

INGENIØRHØJSKOLEN AARHUS

DSB

E, IKT OG EP



Miniprojekt Del 3

Udarbejdet af:

Simon Thrane Hansen

Lars Hjerrild

Kasper Lauge Madsen

201500150

201409555

201409873

Underviser:

Lars G. Johansen

11. maj 2016

Indhold

1	Indledning	2
2	Teori	3
2.0.1	Hørerapperat	3
2.0.2	Z-transformation	3
2.0.3	IIR-filtre	4
2.0.4	FIR-filtre	4
2.0.5	Hanning vindue	4
3	Høreapperatet	6
3.1	Høreapperatet	6
3.2	Matlab kode for de forskellige funktioner	6
3.2.1	FIR-Båndpass funktion	6
3.2.2	IIR-bandpass filter	7
3.2.3	Weight funktion	9
3.2.4	Horeapparat	10
3.3	Resultat	11
4	Resultat og Diskussion	14
5	Konklusion	15

1. Indledning

Denne opgave ophandler design og implementering signalbehandlingen i et høreapparat i Matlab. Der er i opgaveformuleringen blevet stillet følgende krav: Opgaven skal indeholde væsentlige elementer fra hele kurset i DSB. I opgaven er der derfor blevet eksperimenteret med design af både IIR- og FIR-filtre.

2. Teori

I opgaven er der brugt forskellige digitale filtre og redskaber til at implementere digital signalbehandling til et hørerapperat. De forskellige filtre og redskaber, der er arbejdet med i opgaven er:

- Fast fourier transformation
- IIR-filtre
- FIR-filtre
- Hanning funktionen

2.0.1 Hørerapperat

Hørerapperatet er blevet implementeret ved at lave forskellige båndpas filtre, der opdeler Audio-signalet i forskellige frekvens dele. Disse båndpas filtre er blevet designet ved brug af IIR. Ønsket med disse blokke er at forstærke de frekvenser der ligger i menneskets tale frekvenser som ligger omkring 300Hz- 3.4kHz. Mennesket kan sagtens lave lyde udenfor dette men dette spectrum har været anvendt på telefonlinjer, og er derfor valgt.

Efter signalet er blevet opdelt i forskellige frekvensbånd bliver hver enkelt af frekvens båndene vægtet på forskellige vis.

I det nedenstående afsnit følger en beskrivelse af de forskellige filtre. Man kan på nedenstående billede se den grundlæggende forskel på de to filtre.

2.0.2 Z-transformation

Z-transformationen er en tidsdiskret variant af Laplace Transformation, der transformere et tidsdiskret signal til frekvensdomænet.

Det matematiske udtryk for Z-transformation ses i ligning 2.1.

$$H(z) = \sum_{n=-\infty}^{\infty} h(n)z^{-n} \quad (2.1)$$

2.0.3 IIR-filtre

IIR står for Infinite Impulse Response, hvilket betyder, at det har et uendeligt antal output, da filteret benytter sig af feedback fra tidligere output. Hvis man kigger på ligning (2.2) kan man se den generelle differensligning for et IIR-filter.

$$y(n) = \sum_{k=0}^{M-1} b_k * x(n-k) + \sum_{l=1}^{N-1} a_l * y(n-l) \quad (2.2)$$

$x(n)$: Inputsignal

$y(n)$: Filterets output signal

b_k : Filterets feedforward koefficienter

a_l : Filterets feedback koefficienter

$N-1$: filterets orden

Fordelen ved et IIR-filter i forhold til et FIR-filter, at de kan bruges til at forstærke et signal, de benytter sig også af færre udregninger og dermed mindre hukommelse. Ulemperne ved IIR-filtre er, at de i modsætning til FIR-filtre godt kan være ustabile. Dette kan ses af ligning (2.3), der viser overføringsfunktionen for et IIR-filter i Z-domænet. Ligningen er fremkommet ved brug af Z-transformationen.

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^N b(k) * z^{-k}}{1 - \sum_{k=1}^M a(k) * z^{-k}} \quad (2.3)$$

Som fremgår af ligning (2.3) indeholder overføringsfunktionen både poler og nulpunkter. Det gælder for IIR-filtre, at de er stabile, hvis alle deres poler har en magnitude, der er mindre end 1 (Polerne lægger i enhedscirklen).

2.0.4 FIR-filtre

FIR står for Finite Impulse Response, hvilket betyder, at der er et endeligt antal af impuls-svar. Dette kan man se, i ligning (2.4), der er differensligningen for et FIR filter.

$$y(n) = \sum_{k=0}^{M-1} b_k * x(n-k) \quad (2.4)$$

2.0.5 Hanning vindue

Hanning vinduet er en typisk vinduesfunktion med den matematiske funktions foreskrift:

$$w(n) = 0.5 * (1 - (\cos \frac{2 * \pi * n}{N-1})) \quad (2.5)$$

Vindues funktioner er en funktion (værdier), du ganger på dine frekvensbins. Man benytter vindues funktioner til at mindske mængden af lækage, men essensen af en vindues funktion er sådan set bare, at en funktion ganges på en anden funktion. I praksis inden for Digital Signal behandling anvendes stort set altid et Hanning vindue for at undgå lækage.

3. Høreapparatet

3.1 Høreapparatet

Hørerapparatet er som beskrevet i afsnit 2 bestående af båndpas-filtre. Der opdeler input-signalet i frekvensbånd, der bliver vægtet forskelligt før de bliver sat sammen til et output-signal.

Der er blevet lavet et Matlab program, hvor der først bliver foretaget en Diskret Fourier Transformation på indgangssignalet for at finde ud af, hvilke frekvenser indgangssignalet indeholder. Efterfølgende bliver inputsignalet kørt igennem det fremstillede Høreapparatets filter. Efter det er blevet kørt igennem filteret bliver output-signalet også Diskret Fourier Transformeret, så der kan overskues, hvilke frekvenskarakteristika dette signal indeholder.

3.2 Matlab kode for de forskellige funktioner

3.2.1 FIR-Båndpass funktion

```
function [ output_signal, filter1] = FIR_bandpass_function(input_signal,fc1,fc2 )

len = length(input_signal);

fs = 48000;

N = 1000;

df = fs/N;

m1 = round(fc1/df); %bin nummer der passer bedst
m2 = round(fc2/df); %-||-

filter1 = [zeros(1,m1) ones(1,abs(m1-m2)) zeros(1,N/2-m1-abs(m1-m2))];
filter1 = [filter1 zeros(1,N/2-m1-abs(m1-m2)) ones(1,abs(m1-m2)) zeros(1,m1)];
```

```

filter_time = fftshift(real(ifft(filter1)));

x1 = filter(filter_time,1,input_signal);

hann = hanning(len)';

output_signal = x1.*hann;

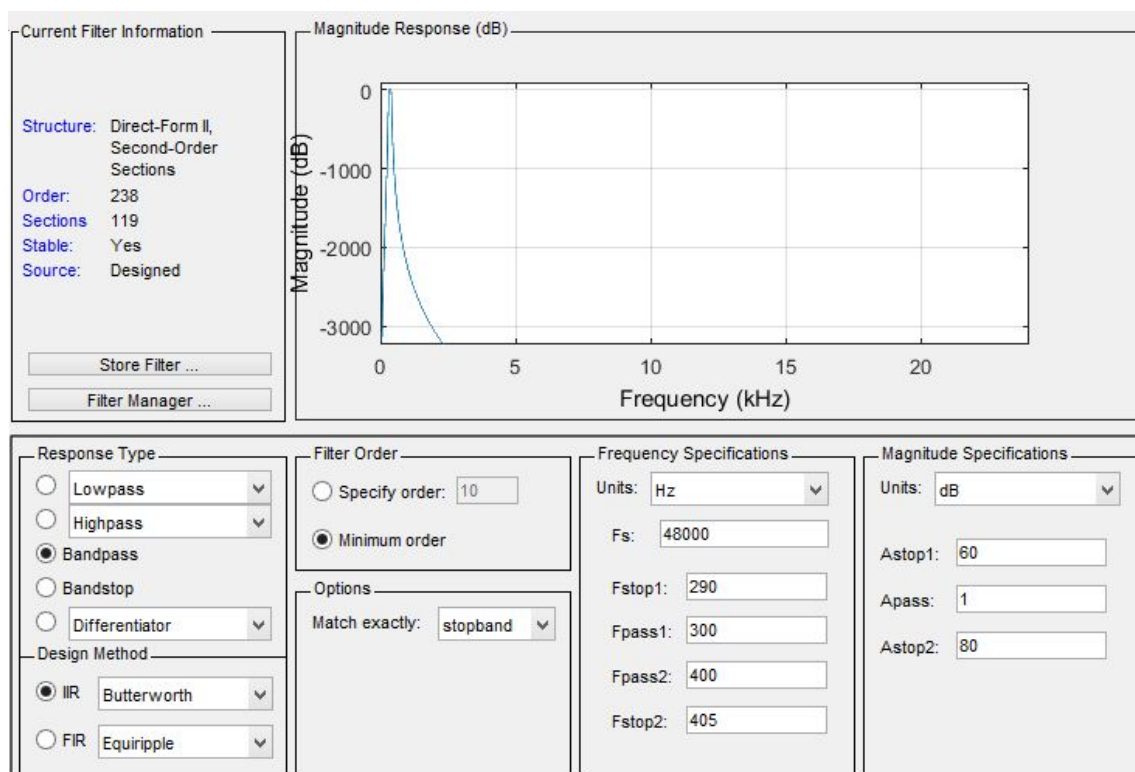
end

```

3.2.2 IIR-bandpass filter

Der blev designet et lavpas-filter i matlab ved brug af Matlabs Filter Design and Analysis Tool. Indstillingerne, der blev sat og det vedhæftede fase og amplitude response kan ses på billede

Filteret blev designt med ønske i at skabe et IIR butterworth filter med to knækfrekvens på 300Hz til 400 Hz og er andenorsdens. Da filteret var designet blev koefficienterne exporteret til et matlab-fil, der blev benyttet i funktionen IIR-lowpass filter.



Figur 3.1: Design af IIR lavpas filter i FDA tools i matlab

Filteret blev desginet med ønske i at skabe et IIR butterworth filter med en knækfrekvens på 10Hz og med en orden på 10. Da filteret var designet blev filteret exporteret til en matlab-funktion gennem matlabs indbyggede code generator der blev benyttet i funktionen IIR-lowpass filter.

Koden for IIR lavpas filter funktionen kan ses nedenuner:

```
function Hd = IIR_lavpas
%IIR_LAVPAS Returns a discrete-time filter object.

% MATLAB Code
% Generated by MATLAB(R) 9.0 and the Signal Processing Toolbox 7.2.
% Generated on: 10-May-2016 12:59:58

% Butterworth Bandpass filter designed using FDESIGN.BANDPASS.

% All frequency values are in Hz.
Fs = 48000; % Sampling Frequency

Fstop1 = 290; % First Stopband Frequency
Fpass1 = 300; % First Passband Frequency
Fpass2 = 400; % Second Passband Frequency
Fstop2 = 405; % Second Stopband Frequency
Astop1 = 60; % First Stopband Attenuation (dB)
Apass = 1; % Passband Ripple (dB)
Astop2 = 80; % Second Stopband Attenuation (dB)
match = 'stopband'; % Band to match exactly

% Construct an FDESIGN object and call its BUTTER method.
h = fdesign.bandpass(Fstop1, Fpass1, Fpass2, Fstop2, Astop1, Apass, ...
Astop2, Fs);
Hd = design(h, 'butter', 'MatchExactly', match);

% [EOF]

ans =

FilterStructure: 'Direct-Form II, Second-Order Sections'
Arithmetic: 'double'
sosMatrix: [119x6 double]
ScaleValues: [120x1 double]
```

```
OptimizeScaleValues: true
PersistentMemory: false
```

3.2.3 Weight funktion

Høreapparatet blev kørt med forskellige signaler, hvor der blev eksperimenteret med forskellige vægtning af de enkelte Båndpas-filtre samt frekvensbåndet, de enkelte filtre dækkede. Efter at havde eksperimenteret med de forskellige frekvensbånd blev det bestemt, at de enkelte frekvensbånd skulle dække følgende frekvensområder, der alle ligger i det hørbare frekvensområde:

- Bånd 1 - 300-400 Hz
- Bånd 2 - 400-600 Hz
- Bånd 3 - 600-1000 Hz
- Bånd 4 - 1000-2000Hz
- Bånd 5 - 2000-3400 Hz

Koden til weight-funktionen:

```
%Give weights in dB
function [ output_signal ] = WEIGHT_func(input_signal,weight,weight2,weight3,weight4,weight5)
% Opdeler signalet i 5 forskellige frekvensbånd, der ligger i båndet for
% menneskeligt tale

x = IIR_lavpas();
b1=x.sosMatrix(:,5);
a1=x.sosMatrix(:,6);
[LP] = filter(b1,a1,input_signal);

[BP1, ~] = FIR_bandpass_function(input_signal, 400, 600);
[BP2, ~] = FIR_bandpass_function(input_signal, 600, 1000);
x1 = hoereapparat; %1000-2000Hz
b2=x1.sosMatrix(:,5);
a2=x1.sosMatrix(:,6);
[BP3] = filter(b2,a2,input_signal);

x2 = IIR_highpas(); %2000-3400Hz
b3=x2.sosMatrix(:,5);
a3=x2.sosMatrix(:,6);
```

```

[HP] = filter(b3,a3,input_signal);

% Vægt de 5 forskellige frekvens bånd, så de tilpasser sig til personens
% individuelle behov:
LP = LP * weight;
BP1 = BP1 * weight2;
BP2 = BP2 * weight3;
BP3 = BP3 * weight4;
HP = HP * weight5;

clc;
figure(1)
subplot(5,1,1)
plot(LP);
subplot(5,1,2)
plot(BP1);
subplot(5,1,3)
plot(BP2);
subplot(5,1,4)
plot(BP3);
subplot(5,1,5)
plot(HP);

output_signal = LP + BP1 + BP2+ BP3+ HP;

output_signal = FIR_bandpass_function(output_signal,300,3400);

%[output_signal] = filter(b4,a4,input_signal);

```

3.2.4 Høreapparat

```

%Høreapparat

[x, Fs]=audioread('Reporters.mp3');
N = 1000000;

%setup frekvensakse
fd = Fs/N;
x_akse = [0:fd:Ff-fd];

x = x(1:N);

```

```

x=rot90(x);

%Foretag FFT på signalet
X = fft(x);

y = WEIGHT_func(x,10,10,10,10,10);

%Foretag FFT på det vægtede signal
Y = fft(y);

%Plot de to signaler, så man kan se forskellen i frekvenserne.
% clc;
% figure(1)
% subplot(2,1,1)
% semilogx(x_akse(1:0.5*end),20*log10(abs((2/N)*X(1:0.5*end))));
% xlabel('Frekvens')
% ylabel('dB')
% subplot(2,1,2)
% semilogx(x_akse(1:0.5*end),20*log10(abs((2/N)*Y(1:0.5*end))));
% xlabel('Frekvens')
% ylabel('dB')

soundsc(x,Fs);
pause(10);
soundsc(y,Fs);

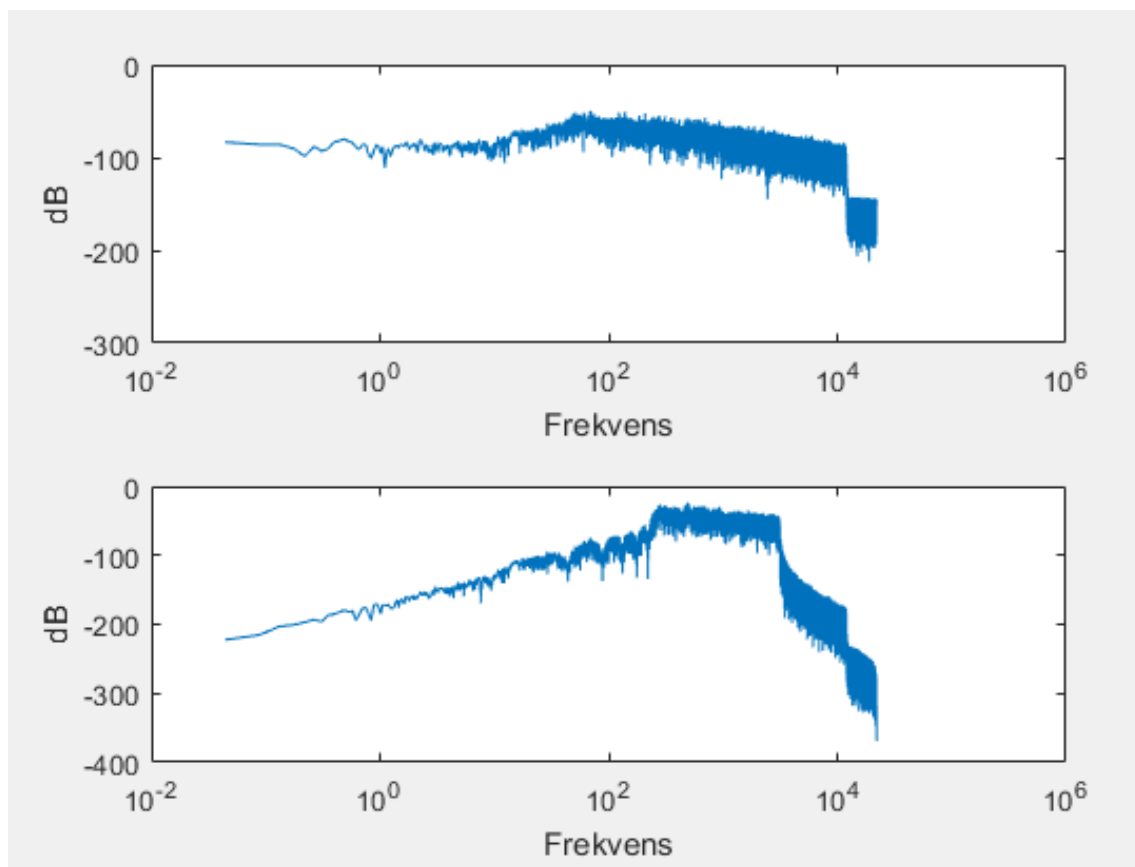
```

I koden ses det at der sker en vægtning i forskellige frekvensbånd, og denne vægtning kan bruges til at møde den individulles behov, men kan også ændre på hvilke frekvensområder de enkelte bånd dækker over. Dette betyder at folk med forskellige hørerproblemer, kan få tilpasset et hørerapparat til netop deres behov.

3.3 Resultat

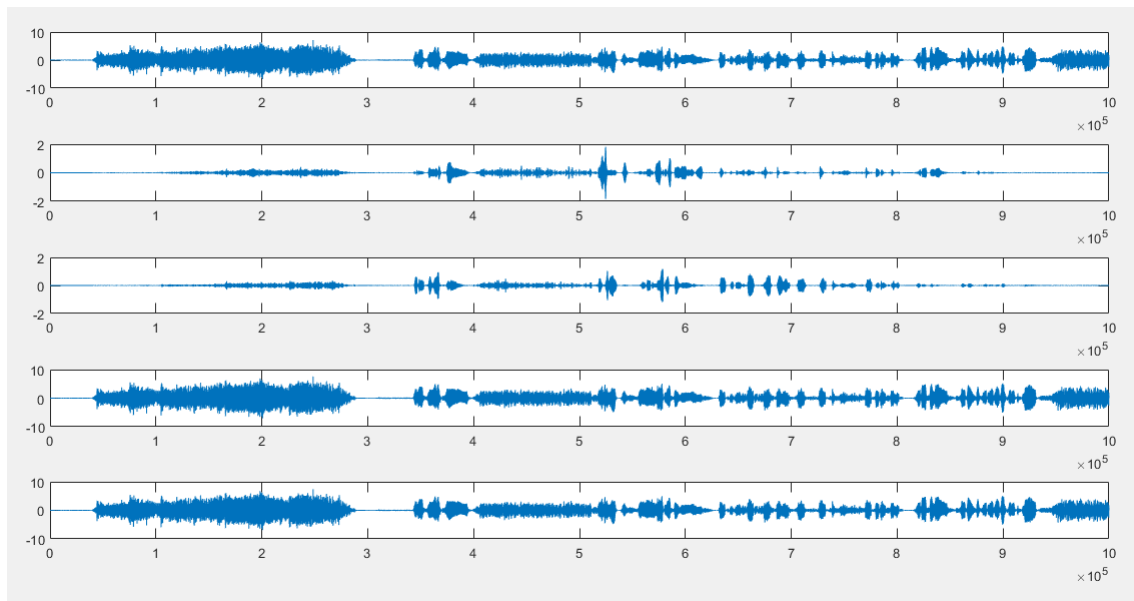
I dette afsnit fremstilles resultaterne af at køre Hørerapparatet på en række forskellige signaler. Signalet, der er blevet valgt at arbejde på er et lydclip af en nyhedsvært i stormvej. I dette lydclip ønskes det at forstærke talen fra nyhedsværten og dæmpe baggrundsstøjen fra stormvejret.

De følgende billeder viser hvordan originalsignalet er blevet manipuleret, så frekvenserne i området 300-3400Hz er blevet forstærket mens de resterende frekvenser er blevet mindsket eller cuttet.



Figur 3.2: frekvenser i hhv. det filtrerede og originale signal

På fig 3.2 ses at der i det filtrerede signal er frekvenserne indenfor bandpassende er forstærket, men frekvenser udenfor er væsentligt dæmpet.



Figur 3.3: Indhold i de forskellige frekvensbånd

På fig 3.3 ses de fem bånd der bliver filtreret i, og det er dermed disse der kunne justeres. Evt. kunne sectionen udvides til flere bånd, men det er valgt ikke at blive gjort.

4. Resultat og Diskussion

Høreapparatet blev kørt med et klip fra en nyhedsreporter i stormvejr. Formålet med dette var at prøve at se om, vi kunne sortere noget baggrundsstøj fra vinden fra signalet, så man bedre kunne høre, hvad nyhedsværten sagde. Der blev derfor arbejdet med forskellige vægtninger af de enkelte frekvensbånd for at finde ud af, hvilke frekvens-områder, der skulle henholdsvis dæmpes og forstærkes for, at man kunne høre nyhedsværten bedst muligt.

5. Konklusion

I denne opgave har vi arbejdet med at fremstille et høreapparat i Matlab ved brug af forskellige filtre. Der er i opgaven blevet anvendt både IIR- og FIR-filtre til at fremstille et høreapparat. Der er set på hvordan man kan tydeliggøre bestemte frekvensbånd ift. andre, og dæmpe støjende frekvenser. I praksis kunne man øge præcisionen mere, og forsøge at dæmpe baggrundsstøjen yderligere, men dette ligger uden for denne opgave.