi/(length(y1));

figure(3)

subplot(1,3,1)

plot(freqt,abs((y1))), title("Linear Chirp Signal 0<t<1"), xlabel("Freq(hz)"),
ylabel("Response")

xlim([0,pi]);

% now writing the case for other time duration

t10=0:1/fsampling:10-1/fsampling;

k=fend/(10); %as duration has increased we should reduce the instaneous freq

phi_t10=k*(t10.^2)/2;

x10=cos(2*pi*phi_t10);

y2=DTLTI(a,b,x10,length(x10));

freqt10=0:pi/length(y2):pi-pi/(length(y2));

subplot(1,3,2)

plot(freqt10,abs((y2))),title("Linear Chirp Signal 0<t<10"), xlabel("Freq(hz)"),
ylabel("Response")

xlim([0,pi]);

% now writing the case for last time duration

t1000=0:1/fsampling:1000-1/fsampling;

k=fend/(1000); %as duration has increased we should reduce the instaneous freq

phi_t1000=k*(t1000.^2)/2;

x1000=cos(2*pi*phi_t1000);
```

```
y3=DTLTI(a,b,x1000,length(x1000));

freqt1000=0:pi/length(y3):pi-pi/(length(y3));

subplot(1,3,3)

plot(freqt1000,abs((y3))), title("Linear Chirp Signal 0<t<1000"), xlabel("Freq(hz)"),
ylabel("Response")

xlim([0,pi]);
```

Here are the plots:

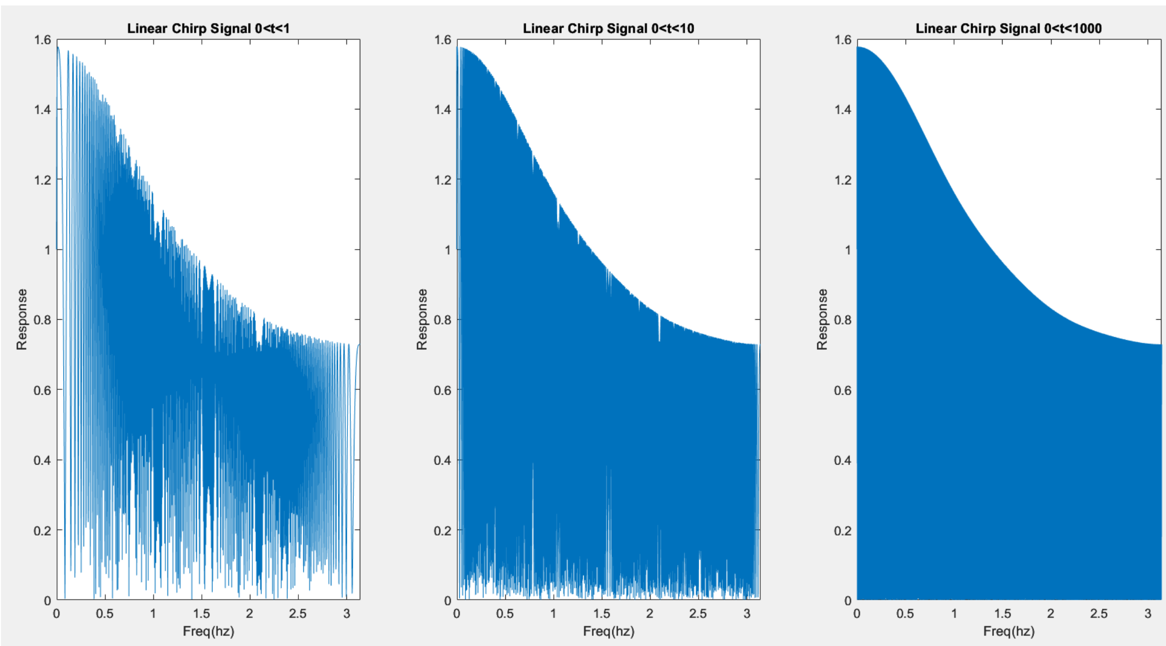



Figure 6: Linear Chirp Signal with Different Sweeps

Comparing the first plot to the one in part e), we can see that magnitude decreases as we get further from the center point which is 0. We can see that the lowpass filter creates an envelope over the output. Regarding the jumps, though there are not any particular in terms of amplitude there are some holes in the frequency. We can reduce these holes by creating more samples in the same duration. This can easily be seen in the plots as we increased the samples, we observed a smoother plot.

## Part 3)

a) ID=22002861

$n = [4, 4, 2, 2, 4, 10, 3, 1]$

$$z_1 = \frac{4+2j}{\sqrt{20}}, \quad p_1 = \frac{4+4j}{\sqrt{33}} \quad p_2 = \frac{3+10j}{\sqrt{110}}$$

$$H(z) = \frac{(1 - z_1 z^{-1})}{(1-p_1 z^{-1})(1-p_2 z^{-1})}$$

b) $$\frac{Y(z)}{X(z)} = \frac{1 - z_1 z^{-1}}{(1-p_1 z^{-1})(1-p_2 z^{-1})} \Rightarrow Y(z)\,(1 - p_1 z^{-1} - p_2 z^{-1} + p_1 p_2 z^{-2}) = X(z)\cdot(1 - z_1 z^{-1})$$

Turning this to time

$$y[n] - (p_1+p_2)\,y[n-1] + p_1 p_2\, y[n-2] = x[n] - z_1\, x[n-1]$$

$$y[n] = (p_1+p_2)\, y[n-1] - p_1 p_2\, y[n-2] + x[n] - z_1\, x[n-1]$$

Figure 7: a and b of Part 3

c)

$$H(z) = \frac{1 - z_1 z^{-1}}{(1-p_1 z^{-1})\cdot(1-p_2 z^{-1})} = \left(\frac{A}{1-p_1 z^{-1}} + \frac{B}{1-p_2 z^{-1}}\right)\cdot(1 - \underbrace{z_1 z^{-1}}_{\text{delay}})$$

$$A = \frac{p_2}{p_2 - p_1} \qquad B = \frac{p_1}{p_1 - p_2}$$

$$h[n] = \frac{p_2}{p_2-p_1}(p_1)^n u[n] + \frac{p_1}{p_1-p_2} p_2^n u[n] - z_1 \frac{p_2}{p_2-p_1} p_1^{n-1} u[n-1] - z_1 \frac{p_1}{p_1-p_2} p_2^{n-1} u[n-1]$$

Figure 8) Impulse Response

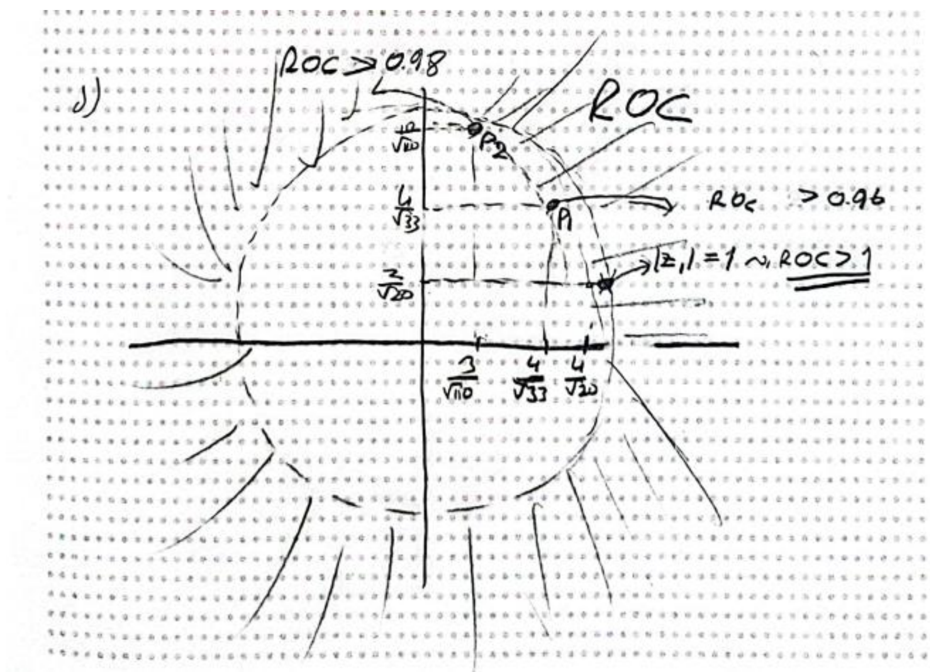



Figure 9: ROC Plot

e) The system can be considered stable as the maximum magnitude of the zeros and poles does not surpass 1.

f) This is an IIR as the system's poles cause a recursive y[n] that can continue infinitely.

g) The transform to the DTFT is done in the code:

```
%part 3 of the lab

%

z1=(4+2j)/sqrt(20);

p1=(4+4j)/sqrt(33);

p2=(3+10j)/sqrt(110);

% H_z=@(z) (z-z1)/((z-p1)*(z-p2));

j=sqrt(-1);

%again, we need to transform z to e^jw in unit circle

H_ejw=@(w) (exp(j*w)-z1)./((exp(j*w)-p1).*(exp(j*w)-p2));

w=-pi:pi/1000:pi;%frequency

magH=abs(H_ejw(w));

figure (4),plot(w,magH),title("e^j^w vs w"),xlabel("w(Freq)"),ylabel("abs(e^j^w)")
```

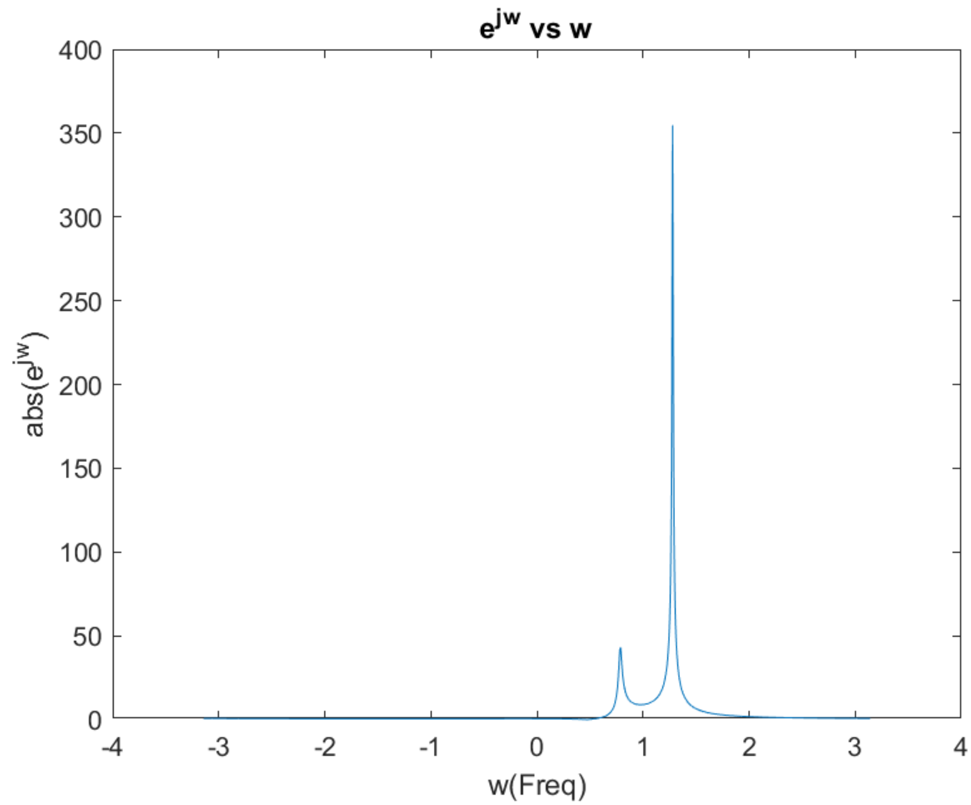



Figure 10: Magnitude of the DTFT Filter

The peaks occur between 0 and 1.5 of the frequency ranges so we can say that this is a bandpass filter.

h) Here is the code for part h:

```
fS = 1400;
beginningfreq=-700;
a=[p1+p2 -p1*p2];
b=[1 -z1];
%% t = 1
t = 0:1/fS:1;
k1 = 700;%same idea as in part 2f was used
x=exp(1j*2*pi*((beginningfreq.*t)+(k1.*t.^2))/2);
y = DTLTI(a, b, x, length(t));
freq=-pi:2*pi/length(y):pi-2*pi/length(y);
figure(5)
subplot(1,2,1)
```

```
plot(freq,abs(y)),title("0<t<1"),xlabel("freq(w)"),ylabel("magnitude")
subplot(1,2,2)
plot(freq,angle(y)),title("0<t<1"),xlabel("freq(w)"),ylabel("phase")
%% t = 10
t=0:1/fS:10;
k1=70;
x=exp(1j*2*pi*((beginningf0.*t)+(k1.*t.^2))/2);
y = DTLTI(a, b, x, length(t));
freq=-pi:2*pi/length(y):pi-2*pi/length(y);
figure(6)
subplot (1,2,1)
plot(freq,abs(y)),title("0<t<10"),xlabel("freq(w)"),ylabel("magnitude")
subplot (1,2,2)
plot(freq,angle(y)),title("0<t<10"),xlabel("freq(w)"),ylabel("phase")
%
%% t = 1000
t=0:1/fS:1000;
k1=0.7;
x=exp(1j*2*pi*((beginningf0.*t)+(k1.*t.^2))/2);
y = DTLTI(a, b, x, length(t));
freq=-pi:2*pi/length(y):pi-2*pi/length(y);
figure(17)
subplot(1,2,1)
plot(freq,abs(y)),title("0<t<1000"),xlabel("freq(w)"),ylabel("magnitude")
subplot(1,2,2)
plot(freq,angle(y)),title("0<t<1000"),xlabel("freq(w)"),ylabel("phase")
```

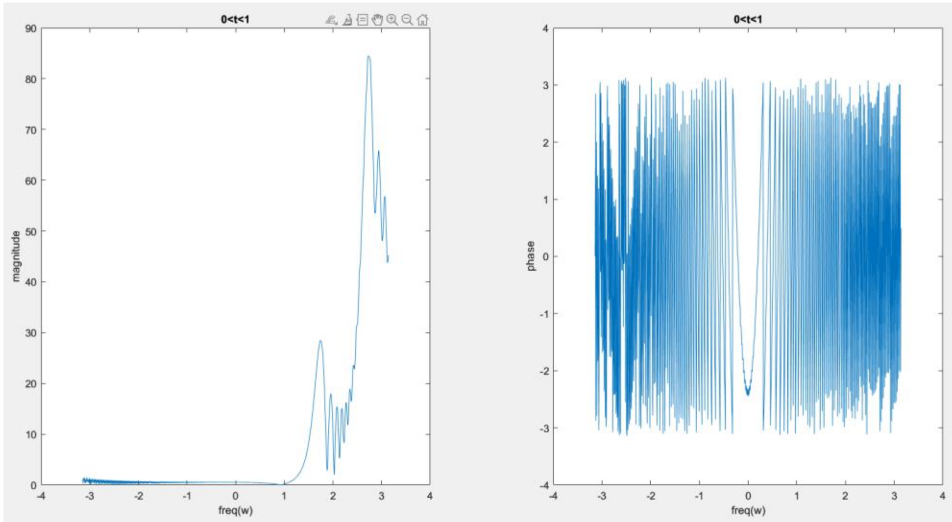



Figure 11: Magnitude and Phase of Filter when 0<t<1

The magnitude plot on the left is similar to magnitude plot we found in part g. The difference In magnitude is caused by how we sample the frequency axis. As there are not enough points in the frequency interval we have chosen. The plot is also not symmetrical as the impulse function itself is a complex function where complex values are simply not conjugates of each other. As we increase the points in the interval, we see that we get more accurate and smooth magnitudes.

If we had swept the frequency range to –600 to 800Hz, all we need to do is simply change the sampling frequency to 1600Hz to meet the Nyquist Criteria.

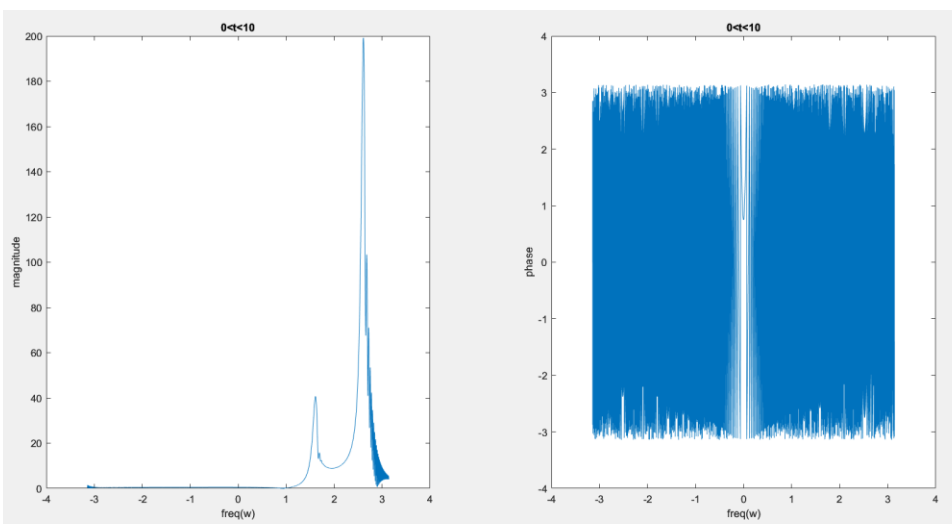
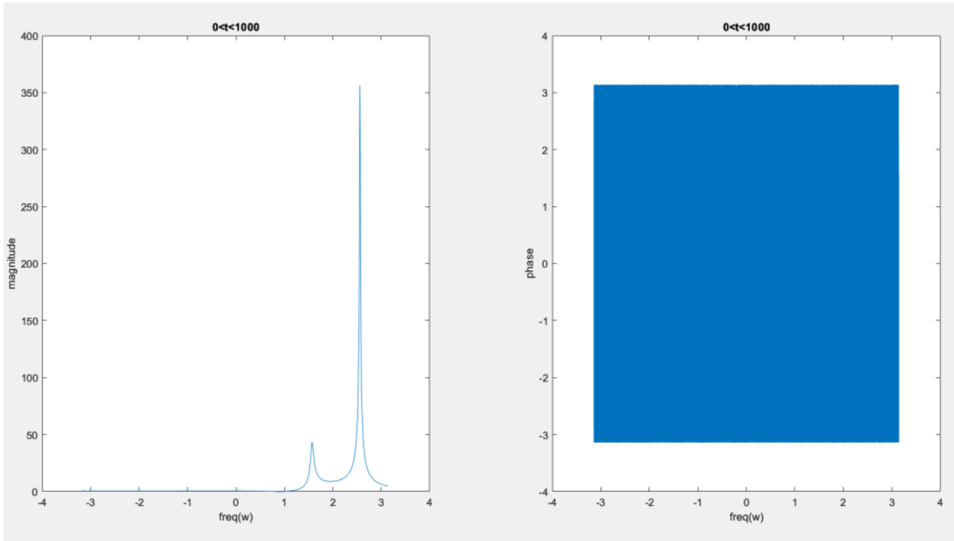



Figure 12: Magnitude and Phase of Filter when 0<t<10




Magnitude and Phase of Filter when 0<t<1000

The points where peaks occur also get closer to the points where our original bandpass filter has peaks as we increase the density of our sweep.