

AARHUS UNIVERSITET

DSB

SEMESTER 3

Mini-projekt

Studerende:

Mette HAMMER NIELSEN-KUDSK

Martin BANASIK

Finja JETTE RALFS

October 1, 2015



Indholdsfortegnelse

1 Begreber	2
2 Aliasering	2
2.1 Aliasering	2
3 Envelopes	3
3.1 Envelopes	3
4 ADSR	3
4.1 ADSR	3
5 LFO	4
5.1 LFO	4
6 Fourier transformation	6
6.1 Fourier transformation	6
7 Zero Padding	6
8 Lækage	6
9 Vinduer (Hanning Vinduet)	7
10 Udglatning (Smoothing)	7
11 Analyse	7
11.1 Analyse af bilmotor	8
11.2 Analyse af vindmøllestøj	11
11.3 Analyse af EKG	13
11.4 Analyse af Mozart	13
11.5 Analyse af The Weeknd	16
11.6 Analyse af Eminem	19
11.7 Analyse af Arctic Monkeys	22
11.8 Analyse af Rednex	25

1 Begreber

Når vi går fra analoge signaler til digitale, så finder vi repræsentationer af det kontinuerlige signal. Dette kalder vi samples og betegnes med N . Når vi har flere samples på et signal, betegnes intervallet i mellem samples som T_s , samplingstid. Når vi har samplingstid kan vi indføre samplingsfrekvens, det inverse af samplingstid.

$$f_s = 1/T_s$$

Så snart at vi har T_s , ved vi at vi har med et digitalt signal at gøre.

Ved opsætning af sampletidsaksen, definerer vi først vores sampletæller, n :

$$n = [0 : N - 1]$$

Hvor N er antal samples. Efterfølgende bestemmer vi vores sampletidspunkter, t :

$$t = n * T_s$$

Vi kan nu indføre:

$$x(t_s) = X(n * T_s) = X(n)$$

Vi har altid en grundfrekvens og den kalder vi altid f_0 .

2 Aliasering

2.1 Aliasering

Alias = et andet navn for noget/tvetydighed. Vi har tre forskellige slags alias:

- Forkert samling - både for mange samples og for få
- Gentakelser
- Spejling (rundt om Nyquist-frekvensen)

Shannons sandheds sætning

$$f_s \geq 2 * f_{\text{øvre}}$$

I praksis er dette aldrig lig med, men skal altid overholdes. Nyquist-frekvens:

$$f_{\text{nyquist}} = \frac{f_s}{2}$$

Altså defineret som halvdelen af samplingsfrekvensen,

$$f_s$$

3 Envelopes

3.1 Envelopes

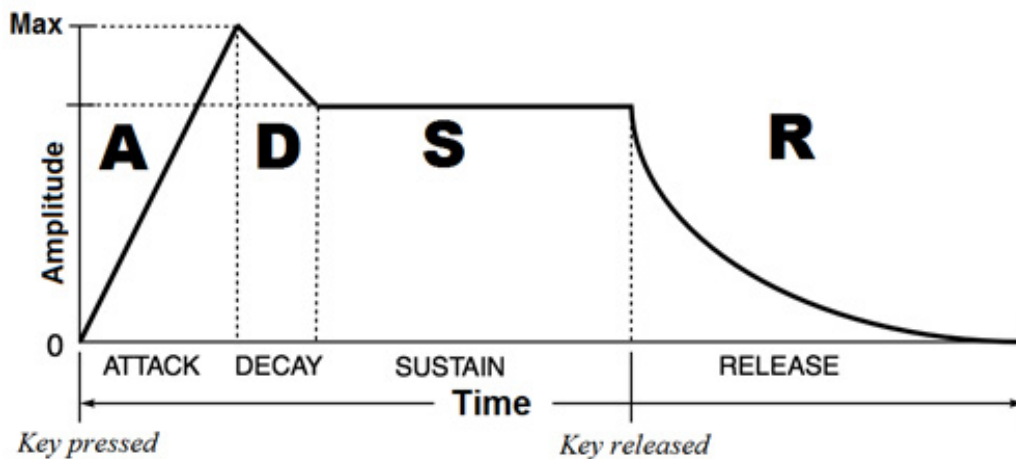
Envelopes = Amplitude billede over et tidsinterval. Under emnet envelopes har vi to punkter:

- ADSR - Attack, Decay, Sustain, Release
- LFO - Low-Frequency Oscillation

4 ADSR

4.1 ADSR

Vi starter med ADSR: Her har vi en figur over den basale ADSR:



Som det ses af figuren har vi fire forskellige stadier:

- Attack - Dette er i starten af signalet, hvor f.eks. en streng på en guitar bliver slået. Som vi kan se så stiger grafen kraftigt i Attack-stadiet.

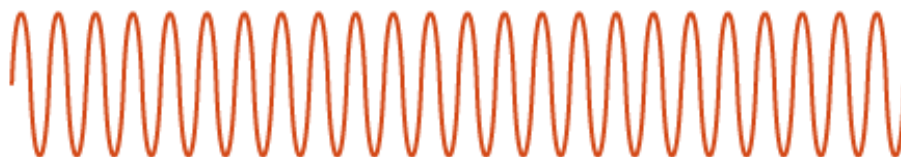
- Decay - Her falder vores graf kraftigt, en smule, da vores streng på guitaren ikke kan holde den kraftige tone i så lang tid. Den skal falde ned til den stationære lyd, hvilket er det næste stadie.
- Sustain - Her holder vores graf et stationært niveau over noget tid. Dvs. at tonen, som vores guitar streng har givet, bliver holdt stabil over længere tid nu - Indtil tonen falder og dør ud (næste stadie).
- Release - Som vi kan se på vores graf dør vores signal ud her. Vores tone har altså holdt så længe den kan og dør nu ud efter det stabile-stadie. Så her falder og til sidst dør tonen ud.

5 LFO

5.1 LFO

Hvis vi så går videre til LFO:

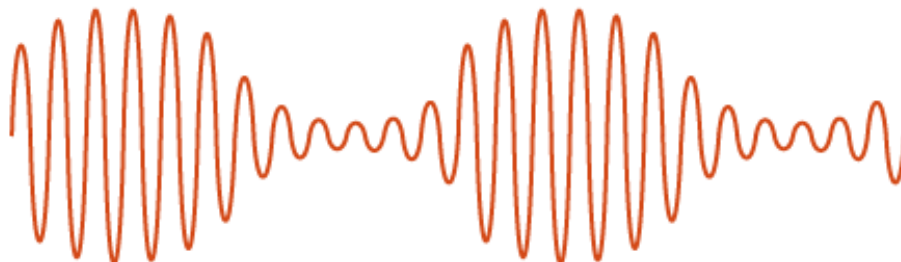
Oscillator's Sound Wave



Low Frequency Oscillator



Resulting Wave with LFO modulating Amp



©2011 ArtificialTunes.tumblr.com

Low-Frequency Oscillation er en anden måde at varierer amplituder på. Dette signal er for det meste under 20 Hz og bruges oftes til lydsignaler. Som navnet af denne metode siger, så bruger man altså kun dette når man har med lavere frekvenser at gøre. Frekvensen man benytter, når man skal have lavet sin sinusbølge, skal altid være lavere end tonen. Når man har fået lavet sin sinus kan vi beregne modeller sinus:

$$S_{mod}(n) = (A_{vo} + 1) * s(n)$$

,hvor $s(n)$ er vores sinus kurve.

Herefter kan vi så beregne modulations graden, som betegnes, f_L .

Det skal så også siges at der er mange forskellige LFO-typer, det er ikke kun sinusser. Der er også firkants-, trekants- og mange andre LFO-typer.

6 Fourier transformation

6.1 Fourier transformation

I DSB har vi nogle forskellige analyse værktøjer. Et af disse er Fourier transformation (DFT). Når vi benytter DFT regner vi med komplekse tal. Definitionen af DFT er betegnet:

$$X(m) = \sum_{n=0}^{N-1} x(n) * e^{-j * \frac{2 * \pi}{N} * m * n}$$

,hvor m = frekvens nummerering.

Det er altså sådan vi går fra tidsdomæne $x(n)$, til frekvensdomæne $X(m)$ - via Fourier transformation, DFT.

Hvis vi så gerne vil tilbage igen, altså fra frekvensdomæne $X(m)$, til tidsdomæne $x(n)$. Så laver vi Invers Fourier Transformation, IDFT. Definitionen af IDFT er betegnet:

$$x(n) = \frac{1}{N} \sum_{m=0}^{N-1} X(m) * e^{j * \frac{2 * \pi}{N} * m * n}$$

Længden er = 1. Herefter er vi altså tilbage i tidsdomænet $x(n)$.

7 Zero Padding

Ved Zero Padding ligger vi x antal nuller ind i enden af vore signal. På den måde bliver længden af vores signal længere. Vi vinder altså ikke mere information omkring signalet, vi får blot flere "nuller-samples" hæftet på i enden. Dette gør at vi lettere kan fange frekvenskomponenternes opførsel (faktisk dobbelt så god frekvensopløsning) og at længderne på de forskellige signaler bliver ens. I dette miniprojekt synes vi ikke det gav mening, at lave Zero Padding på nogle af vores signaler og derfor er dette ikke blevet lavet.

8 Lækage

Hvis man ikke rammer den rigtige frekvens ville alle ens samples fordele sig ud over hele x-aksen. Dette duer ikke, det bliver udoverskueligt og ulæseligt. Dette kaldes lækage. Lækage er noget vi kan få nedsat ved hjælp af vinduer. (se næste afsnit).

9 Vinduer (Hanning Vinduet)

Indenfor digital signal behandling har vi mange forskellige vinduer, der ned-sætter lækage, ved hjælp af forskellige egenskaber. Hanning Vinduet er det vindue som vi benytter os mest af. Hanning Vinduet er energimæssigt mindre, hvilket ikke gør så meget, da vi let kan skrue op for strømmen. Vi mister frekvensopløsning - bliver faktisk næsten halveret, når vi benytter Hanning vinduet. Så vi betaler altså en pris, for at benytte os af Hanning vinduet, for at få vores signal til at se pænt og læseligt ud.

10 Udglatning (Smoothing)

Udglatning fungerer lidt ligesom et lavpasfilter - det fjerner nemlig højfrekvenserne. Ved dette får vi altså pænere grafer, som er lettere læselige. Det er altså en udglatning af vores frekvensspektre, der er tale om.

11 Analyse

Når vi laver sådan en analyse her så opstille vi to grafer: En for længden og en for fasen (Bodeplot).

- X-aksen = Frekvens i Hz (opdelt i decader)
- Y-aksen på længde grafen = $\text{dB} = 20 * \log(10) | X(m) |$
- Y-aksen på fasevinkel grafen = $\angle(X(m))^\circ$

Når vi har fået tegnet vores graf, så har vi det, der hedder Frekvensopløsningen, som er afstanden hver sample:

$$\Delta f = \frac{fs}{N}$$

Det næste vi kan tilføje er Analysefrekvensen:

$$f_{analysis}(m) = m * \Delta f$$

Til sidst har vi Parsevals sætning:

$$\sum_{n=0}^{N-1} |x(n)|^2 = \frac{1}{N} \sum_{m=0}^{N-1} |X(m)|^2$$

Summen af kvadrerede samples i tidsdomænet er lige med summen af kvadrerede samples i frekvensdomænet.

11.1 Analyse af bilmotor

Efter at have fået styr på alle begreberne, går vi nu i gang med bilmotoren. Koden fra MatLab og graferne, der bliver tegnet klipper vi ind, her i dokumentet. Vi starter med at indsætte bilmotor lyden og derefter angiver vi alle de enheder, der skal angives, hvorefter vi opretter grafer, laver DFT og får vores signaler ind. Se nedenunder.

```
Vi indlæser bilmotor lyd (.wav:
[x, fsample] = wavread('ARv6');
N = length(x); —(Antal samples)
Tlength = N/fsample; —(Varighed i sek.)
X = fft(x,N); —( Vi laver DFT)
deltaf = fsample/N; —(Vi finder Δf)
faxis = [0 : deltaf : fsample - deltaf]; — (Vi opretter en akse med interval)

figure(1); clf — (Vores Længde graf)
semilogx(faxis(1 : 0.5 * end), 20 * log10(abs((2/N) * X(1 : 0.5 * end))))
xlabel('Frekvens i Hz')
ylabel('Størrelse dB rel. 1 Volt')
title('DFT størrelse (magnitude)')
grid on — (Gøre vores akser pæne)
axis([10 1000 -95 -25])
```

```
figure(2); clf — (Vores Fase vinkel graf)
semilogx(faxis(1 : 0.5 * end), ((angle((2/N) * X(1 : 0.5 * end))))))
xlabel('Frekvens i Hz')
ylabel('Fase')
title('DFT fase')
grid on
```

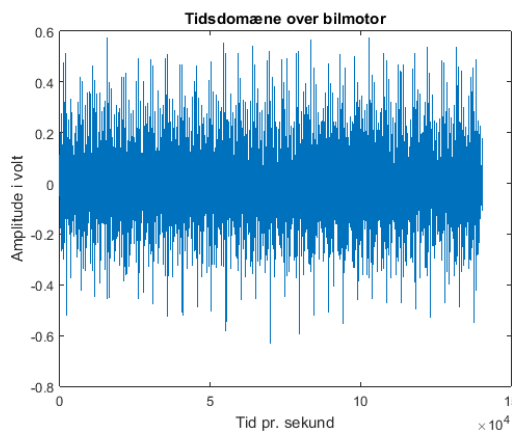


Figure 1:

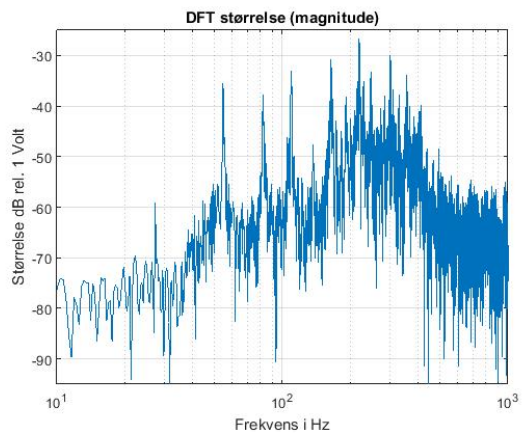


Figure 2:

Når vi kører vores kode, udskrives disse grafer:
hejehej

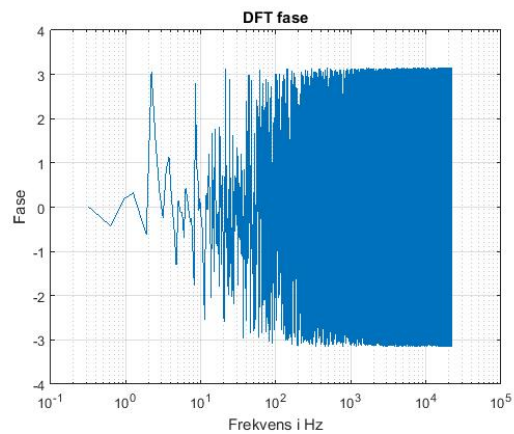
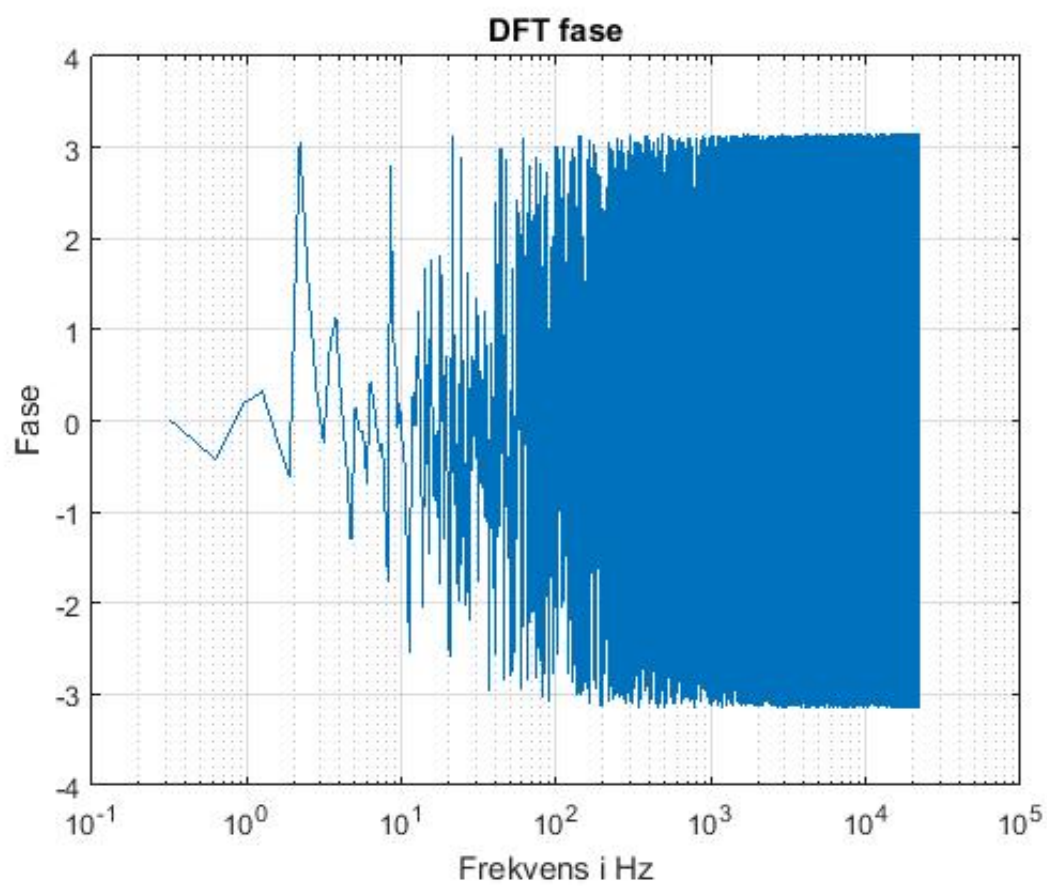
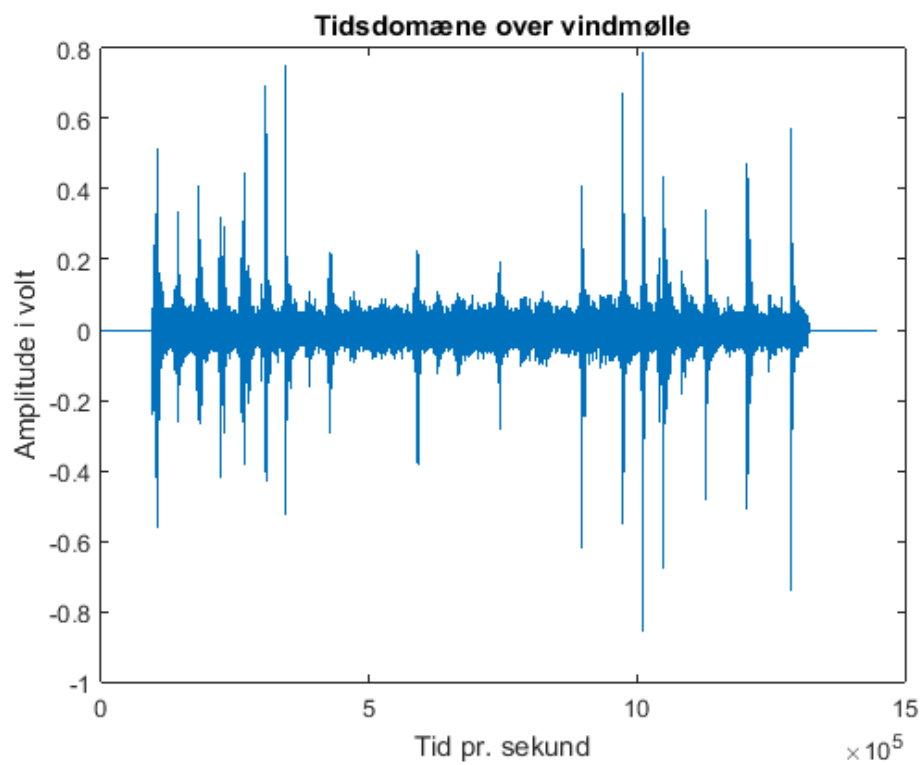
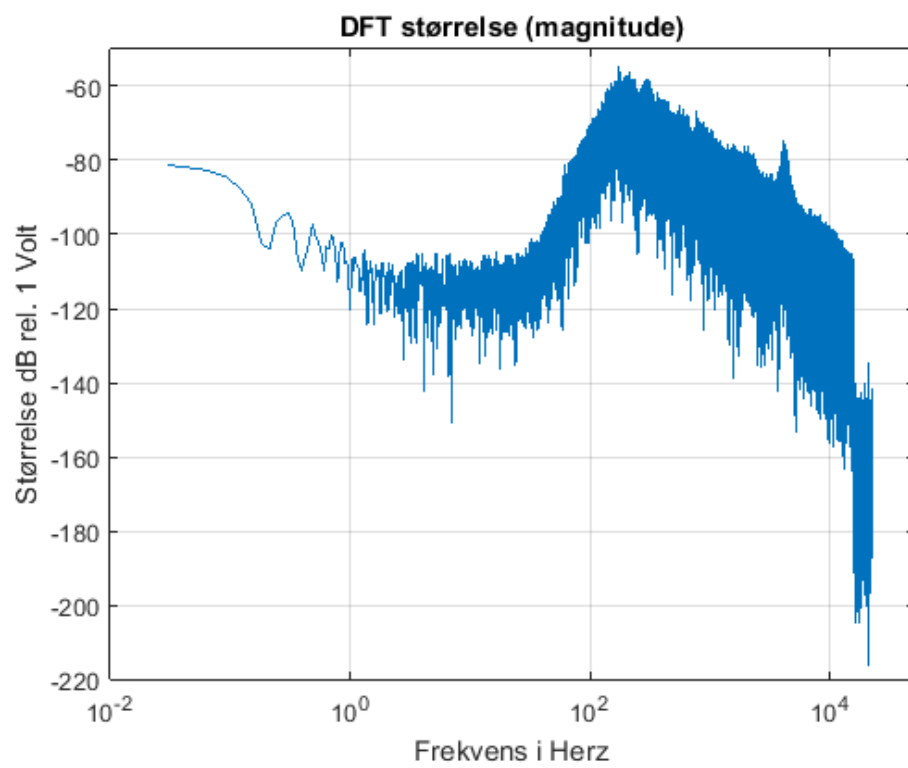


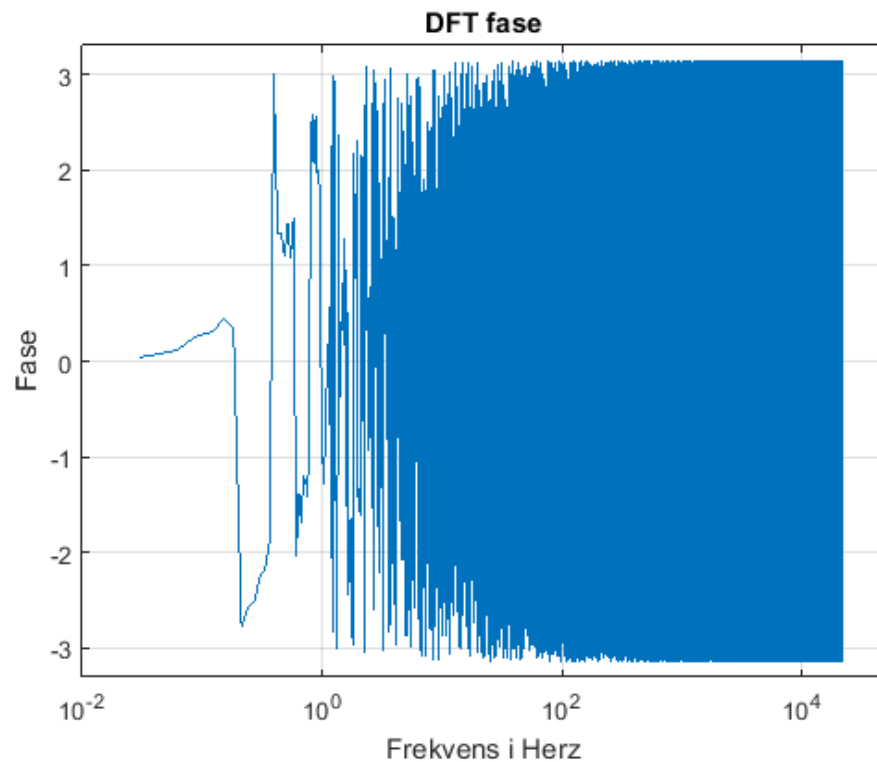
Figure 3:



11.2 Analyse af vindmøllestøj





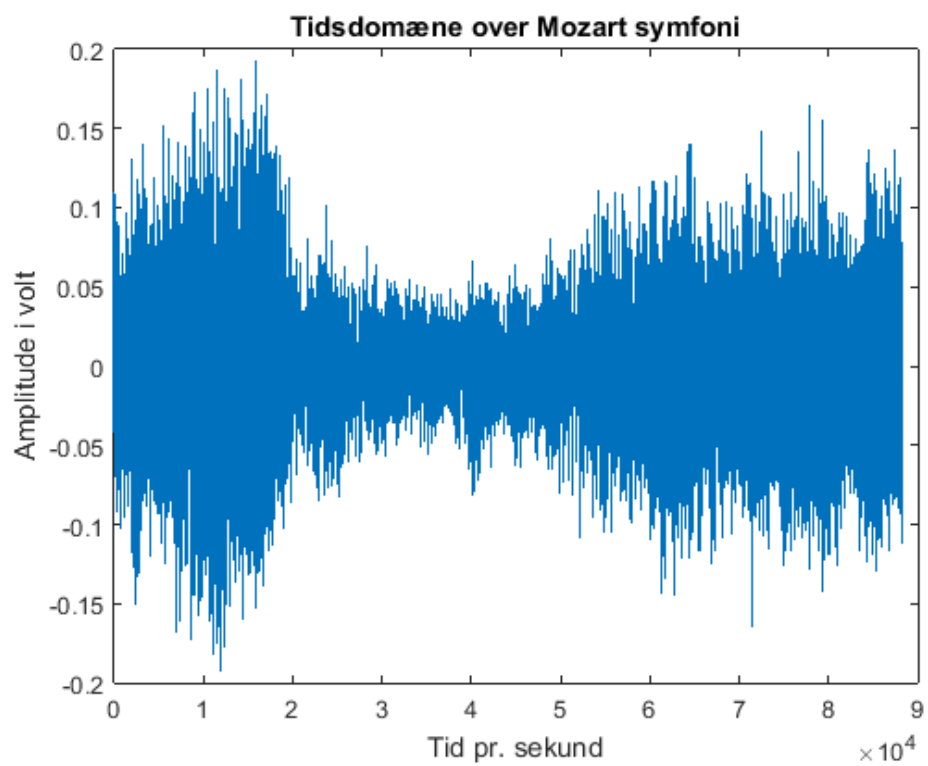


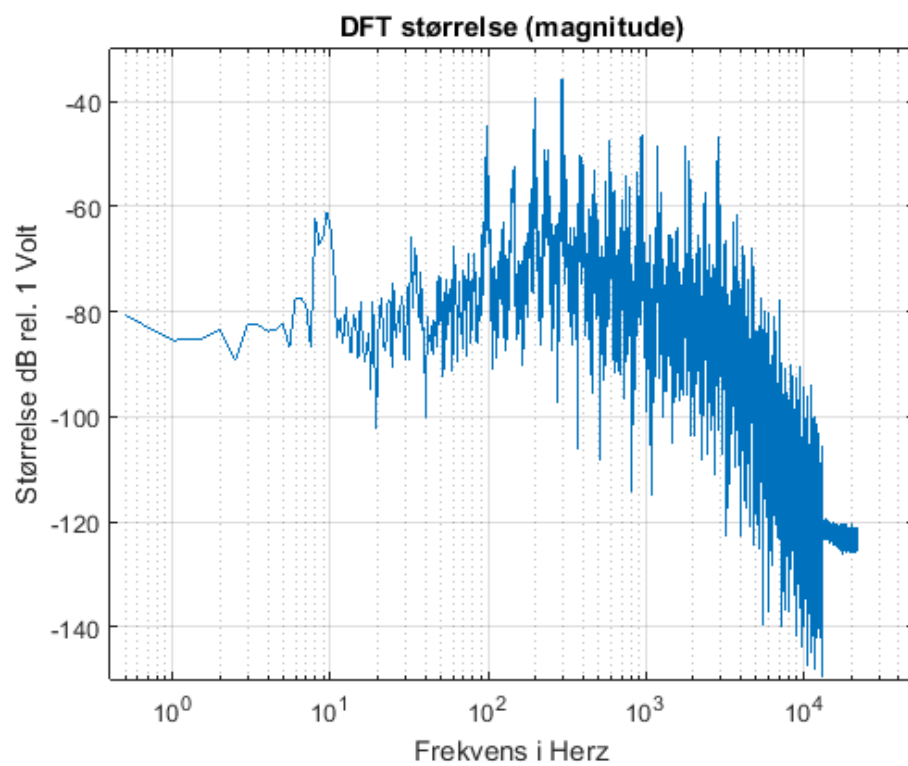
11.3 Analyse af EKG

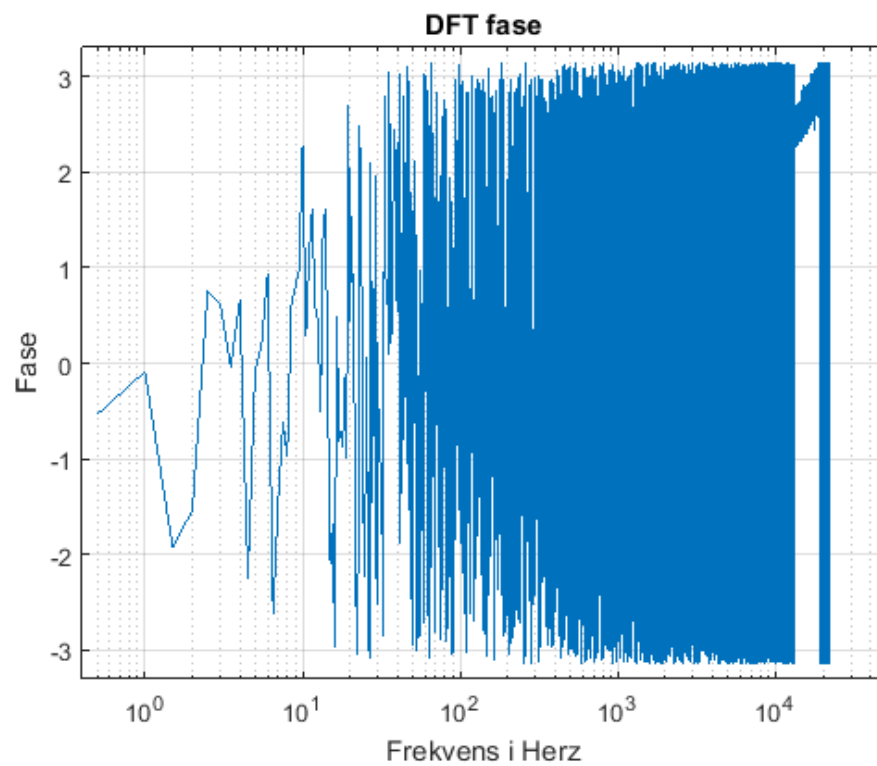
Da vi her ikke har med et lydsignal at gøre, skal vi ikke længe benytte funktionen "semilogx", men blot funktionen "plot".

11.4 Analyse af Mozart

Mozart - 39's Symphony no. 40 - 1st movement.

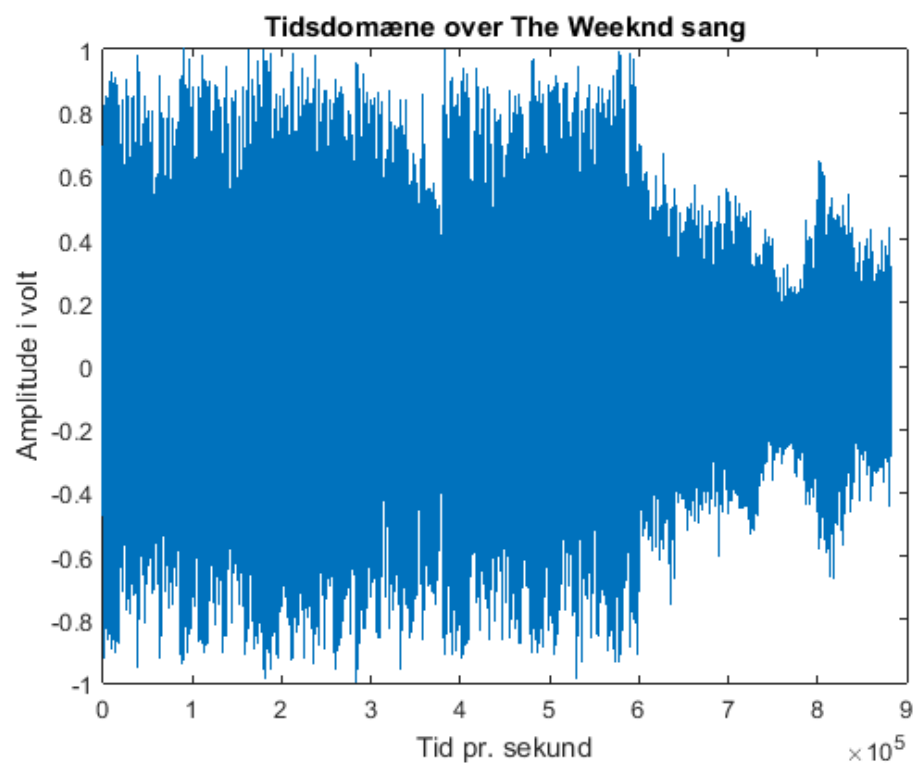


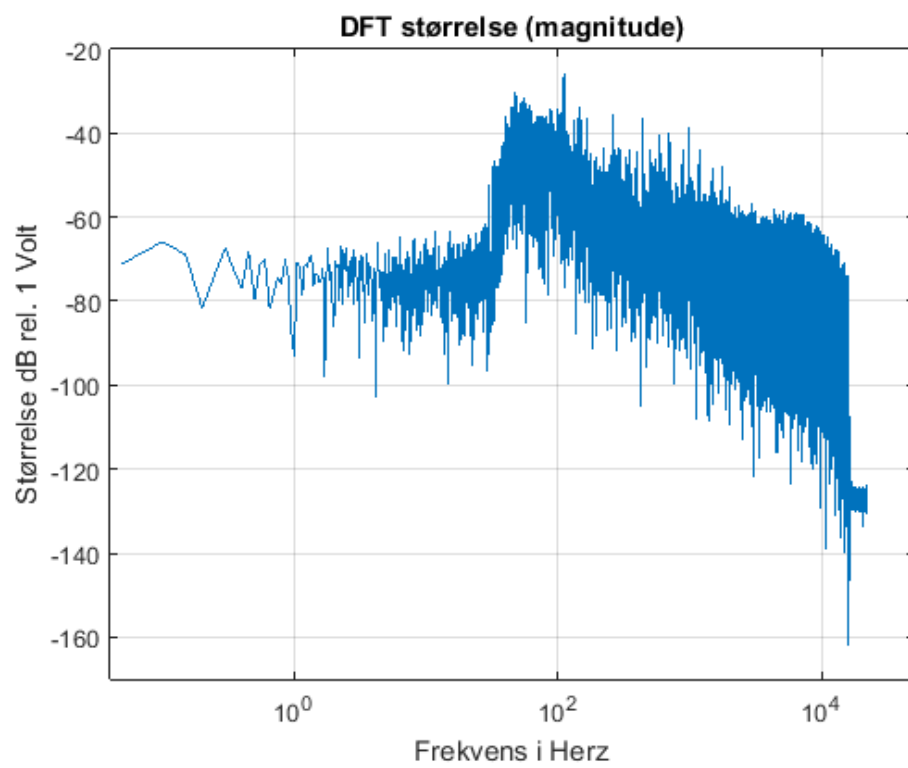


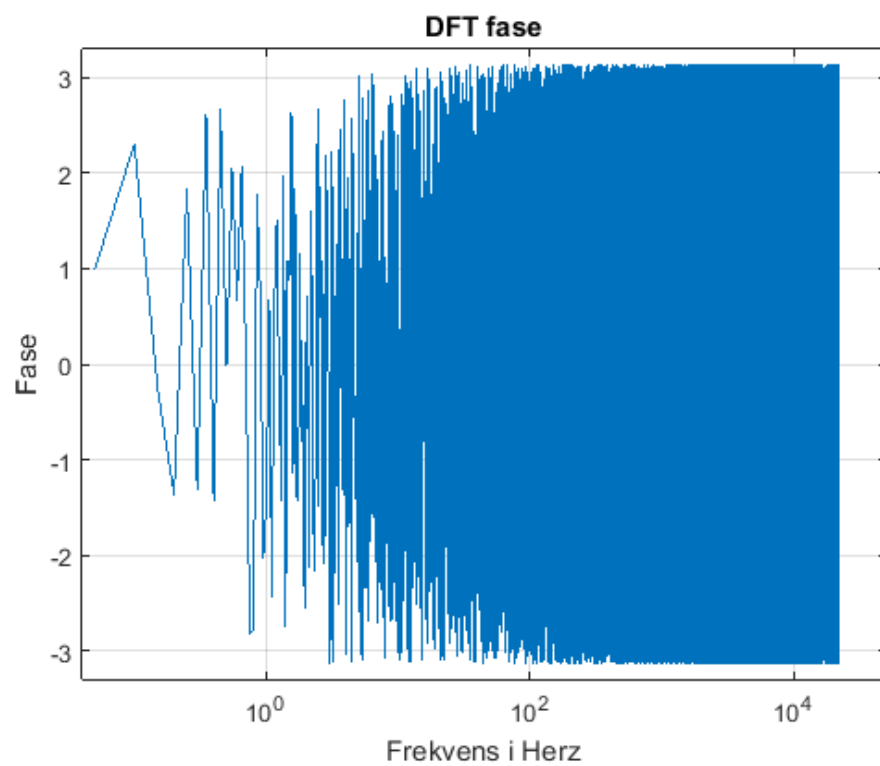


11.5 Analyse af The Weeknd

The Weeknd - Can't feel my face

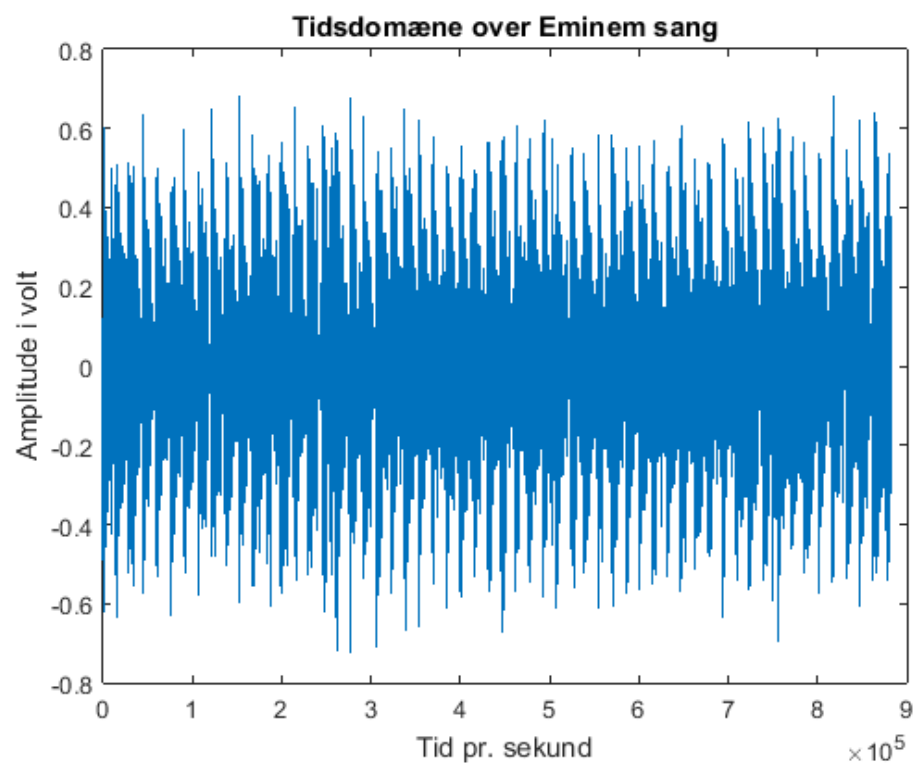


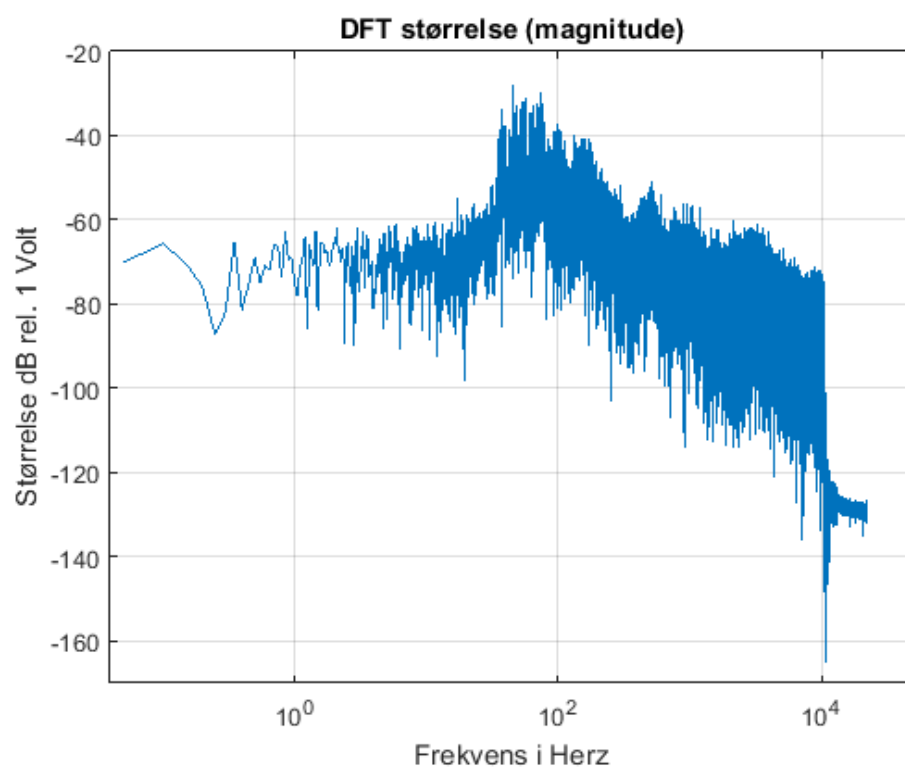


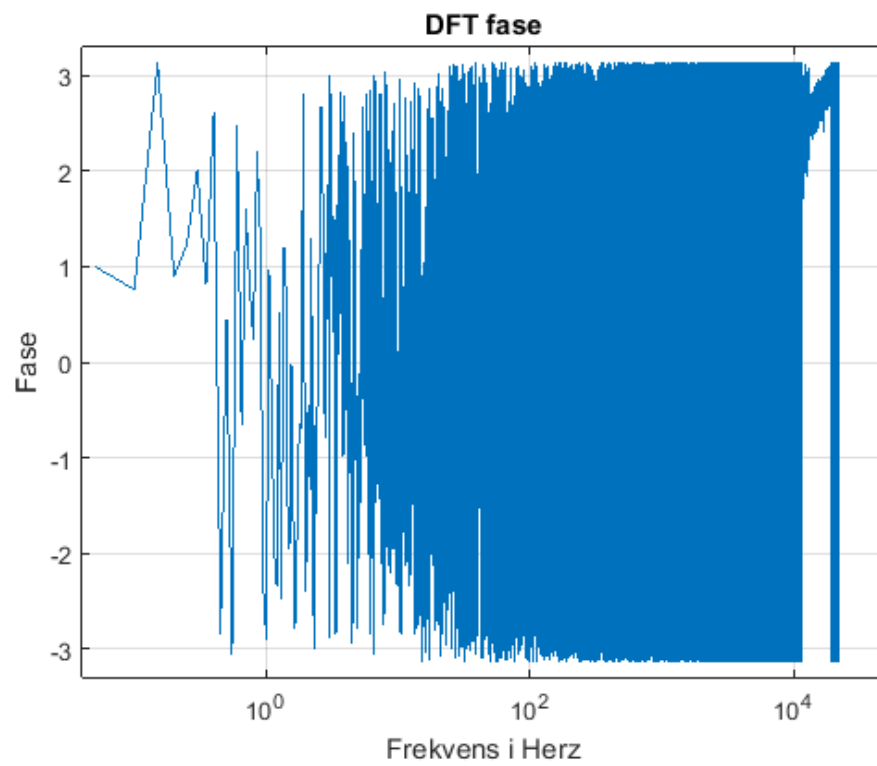


11.6 Analyse af Eminem

Eminem - Lose yourself

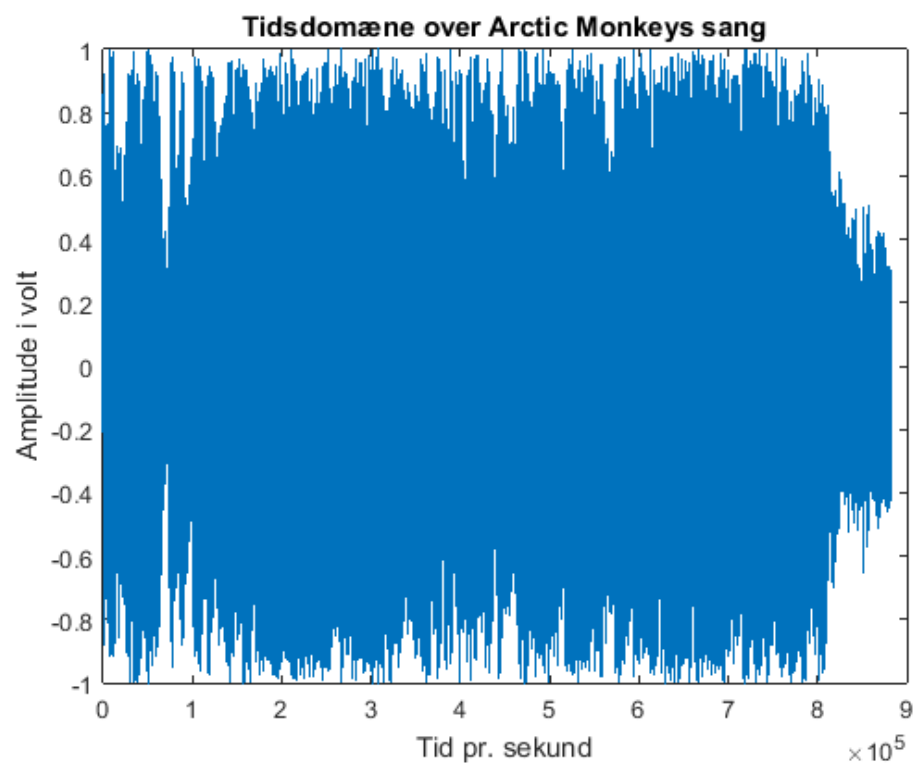


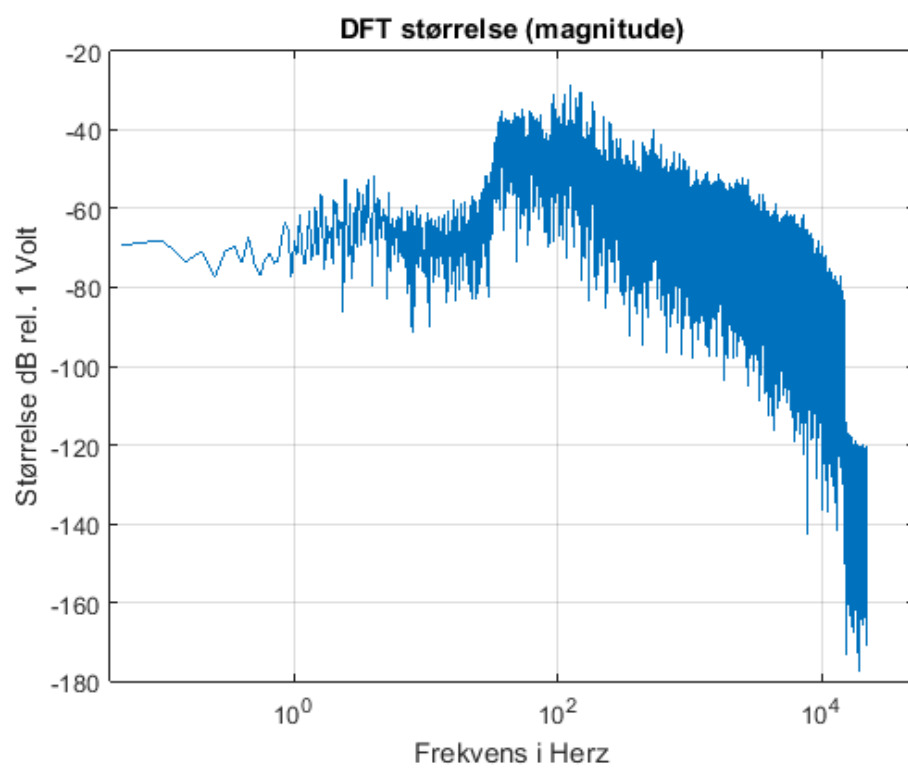


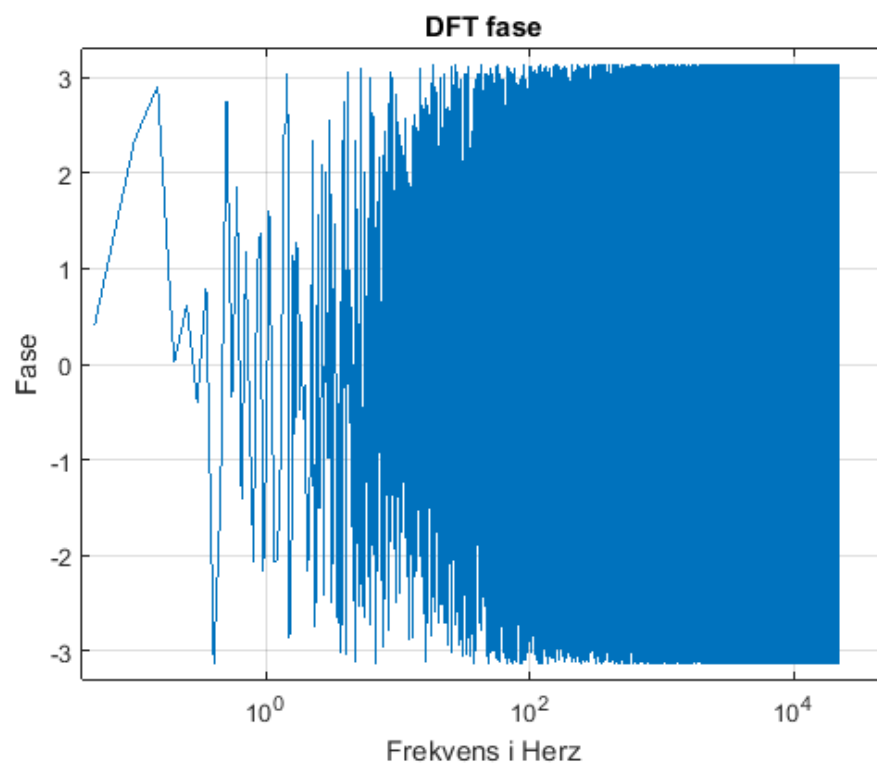


11.7 Analyse af Arctic Monkeys

Arctic Monkeys - R U Mine







11.8 Analyse af Rednex

Rednex - Cotton Eye Joe

