

# E3DSB miniprojekt 1 - Tidsdomæneanalyse

Janus Bo Andersen <sup>1</sup>

13. september 2019

<sup>1</sup>ja67494@post.au.dk

# Indhold

<b>1</b>	<b>Indledning</b>	<b>1</b>
<b>2</b>	<b>Analyser</b>	<b>1</b>
2.1	Afspilning af lydclip . . . . .	1
2.2	Bestemmelse af antal samples . . . . .	1
2.3	Plot af signal . . . . .	1
2.4	Min, max, RMS og energi . . . . .	1
2.5	Venstre vs. højre kanal (for $s_1$ ) . . . . .	3
2.6	Nedsampling af signal (for $s_1$ ) . . . . .	3
2.7	Fade-out med envelopes (for $s_2$ ) . . . . .	3
<b>3</b>	<b>Konklusion</b>	<b>3</b>

# 1. Indledning

Dette første miniprojekt i E3DSB behandler tre lydsignaler med analyser i tidsdomænet. Opgaven er løst individuelt. Dette dokument er genereret i Matlab med en XSL-template. Matlab-kode og template findes på <https://github.com/janusboandersen/E3DSB>. Følgende lydclip benyttes

Signal	Skæring	Genre	Samplingsfrekv.
$s_1$	Spit Out the Bone	Thrash-metal	44.1 kHz
$s_2$	The Wayfaring Stranger	Bluegrass	96 kHz
$s_3$	Svanesøen	Klassisk	44.1 kHz

Tabel 1.1: 3 signaler behandlet i analysen

## 2. Analyser

Før analyser ryddes der op i `Workspace`.

```
1 clc; clear('all'); close('all');
```

### 2.1 Afspilning af lydclip

Filen med signaler åbnes med `load`. Signaler kan afspilles med `soundsc(signal, fs)`. Samplingsfrekvensen  $f_s$  sættes efter værdi i tabel 1.1. Samplingfrekvenser for de tre signaler er inkluderet i `.mat`-filen.

```
1 load('miniprojekt1_lydclip.mat');  
2 soundsc(s1, fs_s1); %playback  
3 clear('sound'); %stop playback
```

### 2.2 Bestemmelse af antal samples

### 2.3 Plot af signal

### 2.4 Min, max, RMS og energi

Signalerne er i stereo (2 kanaler / kolonner). Hvis vi har et system med to højttalere, giver det mening at betragte kanalerne separat. Altså vi ser kanalerne i forlængelse, som en mono serie med  $M = 2N$  samples. Denne løsning bruges, fordi det er sådan et menneske med to ører og sæt hovedtelefoner ville opleve signalet :-)

En sum eller et gennemsnit på tværs af kanalerne ville betyde, at kanaler ude af fase kunne eliminere hinanden. Dette ville give mening som en simpel konvertering til mono, dvs. vi kunne beregne mål på hvad der ville ske i et simpelt mono-system.

**Beregning:** Minimum og maksimum findes nemt med hhv. `min` og `max`. I tidsdomænet er effekten af et signal proportionalt til kvadratet på amplituden. For en sekvens  $x(n) \in \mathbb{R}, n = 0, \dots, N-1$  defineres  $x_{pwr}(n) = |x(n)|^2 = x(n)^2$ . I diskret tid er energien i signalet summen af “effekterne”, dvs.  $E_x = \sum_{n=0}^{N-1} |x(n)|^2$ . RMS-værdien kan så beregnes som kvadratroden af midleffekten, dvs.  $x_{RMS} = \sqrt{\frac{1}{N} E_x}$ .

```

1  signaler = {'s1'; 's2'; 's3'};
2  N = [length(s1); length(s2); length(s3)]; % antal diskrete tidsobservat.
3  M = 2*N; % beregn samlet antal af datapunkter ~ målinger
4
5  s1_vec = reshape(s1,1,[]); %reshape matricer til søjlevektorer
6  s2_vec = reshape(s2,1,[]);
7  s3_vec = reshape(s3,1,[]);
8
9  minima = [min(s1_vec); min(s2_vec); min(s3_vec)];
10 maxima = [max(s1_vec); max(s2_vec); max(s3_vec)];
11 energi = [sum(s1_vec.^2); sum(s2_vec.^2); sum(s3_vec.^2)]; % kvadratsum
12 rms = [energi(1)/M(1); energi(2)/M(2); energi(3)/M(3)].^(1/2);
13
14 T = table(signaler, N, M, minima, maxima, energi, rms)

```

```

T =
3x7 table
    signaler      N      M      minima      maxima      energi      rms
    -----
    's1'      1.323e+06      2.646e+06      -1.0166      1.0191      2.5336e+05      0.30944
    's2'      3.36e+06      6.72e+06      -0.61796      0.62791      34641      0.071797
    's3'      1.4112e+06      2.8224e+06      -0.85016      0.76907      5662.2      0.04479

```

Signal	Min (1;2)	Max (1;2)	Energi	RMS
$s_1$	0; 0	0; 0	0	0
$s_2$	0; 0	0; 0	0	0
$s_3$	0; 0	0; 0	0	0

Tabel 2.1: Statistik på signalerne

**2.5** Venstre vs. højre kanal (for  $s_1$ )

**2.6** Nedsampling af signal (for  $s_1$ )

**2.7** Fade-out med envelopes (for  $s_2$ )

## **3. Konklusion**