CT.2306 : Signal & Systems II

Report

Processing motion signals from a PTZ camera

Philéas CURTIL, Thomas MAFILLE, Rémy MAILLARD

January 7^{th}, 2024

isep
École d'ingénieurs du numérique

Made with LaTeX

Contents

1 Data visualization

1) When we load *data-proj.mat*, we can see that there are two vectors in the file :

```
Name          Size

omega         1x20001
t             1x20001
```

2) Plot of the angular speed *omega* as a function of time.

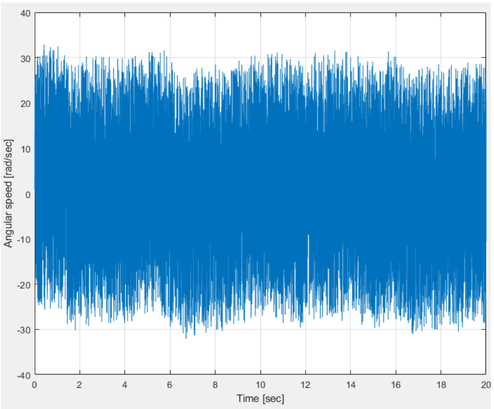


Figure 1: Angular speed as a function of time

To obtain this graph :

```
figure(1)
plot(t, omega)
grid on
hold on
xlabel('Time [sec]')
ylabel('Angular speed [rad/sec]'
   )
```

It is not possible to use the signal as it is now, mostly because there is too much information (too noisy) or the window is too large. This signal is continuous (analog). Electronic control devices requires digital signals.

2 Analog filtering

3) The sampling period T_{e_1} can be calculated with :

```
Te1=t(2)-t(1)

>>Te1 =

1.0000e-03
```

4) Plot of the amplitude spectrum of $omega(t)$, with the use of the workshop 5 :

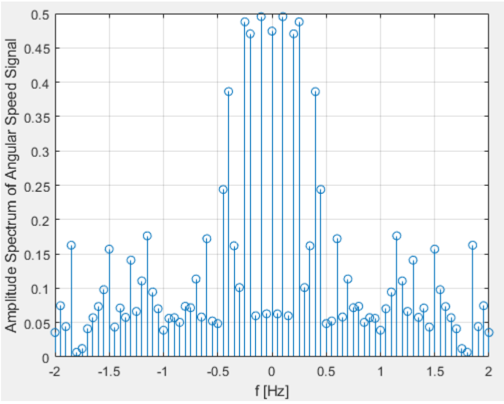


Figure 2: DFT plot of $omega(t)$

To obtain this graph :

```
% Plot of the DFT of omega(t)
Te2= 0.05;
Fe1=1/Te1;
Tf=t(end);
N=Tf/Te1;

f1=-Fe1*(N/2-1)/N:Fe1/N:0;
f2=Fe1/N:Fe1/N:(N/2)*Fe1/N;
f = [f2,f1];
w= zeros(N,1);
for m=1:N
  for k=1:N
    w(m)=w(m)+omega(k)*exp(-1i
       *2*pi*m*k/N);

  end
end

figure(2)
stem(f,abs(w)/N)
grid on
xlim([-2 2])
xlabel('f [Hz]')
ylabel('Amplitude Spectrum of
   Angular Speed Signal')
```

5) The frequencies contained inside the signal are ranging from -2 Hz to 2 Hz with a step of 0.05.

$F_{max} = 2Hz$

6) The cutoff frequency is $f_c = 2Hz$.
On Matlab :

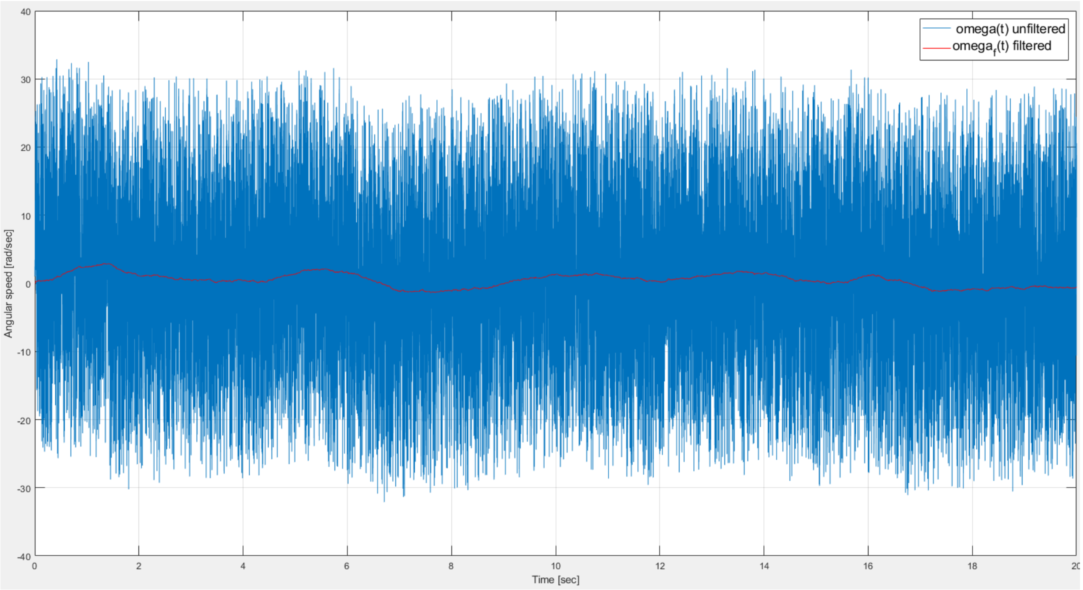


Figure 3: $omega(t)$ filtered and unfiltered $(omega_f(t))$

To obtain this graph :

```
% filter design
t1=0:Te1:t(end)-Te1;
fc=0.2;
wc=2*pi*fc;

H1=tf(1,[1/(2*pi*fc)  1]);
wf=lsim(H1,omega,t);

% plot of filtered signal
figure(1);
plot(t,wf,'r')
hold off
grid on
legend(' omega(t) unfiltered','omega_{f}(t) filtered','Fontsize'
    ,14)
```

On Simulink :

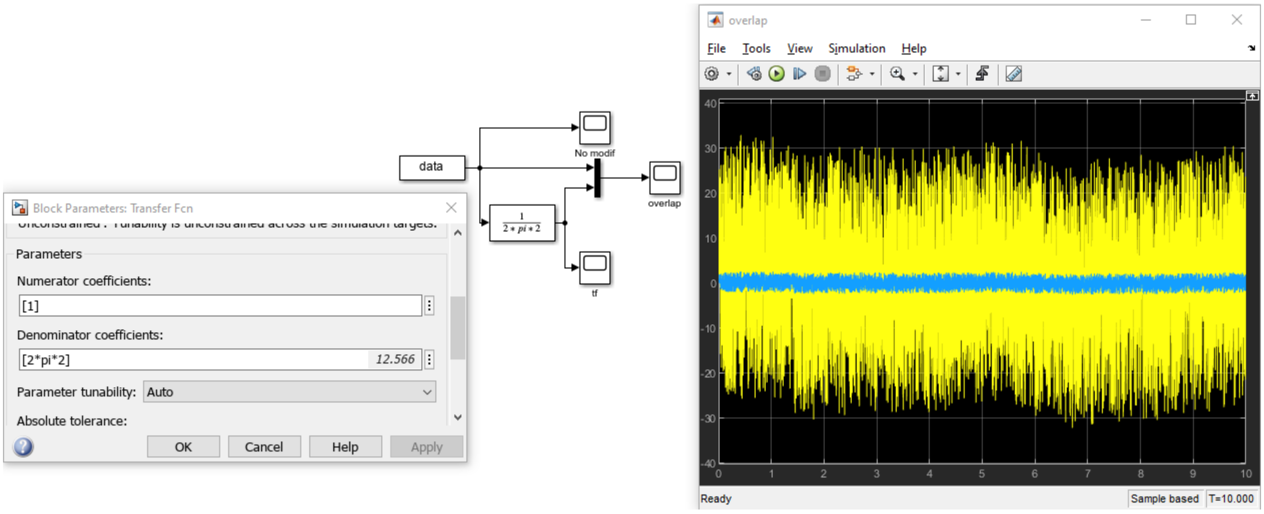


Figure 4: Simulink

We need to add a part in the Matlab code as well :

```
data = [t',omega'];
```

7) Plot of the amplitude spectrum of $omega_f(t)$, with the use of the workshop 5 :

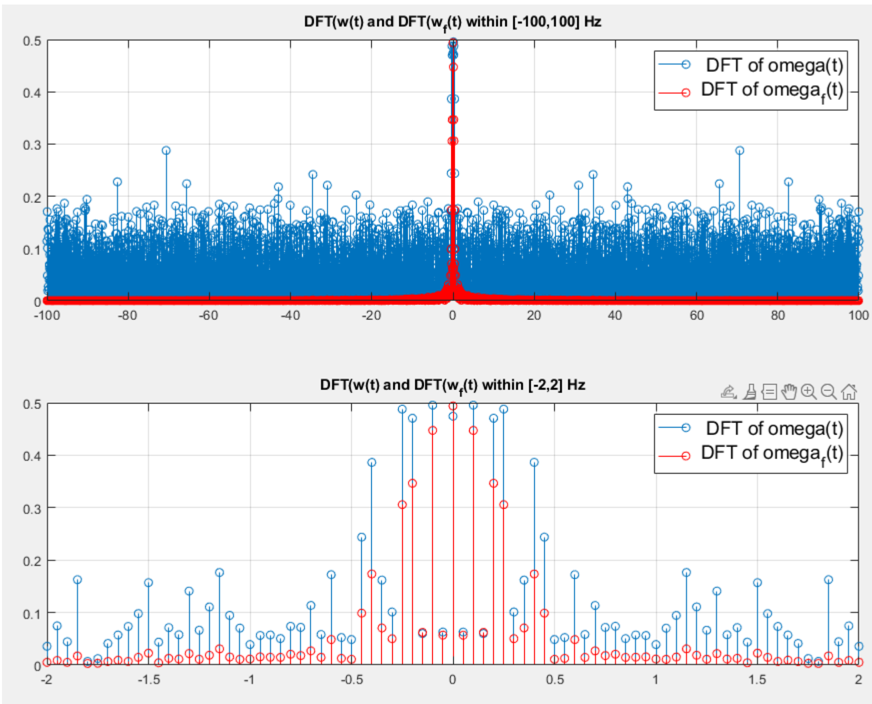


Figure 5: DFT plot of $omega(t)$ and $omega_f(t)$

To obtain this graph :

```
wf1=zeros(N,1);
for m = 1 : N
    for k = 1 : N
        wf1(m) = wf1(m) + wf(k) * exp(-1i*2*pi*m*k/N);
    end
end

figure(3)
subplot(2,1,1);
stem(f,abs(w)/N), hold on
stem(f,abs(wf1)/N, 'r'), hold off
grid on
xlim([-100 100])
legend(' DFT of omega(t)','DFT of omega_{f}(t)','Fontsize',14)
title('DFT(w(t) and DFT(w_{f}(t) within [-100,100] Hz')

subplot(2,1,2);
stem(f,abs(w)/N), hold on
stem(f,abs(wf1)/N, 'r'), hold off
grid on
xlim([-2 2])
legend(' DFT of omega(t)','DFT of omega_{f}(t)','Fontsize',14)
title('DFT(w(t) and DFT(w_{f}(t) within [-2,2] Hz')
```

3 Sampling

8) To create a vector $omega_e(t)$ which contains the values of the vector $omega_f(t)$ with a period between the values of $T_{e2} = 0.05$ sec.

```
temp1 = 1:round(Te2/Te1):length(t);
we=wf(temp1);
Te = t(temp1);
```

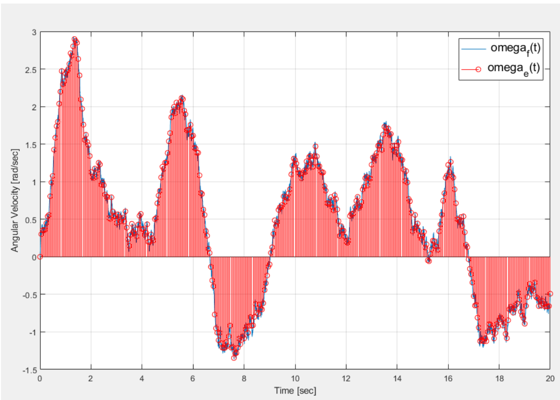
9) To get the size of $omega_e(t)$, we use :

```
>> size(we)

ans =

   401     1
```

10) The new vector t_e which corresponds to the vector $omega_e(t)$ can be created using :

```
temp1 = 1:round(Te2/Te1):length(t);
we=wf(temp1);
Te = t(temp1);
```

11) Plot of $omega_f(t)$ and $omega_e(t)$:




```
figure(4)
plot(t,wf), hold on
xlabel('Time [sec]')
ylabel('Angular Velocity [rad/
   sec]')
grid on
stem(Te,we, 'r'), hold off
legend(' omega_{f}(t)','omega_{e
   }(t)','Fontsize',14)
```

Figure 6: Plot of $omega_f(t)$ and $omega_e(t)$

To plot the graph between 10 and 12, we add xlim([10 12]) to the Matlab code.

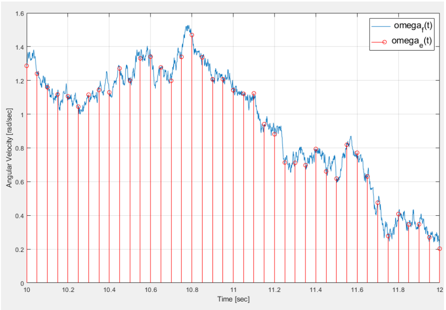


Figure 7: Plot of $omega_f(t)$ and $omega_e(t)$, focused between 10 and 12

```
figure(4)
plot(t,wf), hold on
xlim([10 12])
xlabel('Time [sec]')
ylabel('Angular Velocity [rad/
   sec]')
grid on
stem(Te,we, 'r'), hold off
legend(' omega_{f}(t)','omega_{e
   }(t)','Fontsize',14)
```

12) Plot of $omega(t)$, $omega_f(t)$ and $omega_e(t)$:

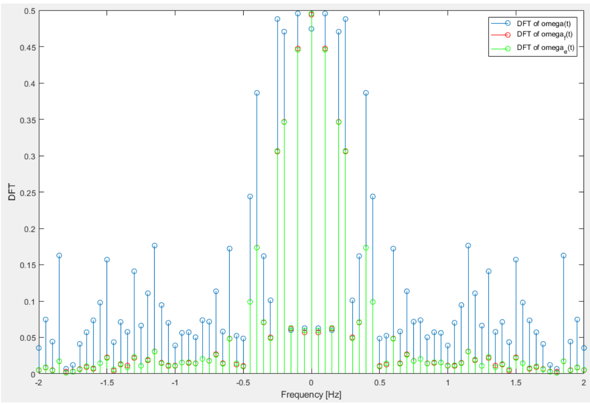


Figure 8: Plot of $omega_f(t)$ and $omega_e(t)$

```
Fe2=1/Te2;
Tf2=Te(end);
N2=Tf2/Te2;

f3=-Fe2*(N2/2-1)/N2:Fe2/N2:0;
f4=Fe2/N2:Fe2/N2:(N2/2)*Fe2/N2;
f_2=[f4,f3];

we_dft = zeros(N2,1);
for m = 1 : N2
    for k = 1 : N2
        we_dft(m) = we_dft(m) +
           we(k) * exp(-1i*2*pi*
           m*k/N2);
    end
end

figure(5)
stem(f,abs(w)/N), hold on
stem(f,abs(wf1)/N, 'r')
stem(f_2,abs(we_dft)/N2, 'g'),
   hold off
xlabel('Frequency [Hz]')
ylabel('DFT')
legend({'DFT of omega(t)', 'DFT
   of omega_f(t)', 'DFT of
   omega_e(t)'})
xlim([-2 2])
```

4 Angular position and acceleration

13) To calculate the angular acceleration$omegadot(t)$ and the position $theta(t)$:

```
% angular acceleration
wd_start=(we(2)-we(1))/Te2;
wd_end=(we(end)-we(end-1))/Te2;
wd_mid=zeros(8,1);
for i=2:N2-1
    wd_mid(i)=(we(i+1)-we(i-1))/(2*Te2);
end

wd=[wd_start;wd_mid;wd_end];


% angular position
theta=zeros(N2,1);

for i=1:N2
    for k=1:i
        theta(i)=theta(i)+Te2*we(k);
    end
end
```

14) We can now plot the two graphs for $theta(t)$ and $omega'(t)$:

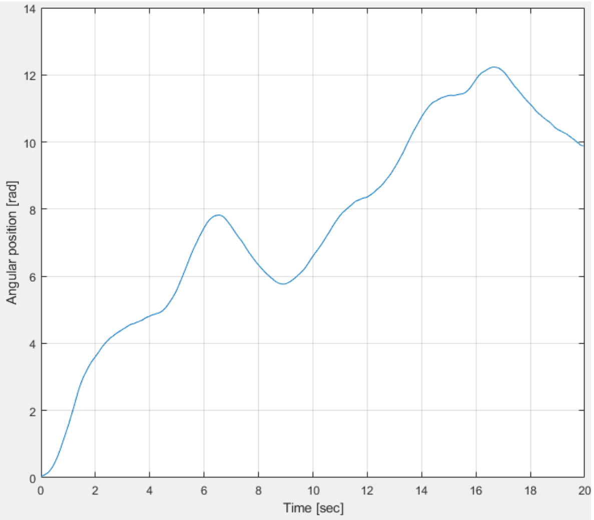


Figure 9: Plot of $theta(t)$

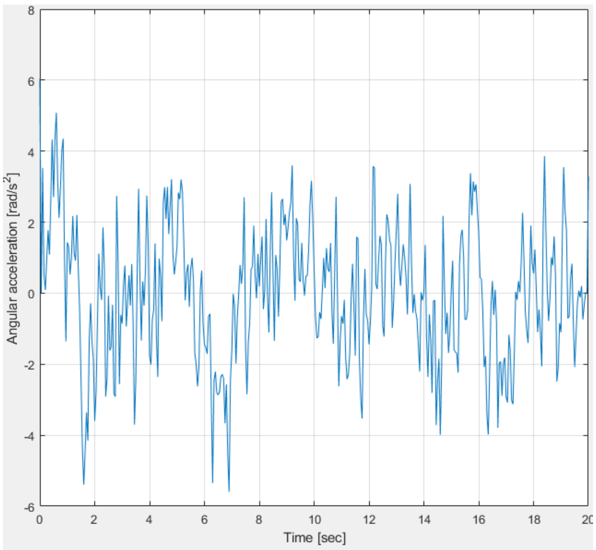


Figure 10: Plot of $omega'(t)$

To get the plots :

```
t_ang=0:Te2:Te(end)-Te2;

figure(6)
plot(t_ang, theta)
xlabel('Time [sec]')
ylabel('Angular position [rad]')
grid on
```

```
figure(7)
plot(Te,wd)
xlabel('Time [sec]')
ylabel('Angular acceleration [rad/s^2]')
grid on
```

15) The plot of the DFT of $theta(t)$ and $omega'(t)$:

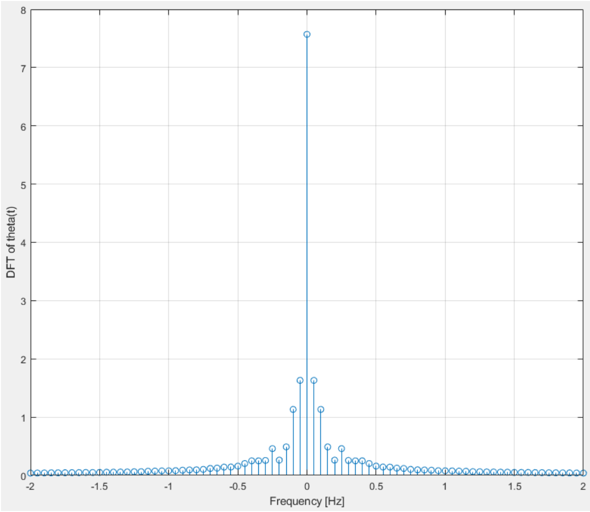


Figure 11: Plot of DFT($theta(t)$)

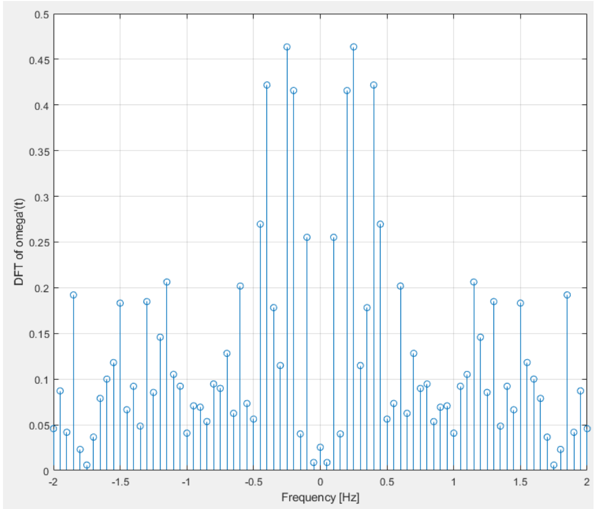


Figure 12: Plot of DFT($omega'(t)$)

To get the plots and calculate the DFT of $omega'(t)$ and $theta(t)$:

```
wd_dft=zeros(N2,1);
for m=1:N2
    for k=1:N2
        wd_dft(m)=wd_dft(m)+wd(k)*exp(-1i*2*pi*m*k/N2);
    end
end

theta_dft = zeros(N2,1);
for m=1:N2
    for k=1:N2
        theta_dft(m)=theta_dft(m)+theta(k)*exp(-1i*2*pi*m*k/N2);
    end
end

figure(8)
stem(f_2,abs(theta_dft)/N2)
xlim([-2 2])
grid on
xlabel('Frequency [Hz]')
ylabel('DFT of theta(t)')

figure(9)
stem(f_2,abs(wd_dft)/N2)
xlim([-2 2])
grid on
xlabel('Frequency [Hz]')
ylabel("DFT of omega'(t)")
```

5 Digital filtering

16) To calculate $H_2(z)$ with Matlab, we can use :

```
[num,denum]=tfdata(H1,'v');
[num_digital, denum_digital] = bilinear(num,denum,Fe2,fc);

H2=tf(num_digital, denum_digital, Te2, 'Variable', 'z');
```

17) We can plot $omega'(t)$ and $omega'_f(t)$ in matlab :

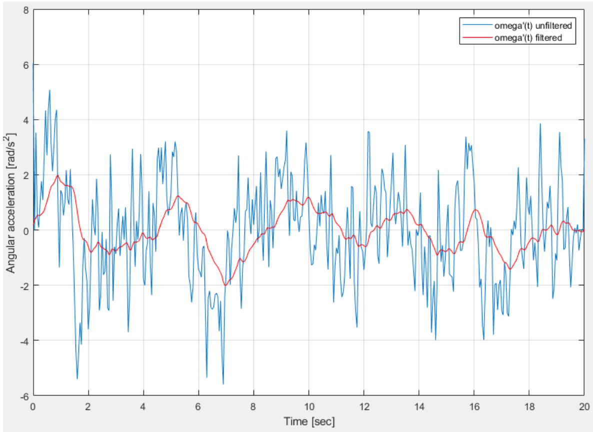


Figure 13: Plot of $omega'(t)$ and $omega'_f(t))$

To obtain this plot :

```
wd_f=lsim(H2,wd,Te);

figure(10)
plot(Te,wd), hold on
plot(Te,wd_f, 'r'), hold off
grid on
xlabel('Time [sec]')
ylabel('Angular acceleration [
    rad/s^2]')
legend({"omega'(t) unfiltered",
    "omega'(t) filtered"})
```

18) The plot of the DFT of $omega'(t)$ and $omega'_f(t)$:

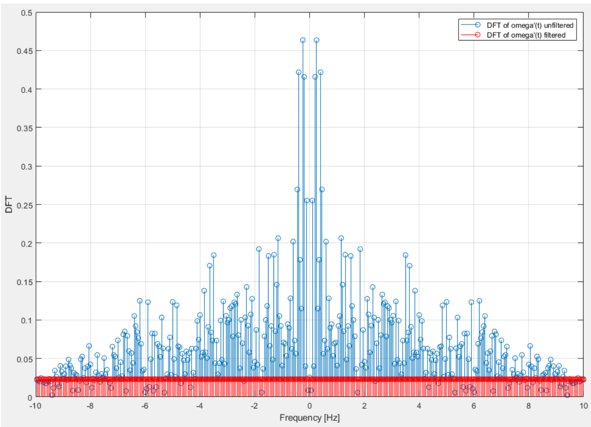


Figure 14: Plot of DFT($omega'(t)$) and DFT($omega'_f(t)$)

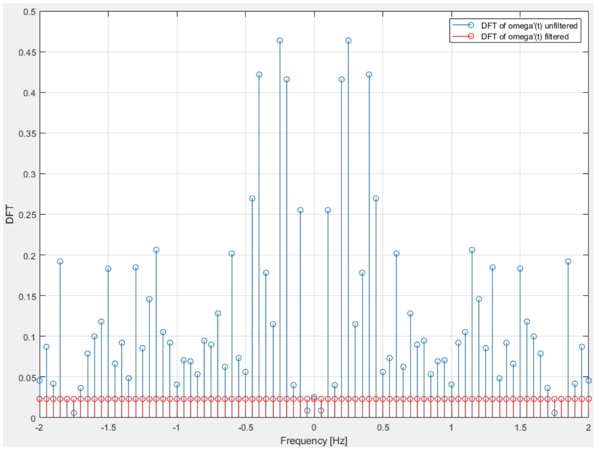


Figure 15: Plot of DFT($omega'(t)$) and DFT($omega'_f(t)$) zoomed

To get the plots :

```
wd_f_dft=zeros(N2,1);
for i=1:N2
    for k=1:N2
        wd_f_dft(i)=wd_f_dft(i)+wd_f(k)*exp(-1i*2*pi*m*k/N2);
    end
end

figure(11)
```

```
stem(f_2,abs(wd_dft)/N2), hold on
stem(f_2,abs(wd_f_dft)/N2, 'r'), hold off
xlim([-2 2])    % Added to get the zoomed between -2 ad 2
grid on
xlabel('Frequency [Hz]')
ylabel("DFT")
legend({"DFT of omega'(t) unfiltered", "DFT of omega'(t) filtered
   "})
```