

AARHUS UNIVERSITET

INTRODUKTION TIL DIGITAL SIGNALBEHANDLING

3. SEMESTER

DSA Case 2

Gruppemedlemmer:

Gustav A. Gammelgaard

Stinus Lykke Skovgaard

Tim Hede Stenholt

AU id:

au538293

au520659

au543518



31. marts 2017

Indhold

1	Problem beskrivelse	3
2	Opgave 1	3
3	Opgave 2	3
3.1	Lineært midlingsfilter	3
3.2	Eksponentielt midlingsfilter	5
4	Opgave 3	5
5	Matlab kode	5

Figurer

1	Histogram af load med og uden filter. Filterorden = 10	3
2	Histogram af noload med og uden filter. Filterorden = 10	4
3	Histogram af load med og uden filter. Filterorden = 50	4

1 Problem beskrivelse

Denne case omhandler måden at forbedre et støjfyldt signal med et midlingsfilter. Dette filter skal midle et signal fra en vægt. Typisk vil signalet fra vægten svinge voldsomt som vil gøre det svært at måle korrekt. Midlingsfilteret vil gøre det muligt at få en pålidelig måling. Filteret er realiseret ved hjælp af matlab og vægt data er importeret ind i matlab og testet i dette miljø.

2 Opgave 1

3 Opgave 2

Det kommende afsnit kommer til at omhandle hvordan midlingsfilteret virker og hvordan det er implementeret. Filteret er blevet testet på dataen fra opgave 1. Vi har eksperimenteret med både lineært og eksponentielt midlingsfilter.

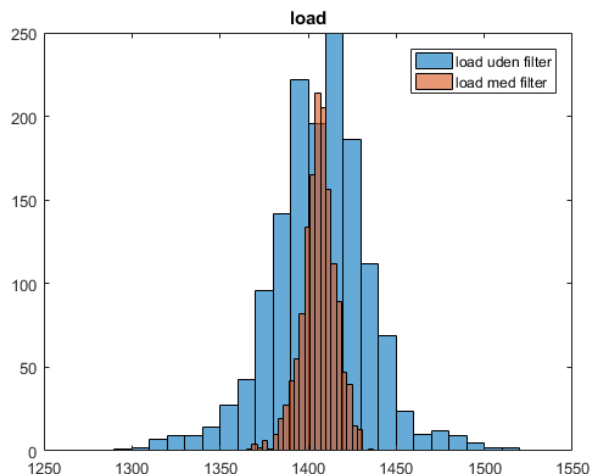
3.1 Lineært midlingsfilter

For at midle data fra opgave 1, lavede vi først et lineært midlingsfilter. Dette er gjort på følgende måde:

```
1 M = 10;  
2 h1 = 1/M * ones(1, M);
```

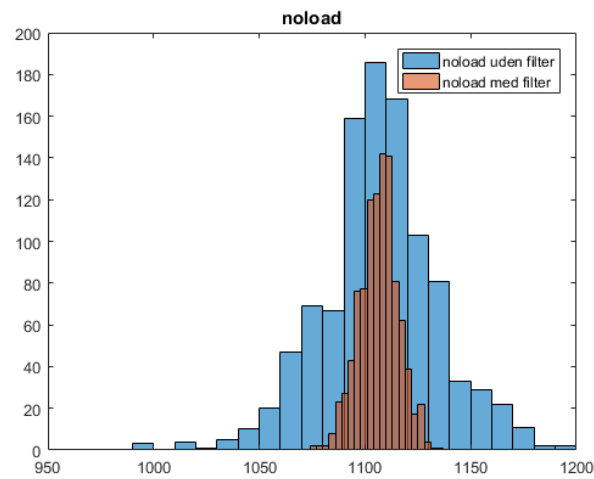
Dette laver et midlingsfilter med M koeficienter, der hver har en værdi på $1/M$. Ved at indstille på ordnen (altså M) vil man kunne få filteret til midle kraftigere.

Da dataen fra opgave 1 viser at der både er en load og noload, er filteret påført de to stadier hver for sig. Dette er blevet plottet på et histogram, som kan ses på figur 1



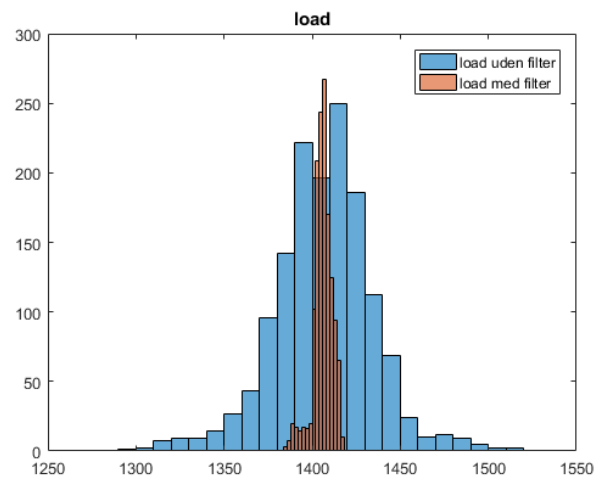
Figur 1: Histogram af load med og uden filter. Filterorden = 10

Man kan se at filteret får vores data til at ligge tættere på hinanden, hvilket er det vi gerne vil se. Det samme kan ses for noload på [2](#)



Figur 2: Histogram af noload med og uden filter. Filterorden = 10

Ved at ændre på ordnen kan man få filtret til at midle kraftigere. Dette kan ses på figur [3](#)



Figur 3: Histogram af load med og uden filter. Filterorden = 50

Det ses tydeligt at de orange pinde er blevet smallere og fylder et mindre område på grafen. Hvis man kigger på varians og spredning kan man se at de stemmer godt overens med vores grafer.

```

3
4 var_load = var(y_load)
5 var_noload = var(y_noload)
6
7 P_load = (1/sqrt(M)*S_load)^2
8 P_noload = (1/sqrt(M)*S_noload)^2

```

Ved $M = 10$ får vi en varians på ca 90, mod en varians på ca 25 ved en orden på 50. Hvis man kigger på støj-effekten ser man det samme. Ved en orden på 10 er støj-effekten ca 10 højere end ved en orden på 50.

Hvis man ønsker en max svingningstid for sit system, bliver man muligvis nødt til at begrænse sin orden på filteret. Hvis vi har et krav om en indsvingningstid på 100ms kan vi lave en hurtigt udregning:

```

9 maxKoeff = fs*0.1;

```

Dette giver os et filter med en orden på 30.

3.2 Eksponentielt midlingsfilter

4 Opgave 3

5 Matlab kode