E3DSB Miniprojekt C

Simon Alexander Alsing, 201304202
Lasse Østerberg Sangild, 20114274

December 12, 2016

Contents

1 Introduction

In this miniproject a man's and a woman's speach will be under study. This project will aim to test whether applying a filter of the opposite gender and enhancing these filtered frequencies, would make it sound more like the opposite gender.

Expectations to this project is that the filtered voice will make a noticable change but it will most likely not make the woman sound like a man and vice versa.

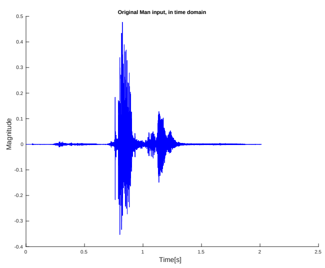
2 Analysis

The following analysis will be presented in two parts, one which will be before filtering with time domain response of the two signals and then the frequency analysis of these with poles and zeroes. Next an appropriate filter will be applied and the two signals will again be analysed as previously explained.

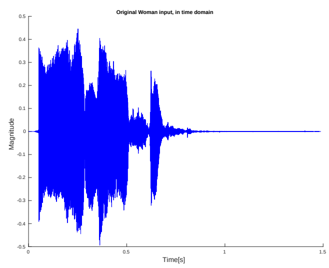
2.1 Before filtering

2.1.1 Time domain analysis

First a formal analysis of the man and the woman's voice saying "Hej Lasse" in the time domain. In fig. 1a the signal for the man can be seen, with the time in seconds on the x-axis and the magnitude on the y-axis and similarily the womans signal is shown in fig. 1b.



(a) The time domain signal of the man. On the x-axis time in seconds, on the y-axis magnitude



(b) The time domain signal of the woman. On the x-axis time in seconds, on the y-axis magnitude

Figure 1: The time samples of a man and a woman saying "Hej Lasse".

2.1.2 Fourier transform with FFT

The fourier transform will show the characteristics of the male and female voice. In fig. 2 the fourier transform based of the two previous signals seen in fig. 1 - in red the male signal and in blue the female signal. It is clear to see that the primary part of the voice of the male is located at lower frequencies than the voice of the woman. In the following sections we will use 2800 Hz to 4500 Hz as the range of the male voice, while the female voice is in the range of 3500 Hz to 8000 Hz.

2.2 Filtering

2.2.1 Highpass filter

Next up is filtering the signal, which will be done with a highpass filter. From fig. 2 a cutoff estimation of 3500 Hz for the female voice and 2800 Hz for the male voice has been chosen. These two cutoff frequencies will be switched for the two signals to see which effect this will have on the signals.

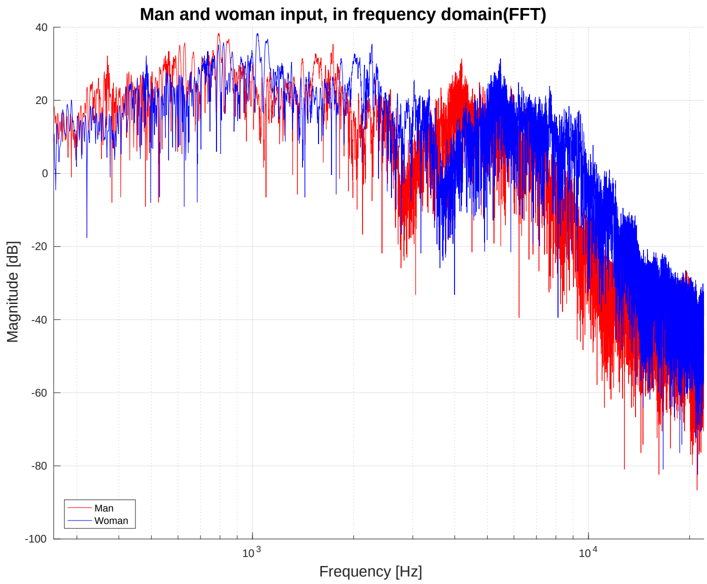


Figure 2: The frequency analysis of the man and woman. On the x-axis frequency in Hz, on the y-axis magnitude in dB

The highpass filter is implemented as a function in matlab and can be seen in the following code example:

```
function [figfreq, figHz] = HP(cutFreq, Fs, y, figOrgFreq, figfreqz)
N = length(y);
frequency_samples = [0:Fs/N:(Fs-(Fs/N))];
HighPass = cutFreq/(Fs/2);
HiPass = fir1(70,HighPass,'high');
figfreq = figure(figfreqz);
hold off
title('Filter characteristics');
hold on
freqz(HiPass,1);

% save and visualise
tic
yHP = filter(HiPass,1,y);
toc
name = ['HP_', num2str(cutFreq), '_Hz.flac'];
audiowrite(name, yHP, Fs);
YHP = fft(yHP);
YdBHP = 20*log10(abs(YHP));

% Plot of discrete fourier transform
```

```
figHz = figure(figOrgFreq);
hold off
title(['Original and HP(512 Hz)', ' in frequency domain(FFT)']);
hold on
semilogx(frequency_samples(1:N/2), YdBHP(1:N/2), 'b');
legend({'original', 'highpass'}, 'FontSize', 16);
legend('Location','best');
end
```

Notice that the filtered signal is stored as a .flac file, which also will be displayed. The filtered signal can be seen in fig. 3 for the male and female in fig. 4. The filtered signals are somewhat disappointing to listen to, because it simply sounds like

FiXme Note: Like what? hvorfor konklusion her?

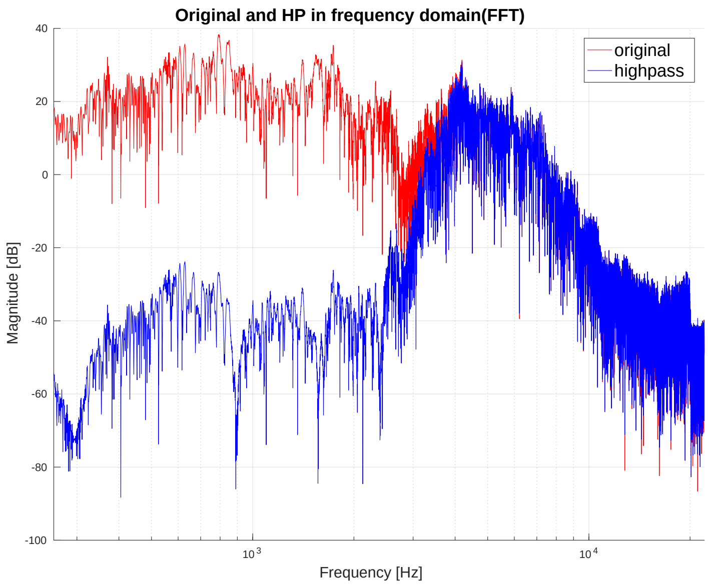


Figure 3: The highpass filtered signal of the male input. On the x-axis frequency in Hz, on the y-axis magnitude in dB

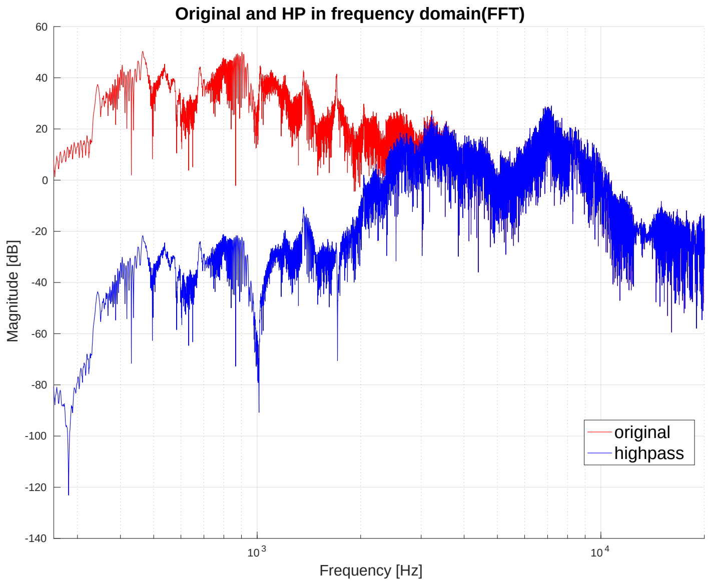


Figure 4: The highpass filtered signal of the female input. On the x-axis frequency in Hz, on the y-axis magnitude in dB

2.2.2 Bandpass filter (FIR)

In the previous section a highpass filter was applied to test which effect it had on the two voices. Now a bandpass filter will be applied with an FIR implementation to the two voices to see which differences there will be. The start of the bandpass filter will again be 2800 Hz for the female signal and 3500 Hz for the male signal, while the cutoffs will be at 8000 Hz for the male signal and 6500 Hz for the female signal. The male filtered signal can be seen in fig. 5 and the female can be seen in fig. 6

The code for the bandpass filter is once again implemented in a function and the implementation is as follows:

```
function [figW, figHz] = BP(freqRange, Fs, y, figOrgFreq)
N = length(y);
frequency_samples = [0:Fs/N:(Fs-(Fs/N))];
BandPass = freqRange/(Fs/2)
BPass = fir1(100,BandPass,'bandpass');%, kaiser(51,0.5));
figW = figure;
hold on
title('Filter characteristics');
freqz(BPass,1);

% Gem og visualiser frekvensændringen
tic
yBP = filter(BPass,1,y);
toc
name = ['BP_', num2str(freqRange(1)), '_Hz_to_', num2str(freqRange(2)), '_Hz.mp4'];
%audiowrite(name, yBP, Fs);
YBP = fft(yBP);
YdBBP= 20*log10(abs(YBP));

% Plot of discrete fourier transform
figHz = figure(figOrgFreq);
hold off
title({['Original and Bandpass'], [num2str(freqRange(1)), ' Hz to ',...
   num2str(freqRange(2)), ' Hz in frequency domain(FFT)']});
hold on
semilogx(frequency_samples(1:N/2), YdBBP(1:N/2), 'b');
legend({'original', 'bandpass'}, 'FontSize', 16);
legend('Location','best');
end
```

2.2.3 Magnitude and phase effect of filtering

From the section "Bandpass filter (FIR)" the frequency response of the filters are analyzed and this data will be analyzed with the function freqz from matlab to tell the magnitude and phase effects of the bandpass filter. In fig. 7 the effects of phase and magnitude are shown for the female signal and in fig. 8 the male

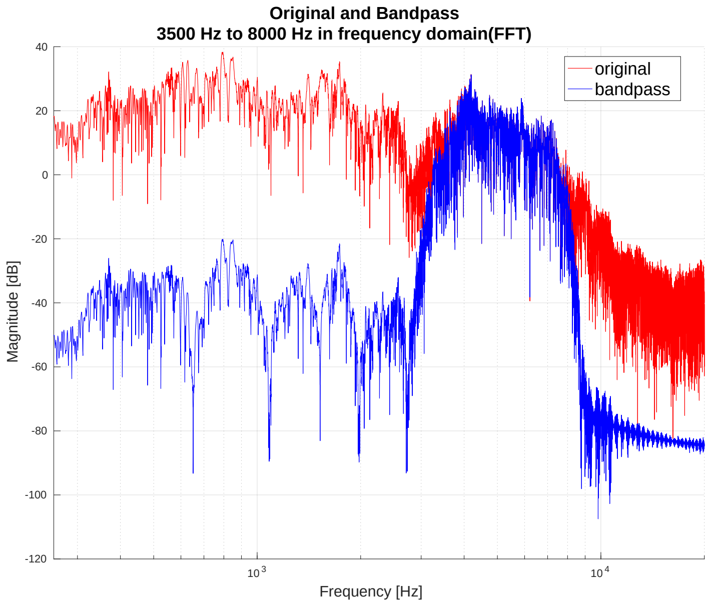


Figure 5: Bandpass from 3500 Hz to 8000 Hz to the male signal. On the x-axis frequency in Hz, on the y-axis magnitude in dB

bandpass filter. On female filter fig. 7 the phase shift is linear as expected, which is also the case for the male on fig. 8. They have different have a phase shift of $-2500°$ for the male and $-1500°$ for the female. The magnitude are damped with about 60 dB outside of the bandpass filter of both filters, which is clearly seen in the two signals after the bandpass filter (see fig. 5 and fig. 6)

In fig. 9 the pole-zero plot is shown for the FIR bandpass filter. From this we can see that it is a 100-order filter (100 in the middle, about 0,0) and that it is stable because the poles are located in 0,0 which is within the unitcircle (dotted circle).

2.2.4 Bandpass filter (IIR)

In section "Bandpass filter (FIR)" the bandpass filter were made with an FIR filter. In this section we will study the effects of the IIR bandpass filter with the same applied signals and frequencies as seen in section "Bandpass filter (FIR)". The filter implementation can be seen in the following code example:

In fig. 10 the male signal applied with the IIR bandpass filter is seen, while in fig. 11 the female is seen. The filter effects are found with the use of function `freqz` from the matlab library and can be seen in fig. 12. The stability of the filter is evaluated with matlab function `zplane`, and from fig. 13 it is evident that the filter is stable, because all of the poles (x'es) are within the unitcircle. A very noticable difference can be seen between this IIR implementation of the bandpass filters poles and the ones seen in the FIR implementation in fig. 9. Differences are also apparent when comparing the phase, on the FIR filter a linear phase is present (see fig. 8 and fig. 7) and the IIR phase which most

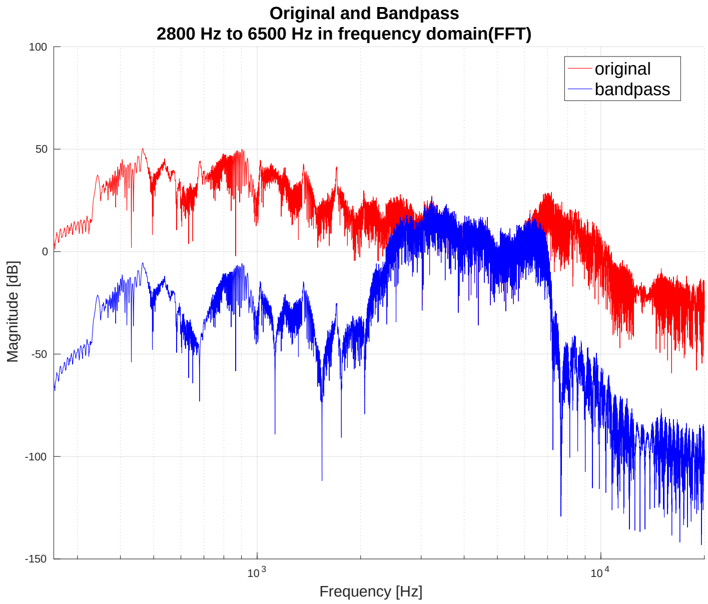


Figure 6: Bandpass from 2800 Hz to 6500 Hz to the female signal. On the x-axis frequency in Hz, on the y-axis magnitude in dB

certainly is not linear (see fig. 12).

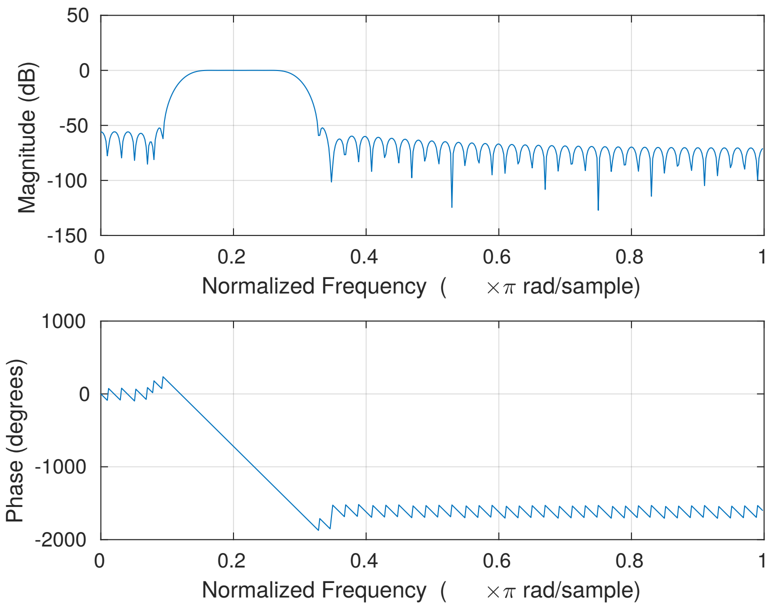


Figure 7: Freqz of the bandpass FIR filter - with the female frequencies

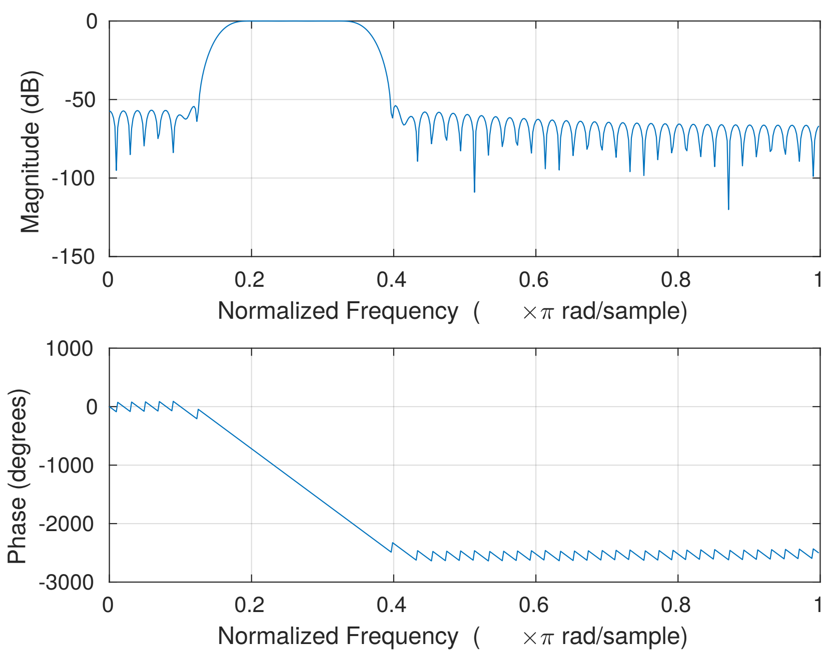


Figure 8: Freqz of the bandpass FIR filter- with the male frequencies

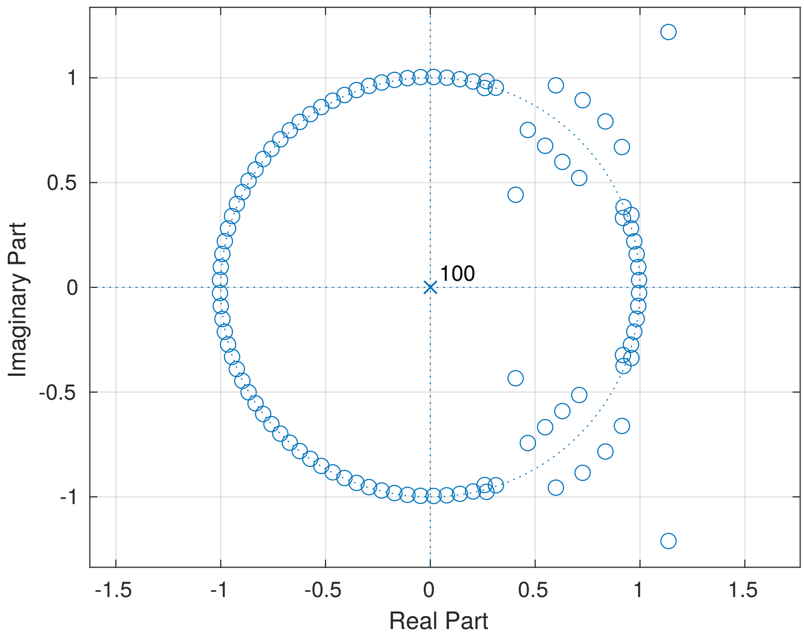


Figure 9: The pole-zero plot of the bandpass FIR filter with function zplane

```
function [figW, figHz] = IIRBP(freqRange, Fs, y, figOrgFreq)
N = length(y);
frequency_samples = [0:Fs/N:(Fs-(Fs/N))];
BandPass = freqRange/(Fs/2);
[b,a] = butter(5,BandPass,'bandpass');
figW = figure;
hold on
title('Filter characteristics');
freqz(b,a);
figure
zplane(b,a);

% Gem og visualiser frekvensændringen
tic
yBP = filter(b,a,y);
toc
name = ['IIRBP_', num2str(freqRange(1)), '_Hz_to_', num2str(freqRange(2)), '_Hz.mp4'];
%audiowrite(name, yBP, Fs);
YBP = fft(yBP);
YdBBP= 20*log10(abs(YBP));

% Plot of discrete fourier transform
figHz = figure(figOrgFreq);
hold off
title({['Original and Bandpass_{IIR}'], [num2str(freqRange(1)), ' Hz to ',...
   num2str(freqRange(2)), ' Hz in frequency domain(FFT)']});
hold on
semilogx(frequency_samples(1:N/2), YdBBP(1:N/2), 'b');
legend({'original', 'bandpass'}, 'FontSize', 16);
legend('Location','best');
end
```

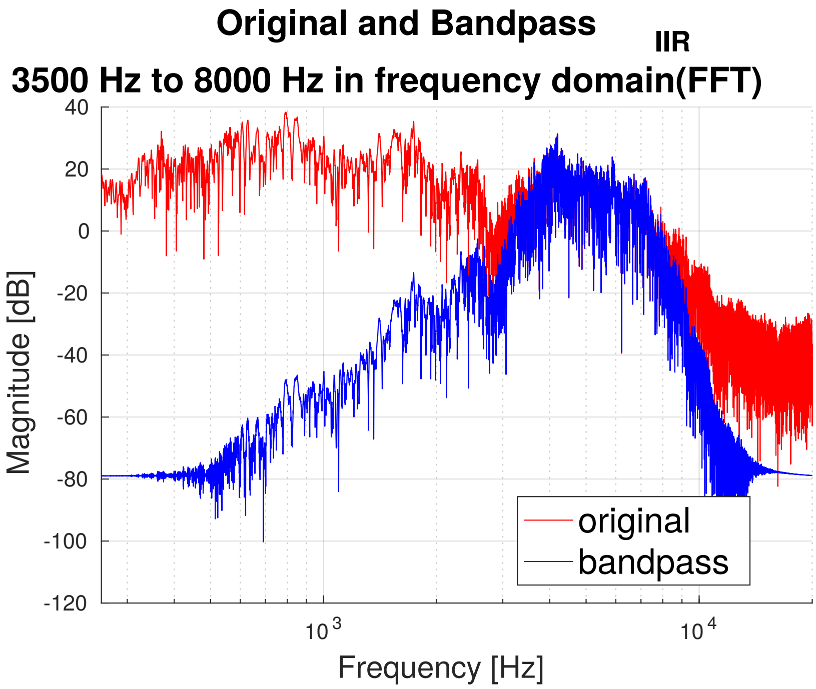


Figure 10: The IIR bandpass filter of the male signal

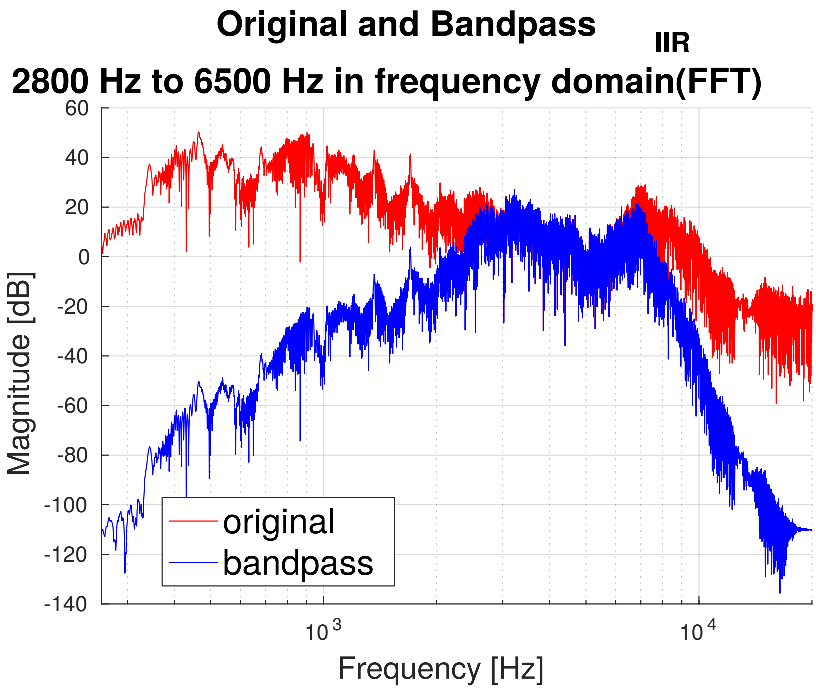


Figure 11: The IIR bandpass filter of the female signal

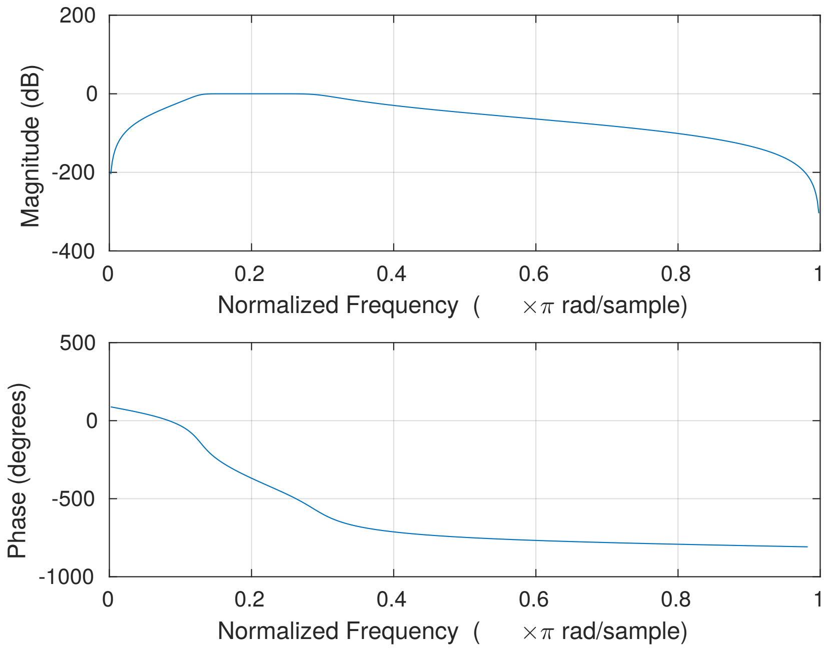


Figure 12: The Magnitude and phase effects of the IIR bandpass filter

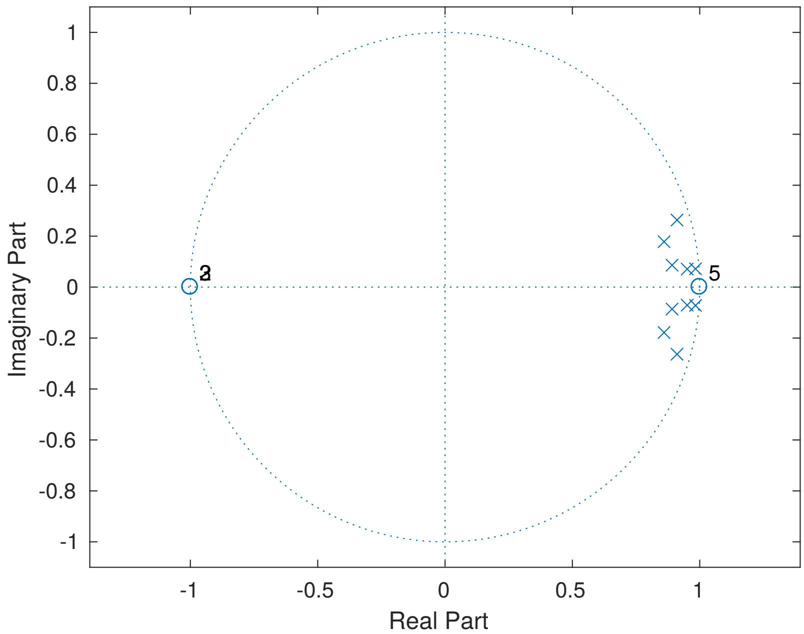


Figure 13: The pole-zero diagram of the IIR filter

3 Results

In the section "Analysis" the data analysis have been done, in this section a discussion upon the effects to the sound samples will be made. First off a highpass filter was applied. The first thing noticed was that the sound levels were lowered a lot. The voice sounded like if someone had recorded it from far away. When put on replay a couple of times, it is also apparent that the two samples have lost some detail in the voice - which would also be expected when filtering out some of the frequencies. None of the two samples sounded like it had "changed gender", it just sounds distorded and far away. The same applies for the bandpass and the male voice, the female does sound a bit different with the bandpass, it actually sounds like her voice is even further away than with the highpass filter. A better apporoach to doing this would be to have done some shifting after doing the fourier transform, because it is the frequencies that decide if we would think it as a male or female - the males tend to have lower frequencies, hence the female signal should not filter out the signals outside the male but should have moved the frequencies down to were the male frequencies lie.

From testing the filters (IIR and FIR) it became evident that the order of the filters were important. The FIR bandpass filter was first implemented with an order 5 and 10, which did not affect the signal as expected. When the filter order of the FIR filter was changed to 100, the outputs were as expected. The IIR bandpass filters were implemented as a fifth order filter, when applied this gave the expected results, hence an experience from this is that the FIR filter need a greater order to function as expected compared to the IIR, while the IIR filter can become unstable when using too high orders. When using higher orders of IIR filters, we found the poles to appear outside of the unit circle, making the filter unstable

4 Conclusion

This miniproject has proven that with the use of filters, we can manipulate the sound signal, but if you want to make the changes to the voice as the focus were, some more work has to be done - like moving the frequencies after doing the fourier transform. This project has not turned out to change the voices as expected but an important lesson regarding the order of IIR and FIR filters were learned.