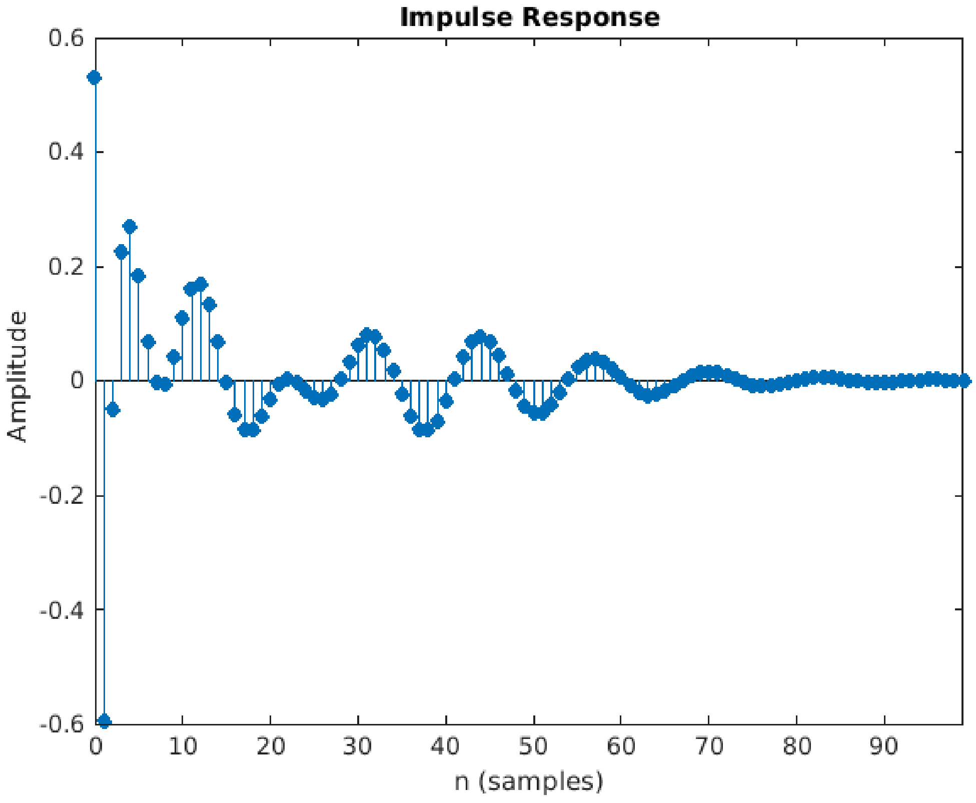
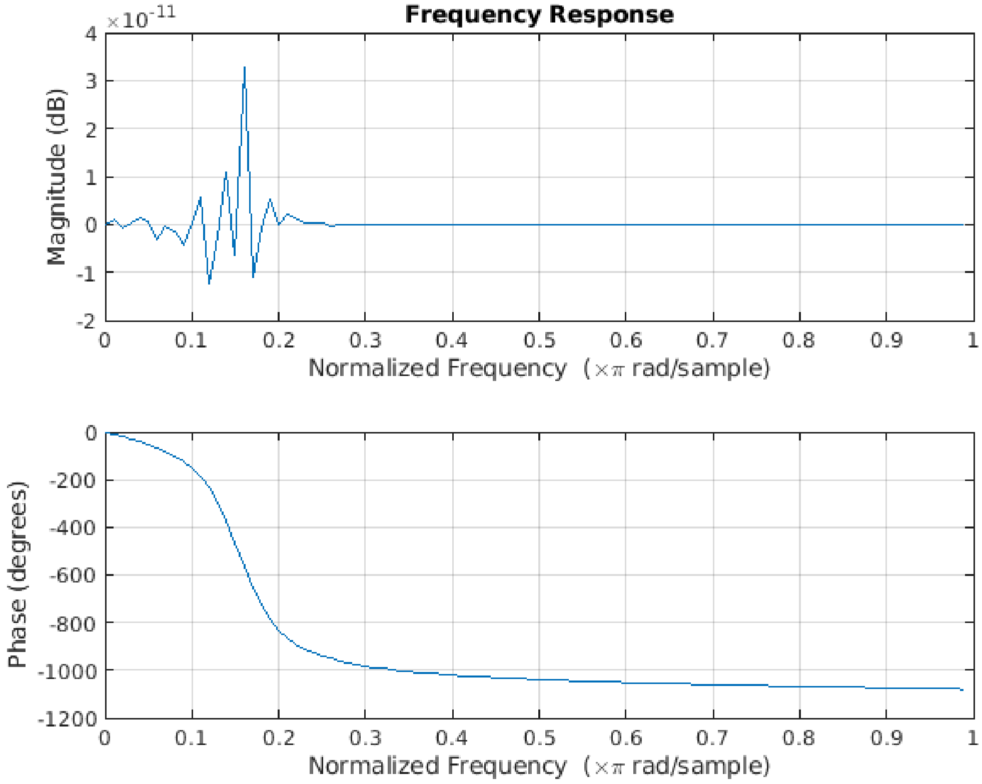
Table of Contents

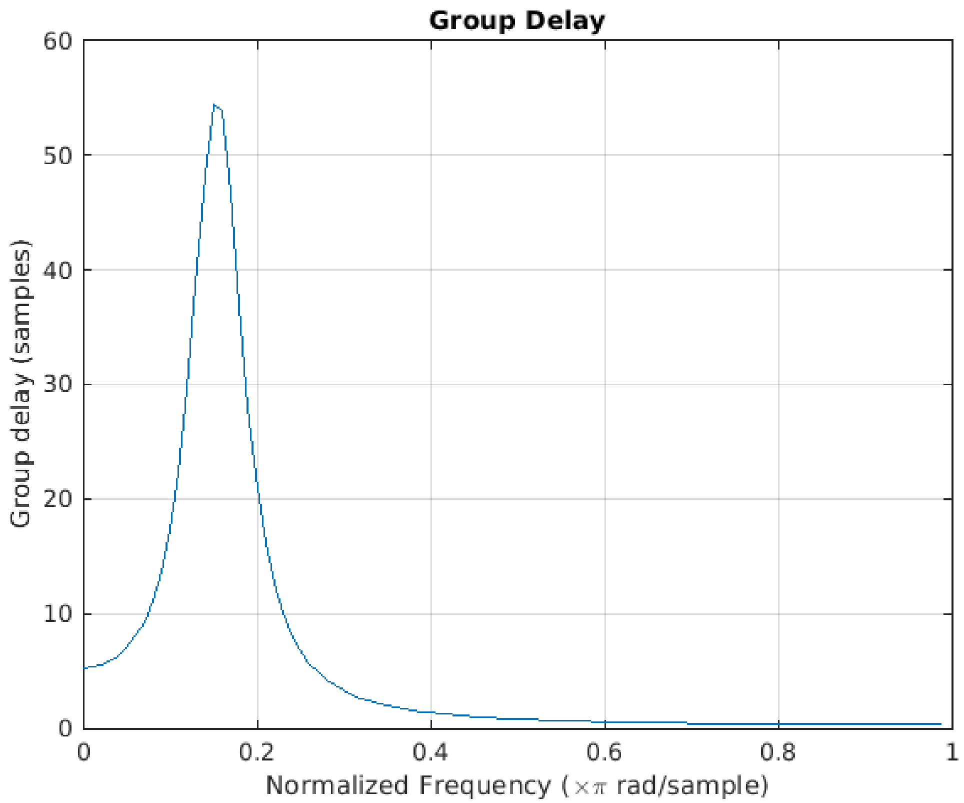
```
%Allister Liu, Amy Leong
%DSP Project 2

clc;
clear all;
close all;

load ('projIA.mat');        %load in the file and play the audio.
 Audio is used to compare with the audios for DF1, DF1 SOS, DF2 SOS,
 DF2 Transposed SOS
sound(speech, fs);
```

Part A

```
[h,t]=impz (b,a);           % graph the first 100 samples when N=1 for
 the impulse response, frequency response, group delay
figure;
impz(b,a,100);
title ('Impulse Response');

figure;
freqz(b,a,100);
title('Frequency Response');

figure;
grpdelay(b,a,100);
title('Group Delay');
```

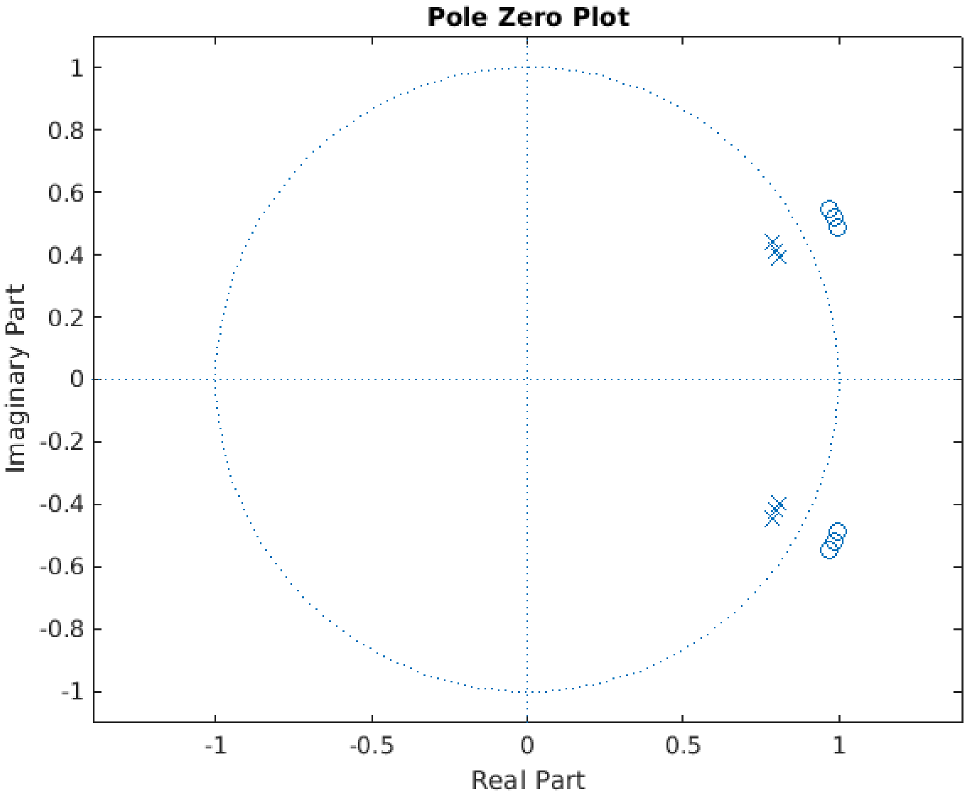

Impulse Response
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
Amplitude
0
10
20
30
40
50
60
70
80
90
n (samples)


Frequency Response
×10^-11
4
3
2
1
0
-1
-2
Magnitude (dB)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalized Frequency (×π rad/sample)
0
-200
-400
-600
-800
-1000
-1200
Phase (degrees)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalized Frequency (×π rad/sample)



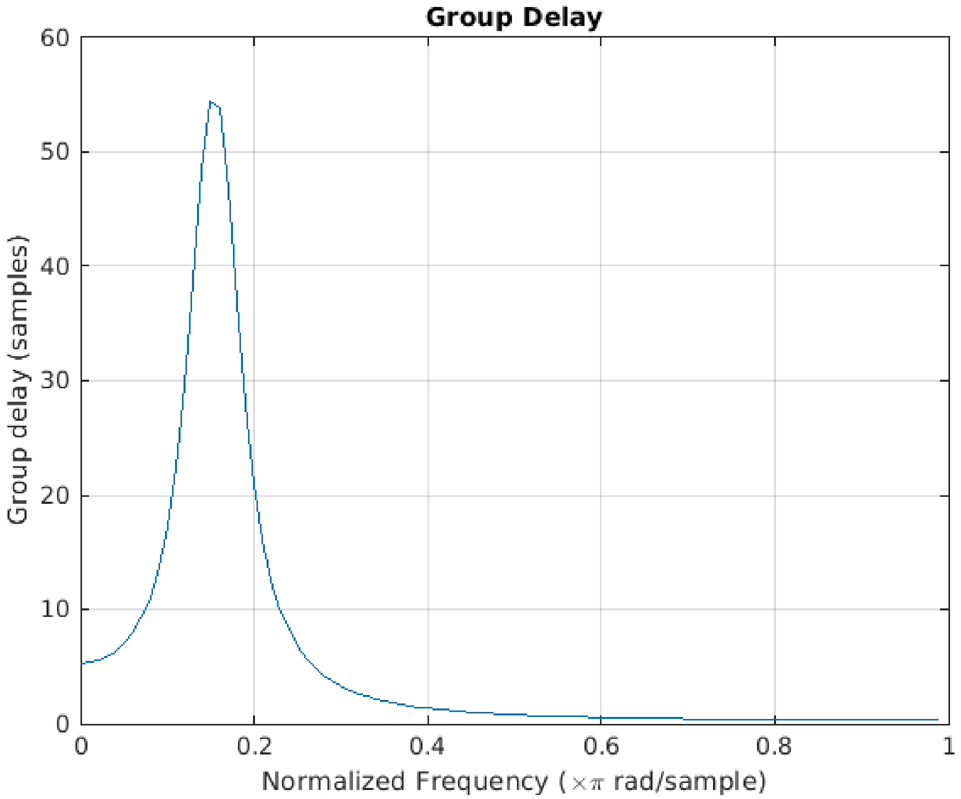
Part B

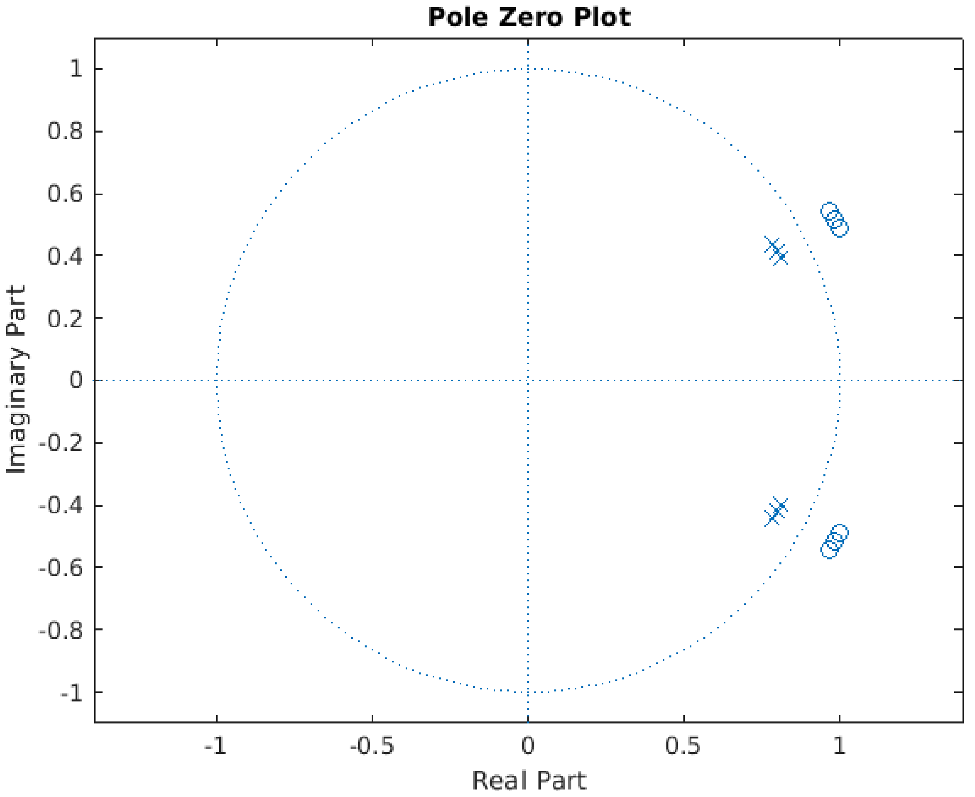
```
figure;
zplane(b,a);                      %graph the pole zero plot for N=1 using
 built in matlab function zplane
title ('Pole Zero Plot');
```



Part C

```
y=filter(b,a,speech);
sound(y,fs);

%there is little to none audio distortion. When n=1, the audio with
 the all
%pass filter sounded relatively the same as the orignal
```

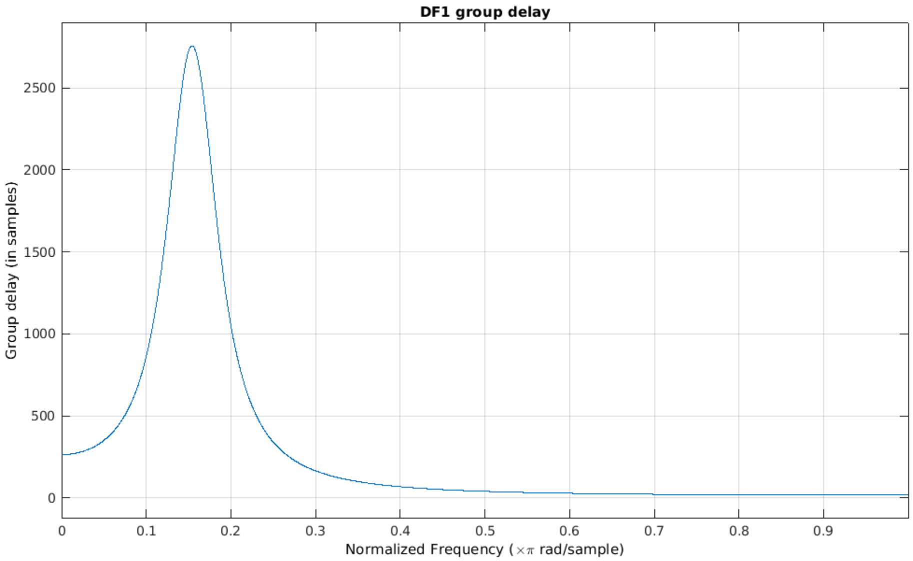

Group Delay
Group delay (samples)
60
50
40
30
20
10
0
0
0.2
0.4
0.6
0.8
1
Normalized Frequency (×π rad/sample)

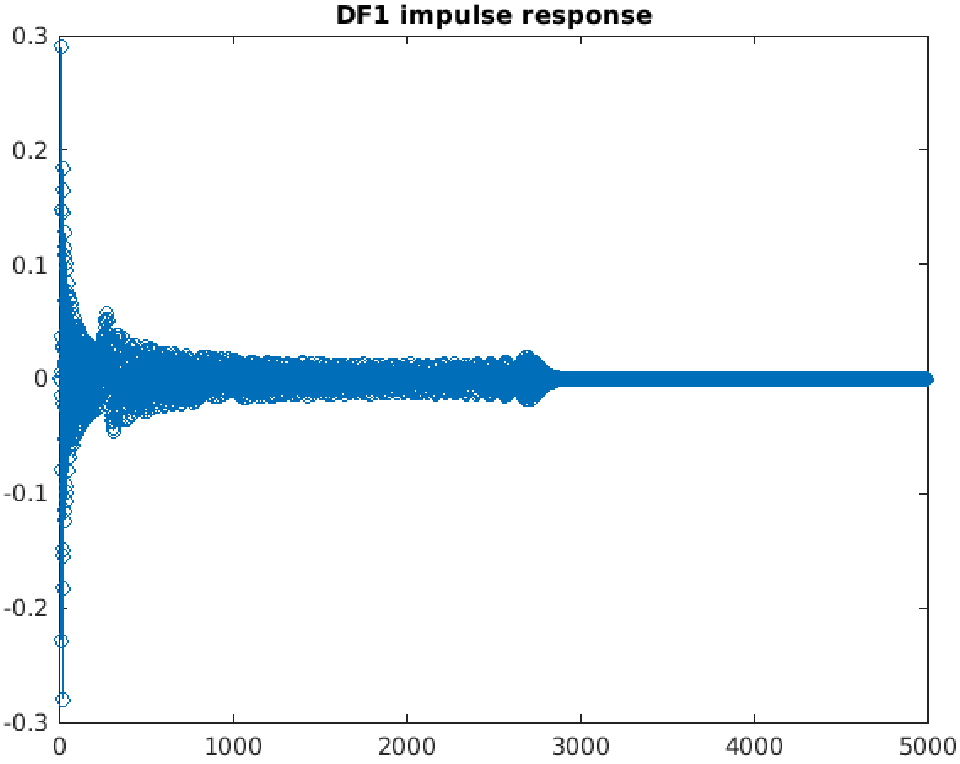

Pole Zero Plot
Imaginary Part
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-1
-0.5
0
0.5
1
Real Part

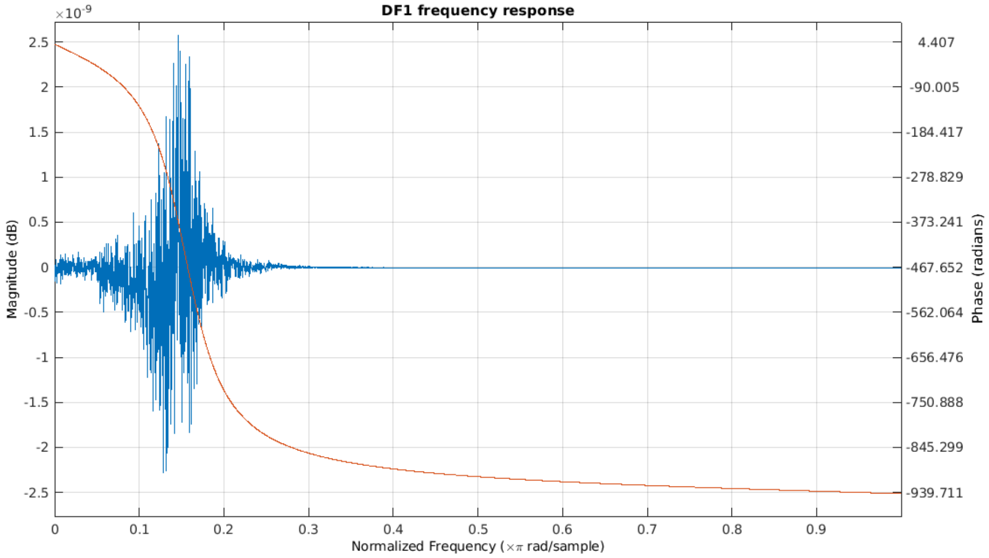
Part D and Part E:

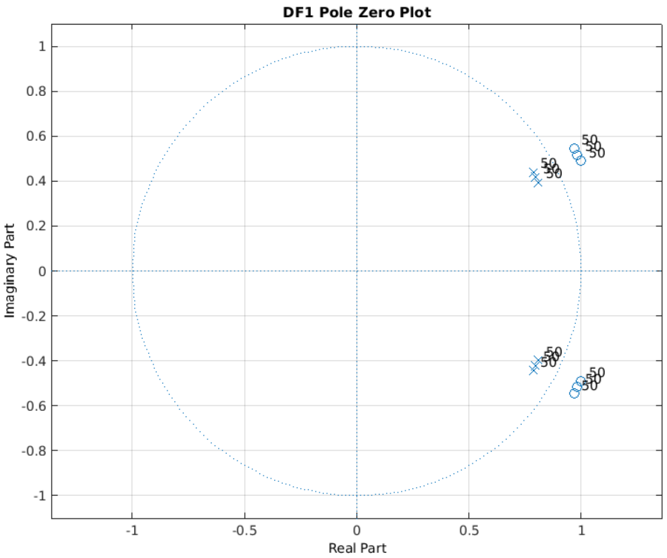
DF1 with machine precision

```
hd1=dfilt.df1(b,a);
df1_cascade= dfilt.cascade(repmat(hd1,1,50));    %uses cascade for
 machine precision with N=50

%figure;
grpdelay(df1_cascade, 5000);      %graph the group delay, impulse
 response, frequency response, and pole zero plot for the first 5000
 samples
title('DF1 group delay');

stem(impz(df1_cascade, 5000));
title ('DF1 impulse response');

freqz(df1_cascade, 5000);
title('DF1 frequency response');

zplane(df1_cascade);
title('DF1 Pole Zero Plot');

df1_filter=filter(df1_cascade,speech);  %processes the speech file
 with filter. there is audio distortion
soundsc(df1_filter,fs);
```



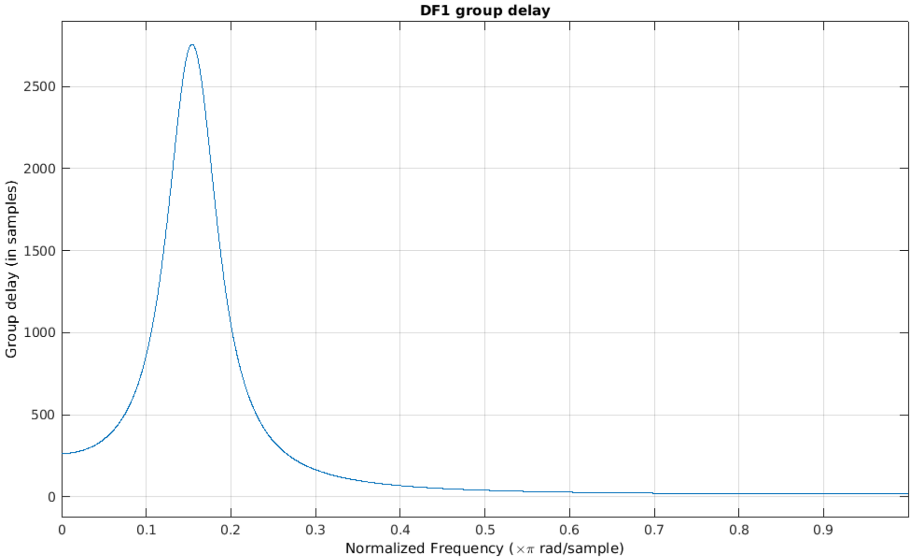

DF1 impulse response
0.3
0.2
0.1
0
-0.1
-0.2
-0.3
0
1000
2000
3000
4000
5000

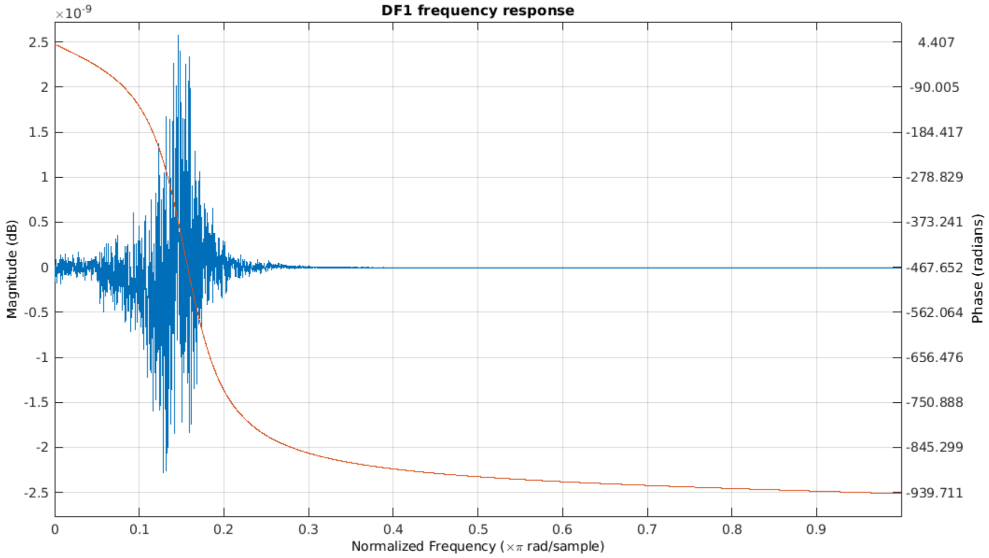

DF1 frequency response
×10^-9
2.5
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-2.5
Magnitude (dB)
4.407
-90.005
-184.417
-278.829
-373.241
-467.652
-562.064
-656.476
-750.888
-845.299
-939.711
Phase (radians)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
Normalized Frequency (×π rad/sample)

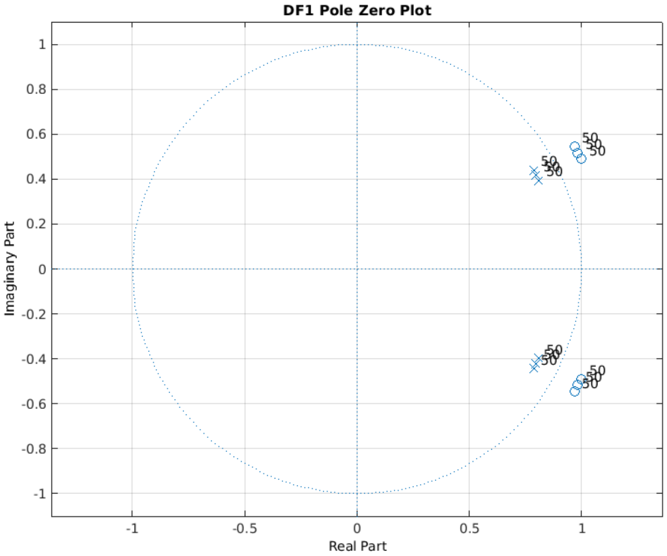


DF1 Alternative method

```
a1=a;
b1=b;

for N=1:50                          %foiling is used instead of the
 cascading function. the numerators and denominators (a and b) are
 foiled
    a1=conv(a1,a);                  %Multiplication is the same as conv. the
 for loop gets N=50
    b1=conv(b1,b);
end

alternate_df1=dfilt.df1(b1,a1);

%grpdelay(alternate_df1,5000);
%impz(alternate_df1,5000);
%freqz(alternate_df1,5000);
%zplane(alternate_df1);

alternate_df1_filter=filter(alternate_df1, speech);
%soundsc(alternate_df1_filter, fs);

%This method does not work. The audio is inaudible
```

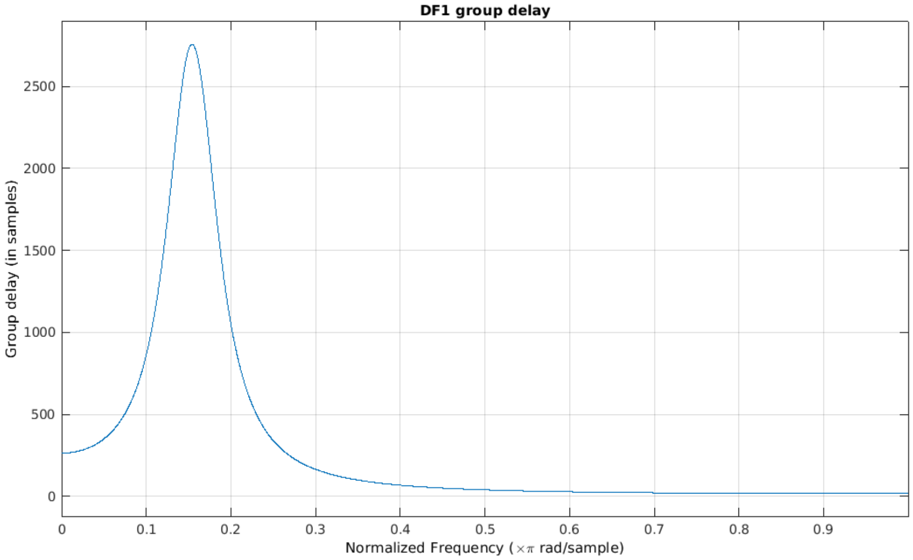

DF1 group delay
Group delay (in samples)
0
500
1000
1500
2000
2500
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
Normalized Frequency (×π rad/sample)

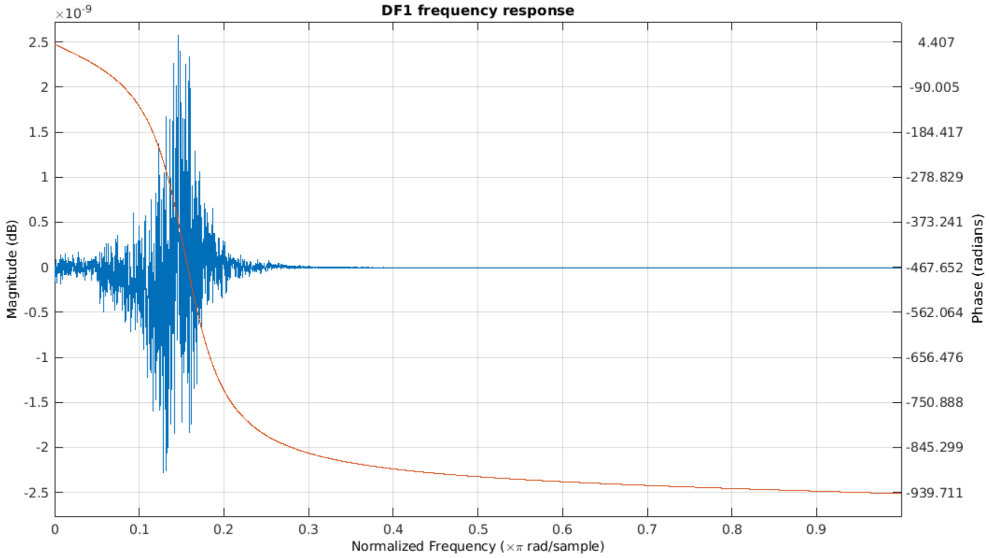

DF1 frequency response
×10⁻⁹
Magnitude (dB)
2.5
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-2.5
Phase (radians)
4.407
-90.005
-184.417
-278.829
-373.241
-467.652
-562.064
-656.476
-750.888
-845.299
-939.711
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
Normalized Frequency (×π rad/sample)

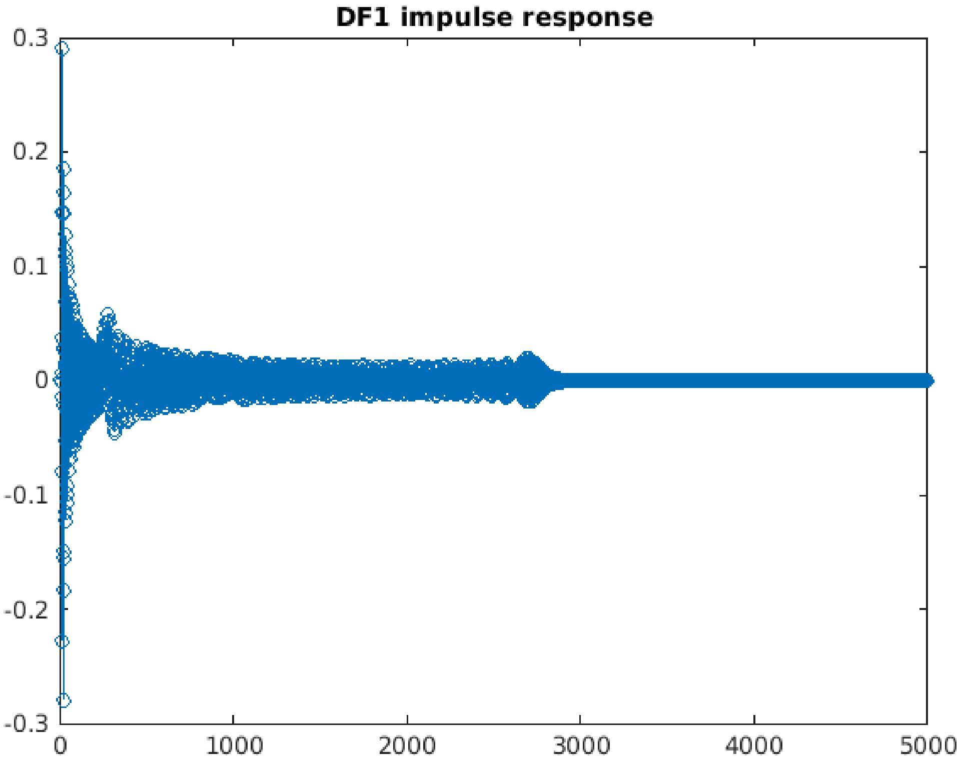


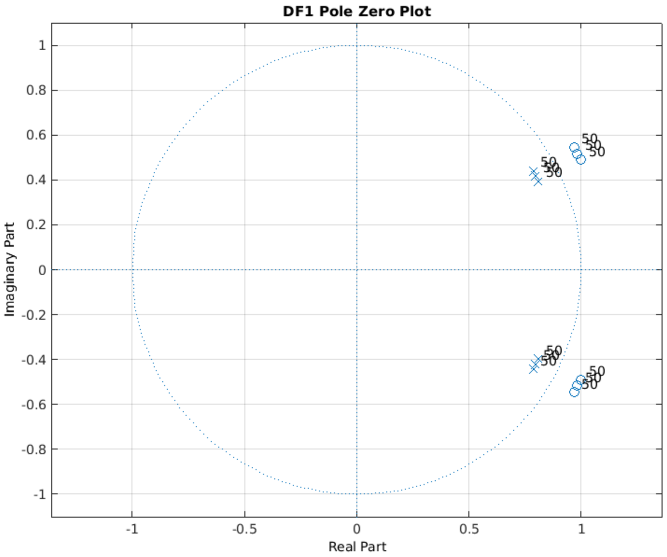
DF1 SOS

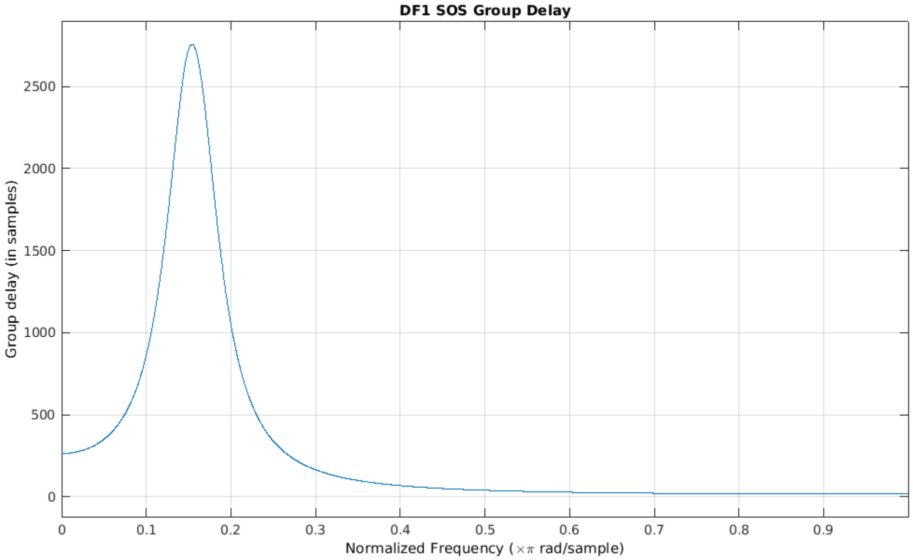
```
hd1_sos= sos(hd1);                    %uses built in matlab function sos
 to get a quantized filter Hq2 that has second-order sections and the
 dft2 structure.
df1_sos=dfilt.cascade(repmat(hd1_sos,1,50));

grpdelay(df1_sos,5000);
title('DF1 SOS Group Delay');

impz(df1_sos,5000);
title('DF1 SOS Impulse Response');

freqz(df1_sos,5000);
title ('DF1 SOS Frequency Response');

zplane (df1_sos);
title('DF1 SOS Pole Zero Plot');

df1_sos_filter=filter(df1_sos,speech);
soundsc(df1_sos_filter,fs);
```

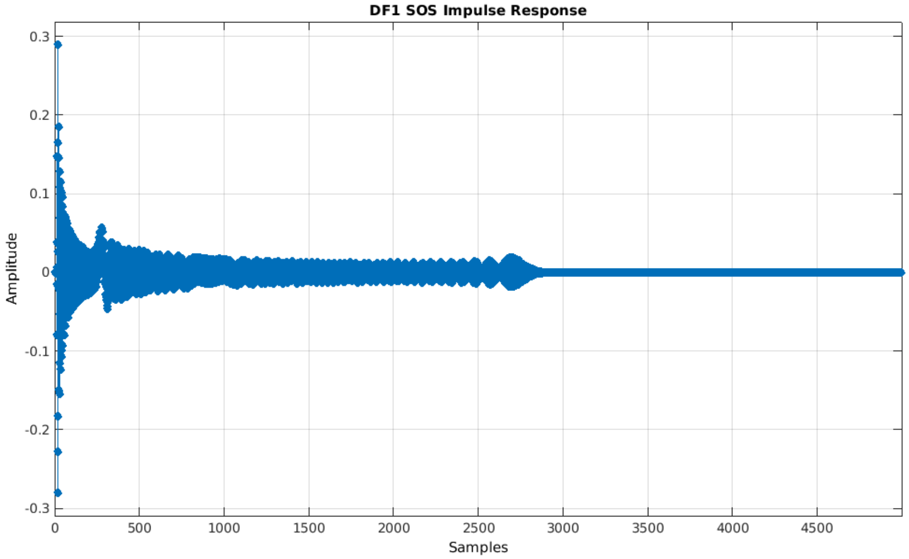

DF1 group delay
Group delay (in samples)
2500
2000
1500
1000
500
0
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
Normalized Frequency (×π rad/sample)

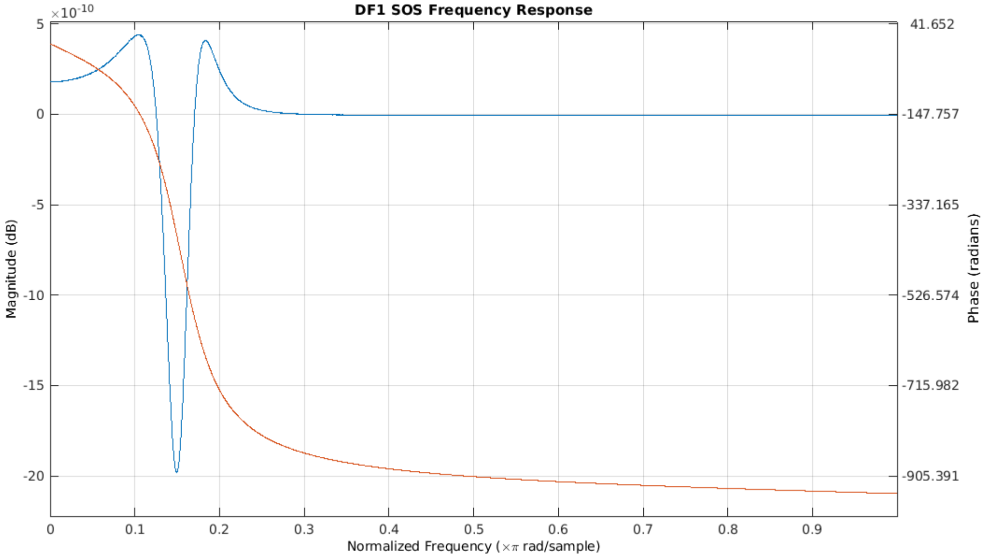

×10^-9
DF1 frequency response
Magnitude (dB)
2.5
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-2.5
Phase (radians)
4.407
-90.005
-184.417
-278.829
-373.241
-467.652
-562.064
-656.476
-750.888
-845.299
-939.711
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
Normalized Frequency (×π rad/sample)

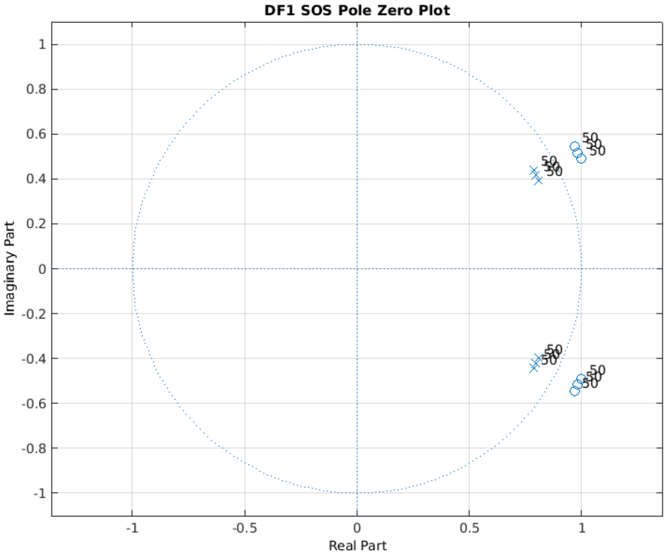

DF1 impulse response
0.3
0.2
0.1
0
-0.1
-0.2
-0.3
0
1000
2000
3000
4000
5000


DF1 Pole Zero Plot
Imaginary Part
Real Part
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-1
-0.5
0
0.5
1
50
50
50
50
50
50
50
50
50
50
50
50


DF1 SOS Group Delay
Group delay (in samples)
0
500
1000
1500
2000
2500
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
Normalized Frequency (×π rad/sample)


DF1 SOS Impulse Response
Amplitude
0.3
0.2
0.1
0
-0.1
-0.2
-0.3
0
500
1000
1500
2000
2500
3000
3500
4000
4500
Samples





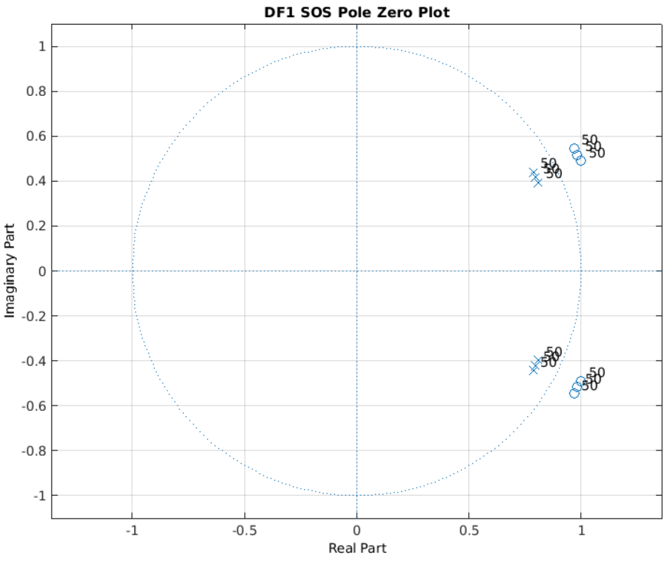
DF2

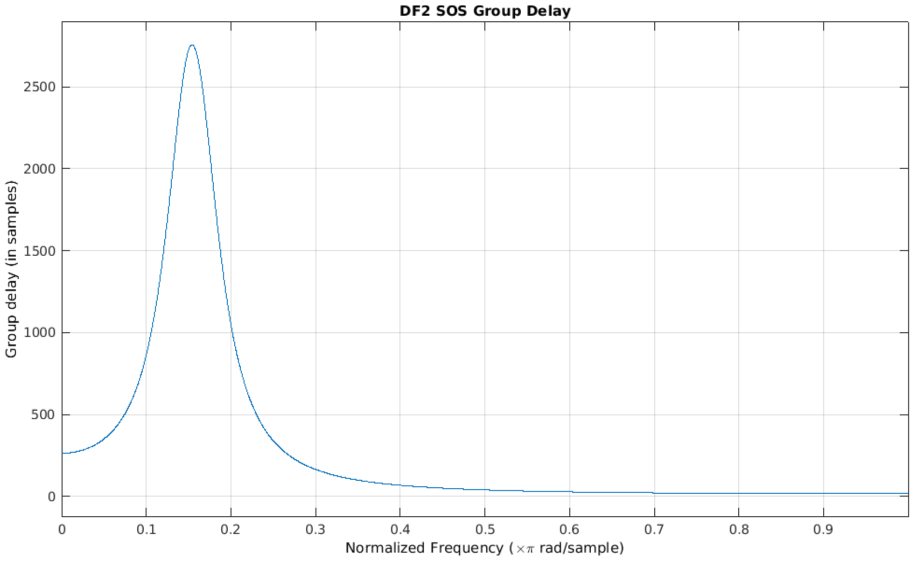
```
hd2=dfilt.df2(b,a);              %uses built in matlab functions to get
 the df2. Result is used for df2 sos
df2_cascade= dfilt.cascade(repmat(hd2,1,50));
```

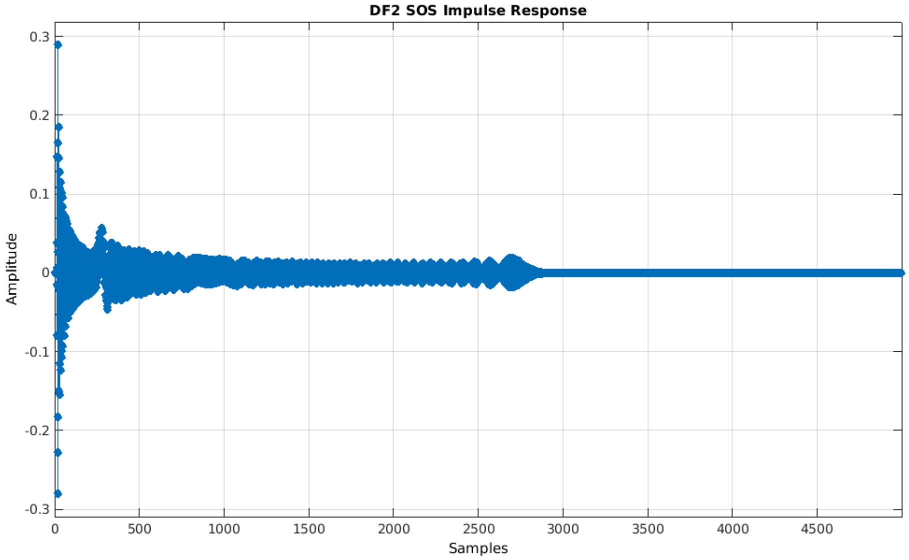
DF2 SOS

```
hd2_sos= sos(hd2);
df2_sos=dfilt.cascade(repmat(hd2_sos,1,50));

grpdelay(df2_sos,5000);
title ('DF2 SOS Group Delay');
```

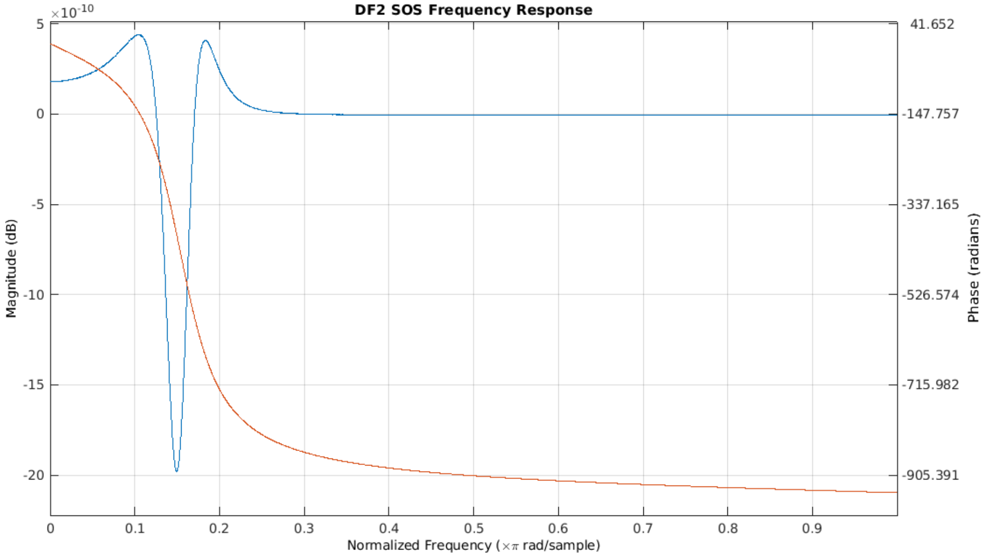
```
impz(df2_sos,5000);
title('DF2 SOS Impulse Response');

freqz(df2_sos,5000);
title('DF2 SOS Frequency Response');

zplane (df2_sos);
title('Df2 SOS Pole Zero Plot');

df2_sos_filter=filter(df2_sos,speech);
soundsc(df2_sos_filter,fs);
```



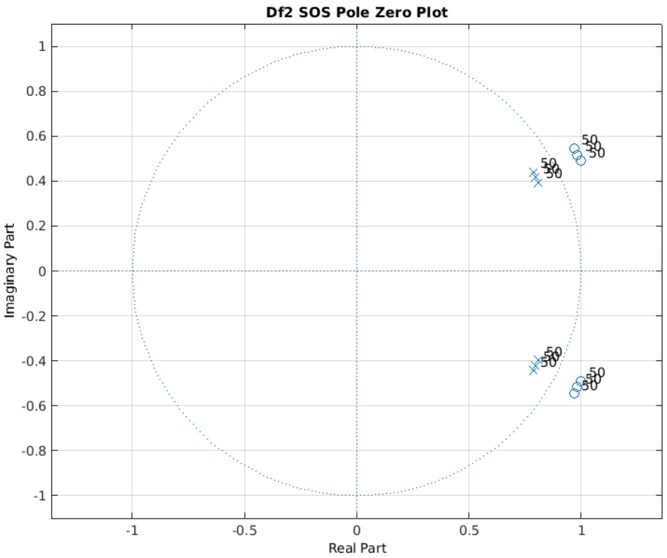




DF2 SOS Impulse Response
Amplitude
0.3
0.2
0.1
0
-0.1
-0.2
-0.3
0
500
1000
1500
2000
2500
3000
3500
4000
4500
Samples

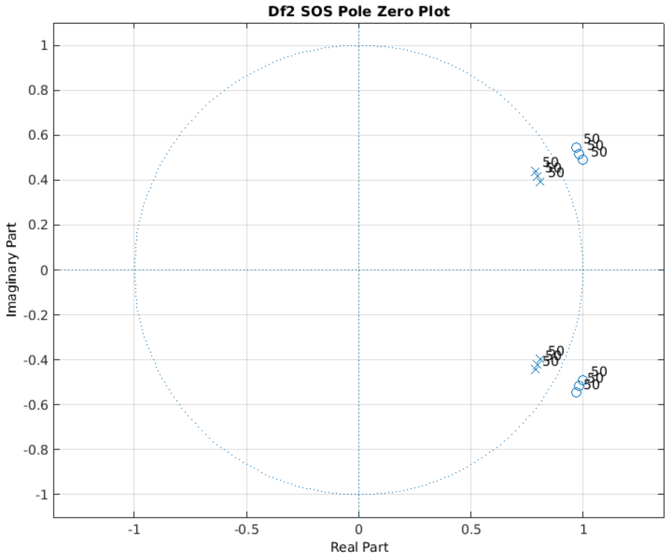


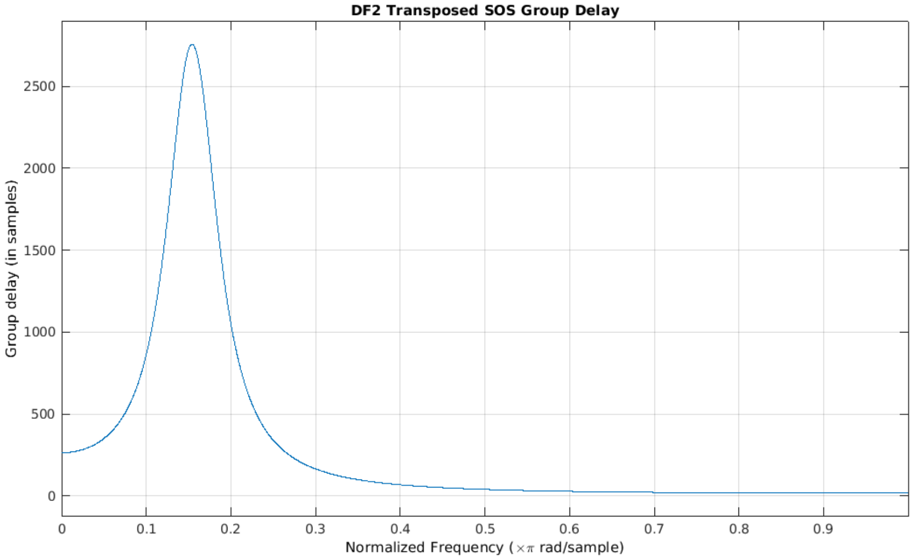
DF2 SOS Frequency Response
$\times 10^{-10}$
Magnitude (dB)
5
0
-5
-10
-15
-20
Phase (radians)
41.652
-147.757
-337.165
-526.574
-715.982
-905.391
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
Normalized Frequency ($\times\pi$ rad/sample)

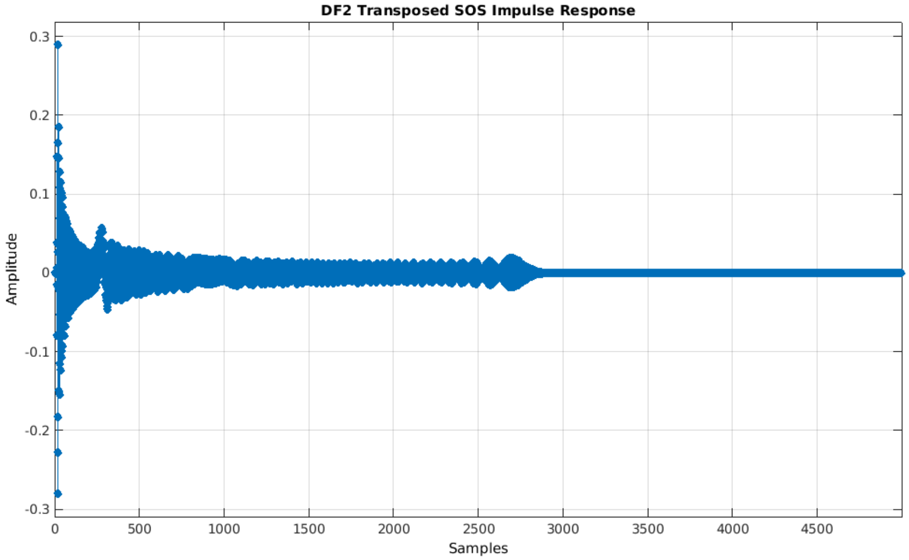


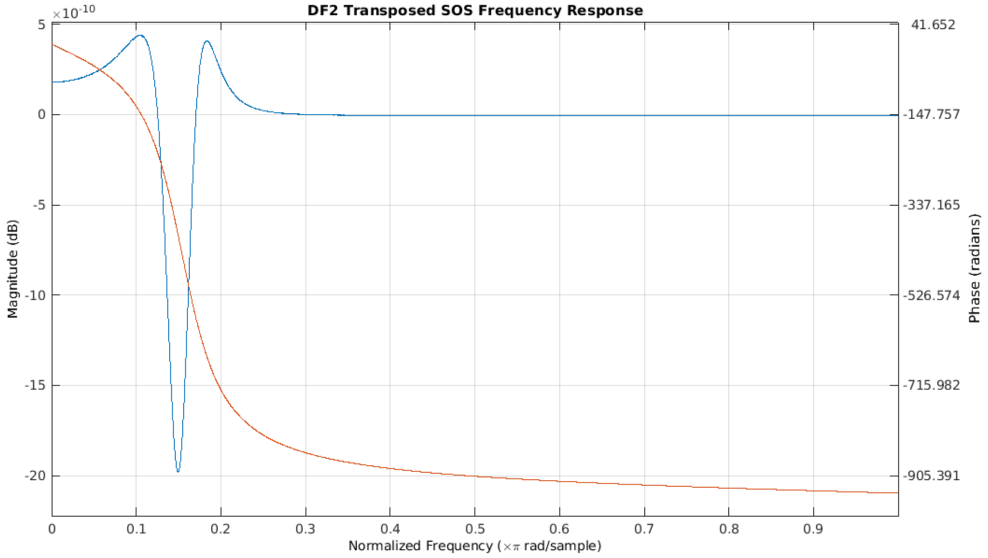
DF2 Transposed SOS

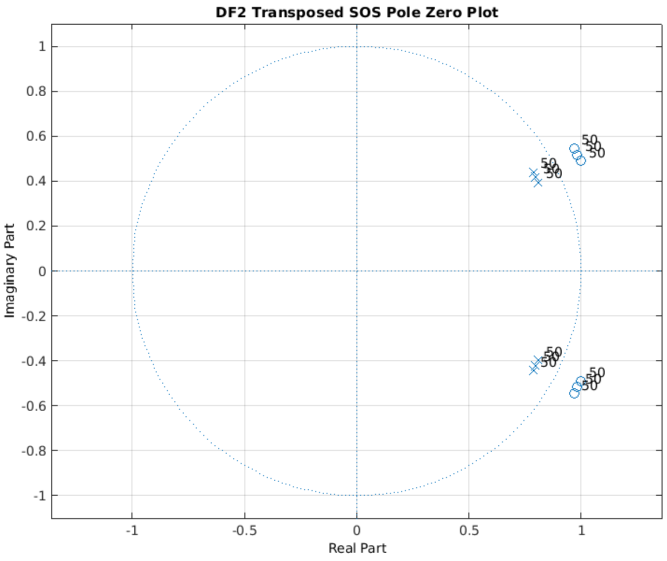
```
[sos,g]=tf2sos(b,a);                              %uses built in matlab
 functions tf2sos and dfilt.df2tsos to get the df2 transposed sos
hd2_sos_t=dfilt.df2tsos(sos,g);
final_hd2_sos_t=dfilt.cascade(repmat(hd2_sos_t,1,50));

grpdelay(final_hd2_sos_t,5000);
title('DF2 Transposed SOS Group Delay');

impz(final_hd2_sos_t,5000);
title ('DF2 Transposed SOS Impulse Response');

freqz(final_hd2_sos_t,5000);
title ('DF2 Transposed SOS Frequency Response');

zplane (final_hd2_sos_t);
title ('DF2 Transposed SOS Pole Zero Plot');

df2_sos_t_filter=filter(final_hd2_sos_t,speech);
soundsc(df2_sos_t_filter,fs);
```


Df2 SOS Pole Zero Plot
Imaginary Part
Real Part
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-1
-0.5
0
0.5
1
50
50
50
50
50
50
50
50
50
50
50
50


DF2 Transposed SOS Group Delay
Group delay (in samples)
Normalized Frequency (×π rad/sample)
2500
2000
1500
1000
500
0
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9


DF2 Transposed SOS Impulse Response
Amplitude
0.3
0.2
0.1
0
-0.1
-0.2
-0.3
0
500
1000
1500
2000
2500
3000
3500
4000
4500
Samples


DF2 Transposed SOS Frequency Response
×10-10
Magnitude (dB)
5
0
-5
-10
-15
-20
Phase (radians)
41.652
-147.757
-337.165
-526.574
-715.982
-905.391
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
Normalized Frequency (×π rad/sample)



Part E explanation

At N=50 for DF1, DF1 SOS, DF2 SOS, and DF2 Transposed SOS, all the audio had this weird distorted sound. However, you can still make out the words

```
%in the message. The folk theorem is false. Addiitonally, the graphs for
%these four cases look the same. The reason for the distortion is because
%of the group delay. When comparing the group delay from N=1 to N=50, the
%group delay for N=50 increased.
```

Published with MATLAB® R2020b