

Litt om matlab

MUS2006 - Musikk og bevegelse

20 mars 2014

Installasjon

Matlab er tilgjengelig for alle studenter og ansatte på UiO:

<http://www.uio.no/tjenester/it/maskin/programvare/p-base/matlab.html>

Mocap Toolbox, fra universitetet i Jyväskylä

<https://www.jyu.fi/hum/laitokset/musiikki/en/research/coe/materials/mocaptoolbox>

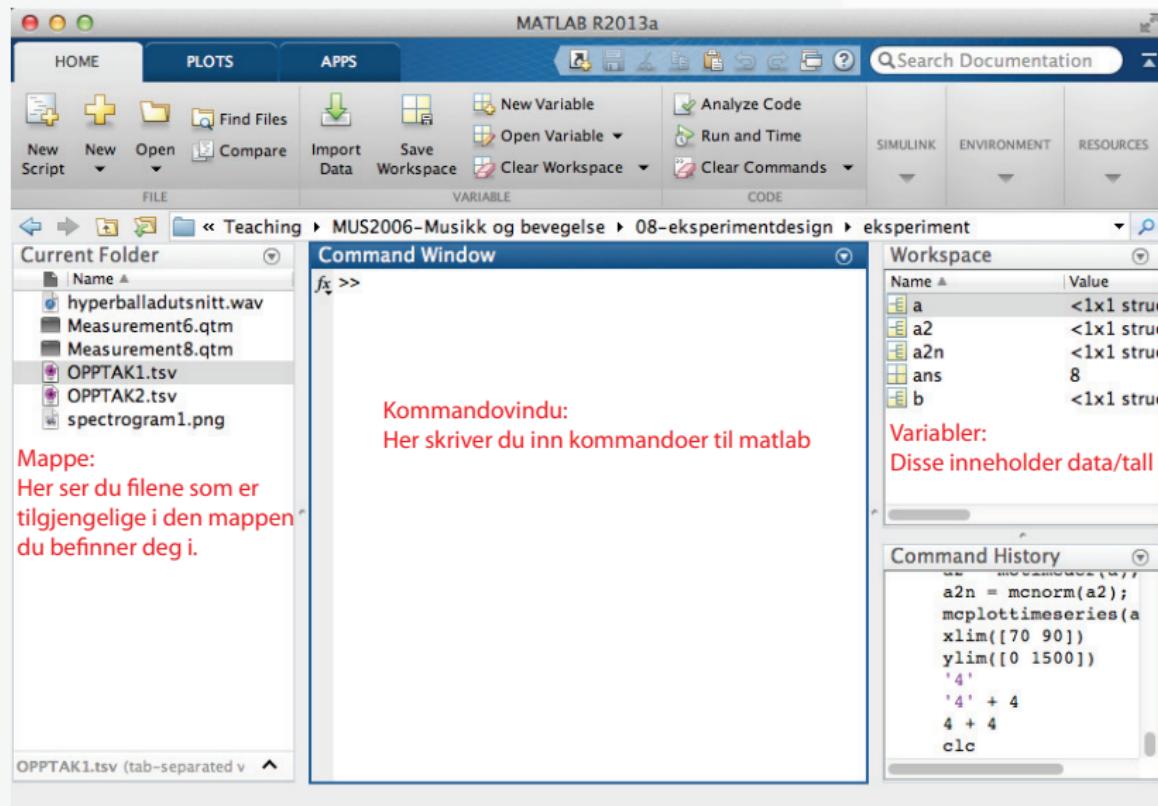
- ▶ analyse av motion capture data

MIR Toolbox, fra universitetet i Jyväskylä

<https://www.jyu.fi/hum/laitokset/musiikki/en/research/coe/materials/mocaptoolbox>

- ▶ avansert lydanalyse

Matlabvinduet

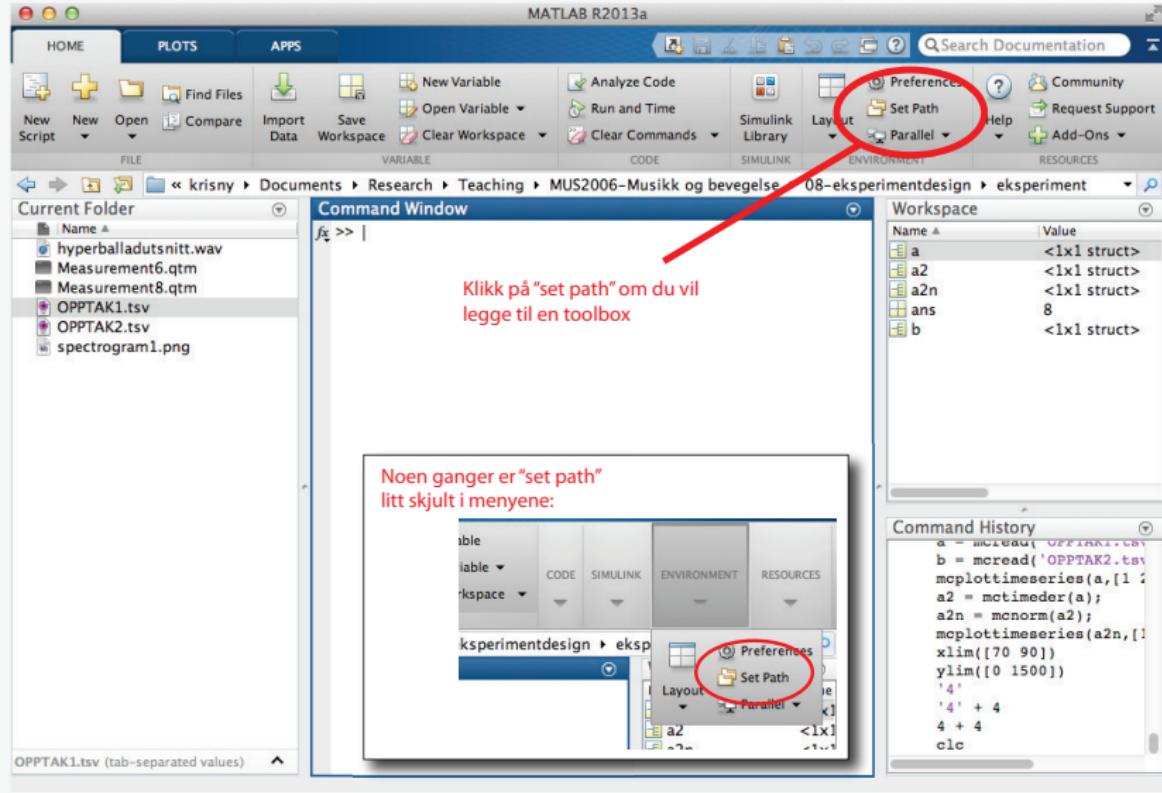


Basic matlab — uten toolboxer

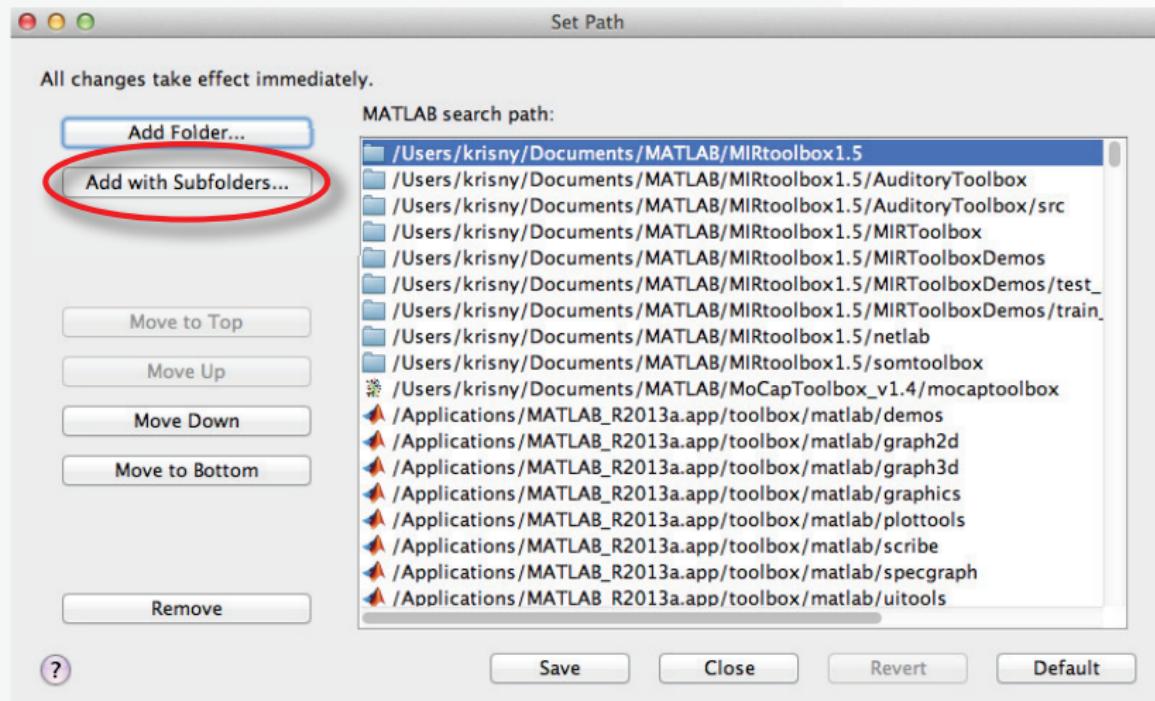
Sette variabelen a til verdien 4	a = 4 eller a = 4; semikolon er valgfritt
Sette variablen b til verdien av a ganger 2	b = a * 2; semikolon gjør at svaret ikke vises på skjermen
Sette variabelen c til tallrekken 4, 5, 6, 3, 5	c = [4 5 6 3 5] tallene kan også skilles med komma
Lage en graf av c	plot(c)
Justere aksene på grafen	xlim([0 8]) og ylim([-1 10]) justerer henholdsvis x-aksen og y-aksen
Finne det fjerde tallet i c	c(4) returnerer verdien 3
Sette den tredje verdien i c til verdien 2	c(3) = 2;
Slette alle variablene	clear NB! ingen angrefunksjon
Lage en ny (tom) figur	figure

Mange av disse funksjonene kan du også gjøre uten å skrive kommandoer. F.eks. høyreklikk på en variabel og velg "plot", eller bruk zoomefunksjonen i plottevinduet.

Legge til toolbox:



Legge til toolbox:



Klikk på "add with subfolders" og naviger til mappen der toolboxen ligger. Trykk "save" etterpå.

MIR Toolbox — avansert lydanalyse

Lese inn en fil i variabelen minlyd :	minlyd = miraudio('minlydfil.wav')
Spill av filen:	mirplay(minlyd)
Spektrum (for hele filen)	mirspectrum(minlyd)
Spektrum i desibel	mirspectrum(minlyd,'dB')
Spektrogram	mirspectrum(minlyd,'frame')
Spektrogram i desibel	mirspectrum(minlyd,'frame','db')
Beregn energien for hele lydfilen:	d = mirrms(minlyd) Her legges svaret i variabelen "d"
Beregn energien over tid:	mirrms(minlyd, 'frame')
Konverter mirdata til vanlig matlab variabel:	data = mirgetdata(d);

- ▶ Bruk et semikolon på slutten av kommandoene for å ikke vise figur.
- ▶ For å lære mer om MIR toolbox: MUS4831 — Lydanalyse

En liten eksemplanalyse av lydfilen vår

Vi leser inn filen i variabelen bjork ved å skrive:
bjork = miraudio('hyperballadutsnitt.wav');

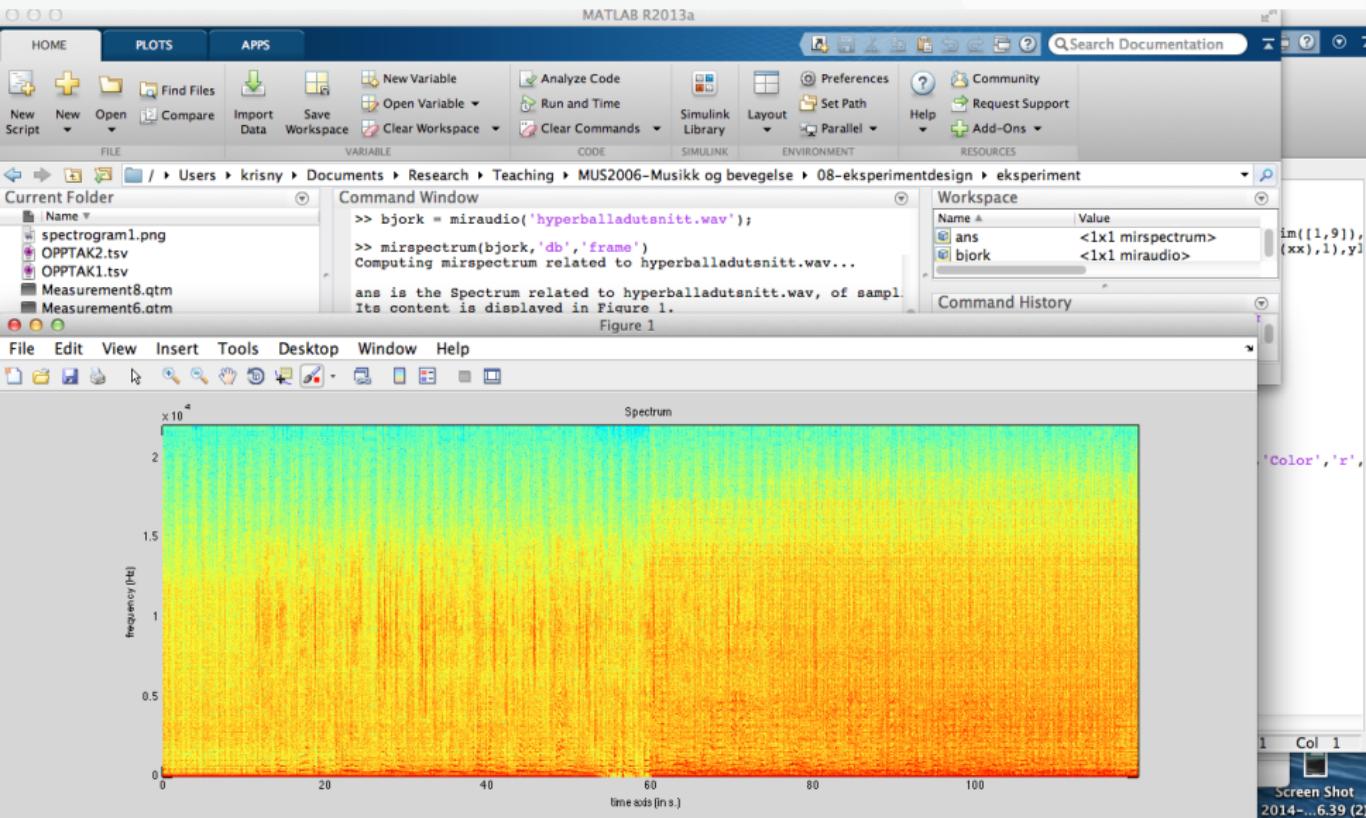
The screenshot shows the MATLAB R2013a interface with the following details:

- Toolbar:** HOME, PLOTS, APPS, New Script, New, Open, Find Files, Import Data, Save Workspace, New Variable, Open Variable, Run and Time, Clear Workspace, Clear Commands, Preferences, Help, Community, Request Support, Add-Ons.
- Current Folder Browser:** Shows files like spectrogram1.png, OPPTAK2.tsv, OPPTAK1.tsv, Measurement8.qtm, Measurement6.qtm, and hyperballadutsnitt.wav.
- Command Window:** Displays the command `>> bjork = miraudio('hyperballadutsnitt.wav');` and its execution output:

```
Vi leser inn lydfilen i variabelen bjork
Chunk 1/11...
Computing miraudio related to hyperballadutsnitt.wav...
Chunk 2/11...
Computing miraudio related to hyperballadutsnitt.wav...
Chunk 3/11...
Computing miraudio related to hyperballadutsnitt.wav...
Chunk 4/11...
Computing miraudio related to hyperballadutsnitt.wav...
Chunk 5/11...
Computing miraudio related to hyperballadutsnitt.wav...
Chunk 6/11...
Computing miraudio related to hyperballadutsnitt.wav...
Chunk 7/11...
Computing miraudio related to hyperballadutsnitt.wav...
Chunk 8/11...
Computing miraudio related to hyperballadutsnitt.wav...
Chunk 9/11...
Computing miraudio related to hyperballadutsnitt.wav...
Chunk 10/11...
Computing miraudio related to hyperballadutsnitt.wav...
Chunk 11/11...
Computing miraudio related to hyperballadutsnitt.wav...
Elapsed time is 4.430943 seconds.
```
- Workspace:** Shows the variable `bjork` as a `<1x1 miraudio>`.
- Command History:** Shows previous commands including `xlim([70 90])`, `ylim([0 1500])`, and `bjork = miraudio('hyperballadutsnitt.wav');`.
- Dimensions:** 999 x 684
- Page-Footer:** 1 of 10 selected, 19.78 GB available

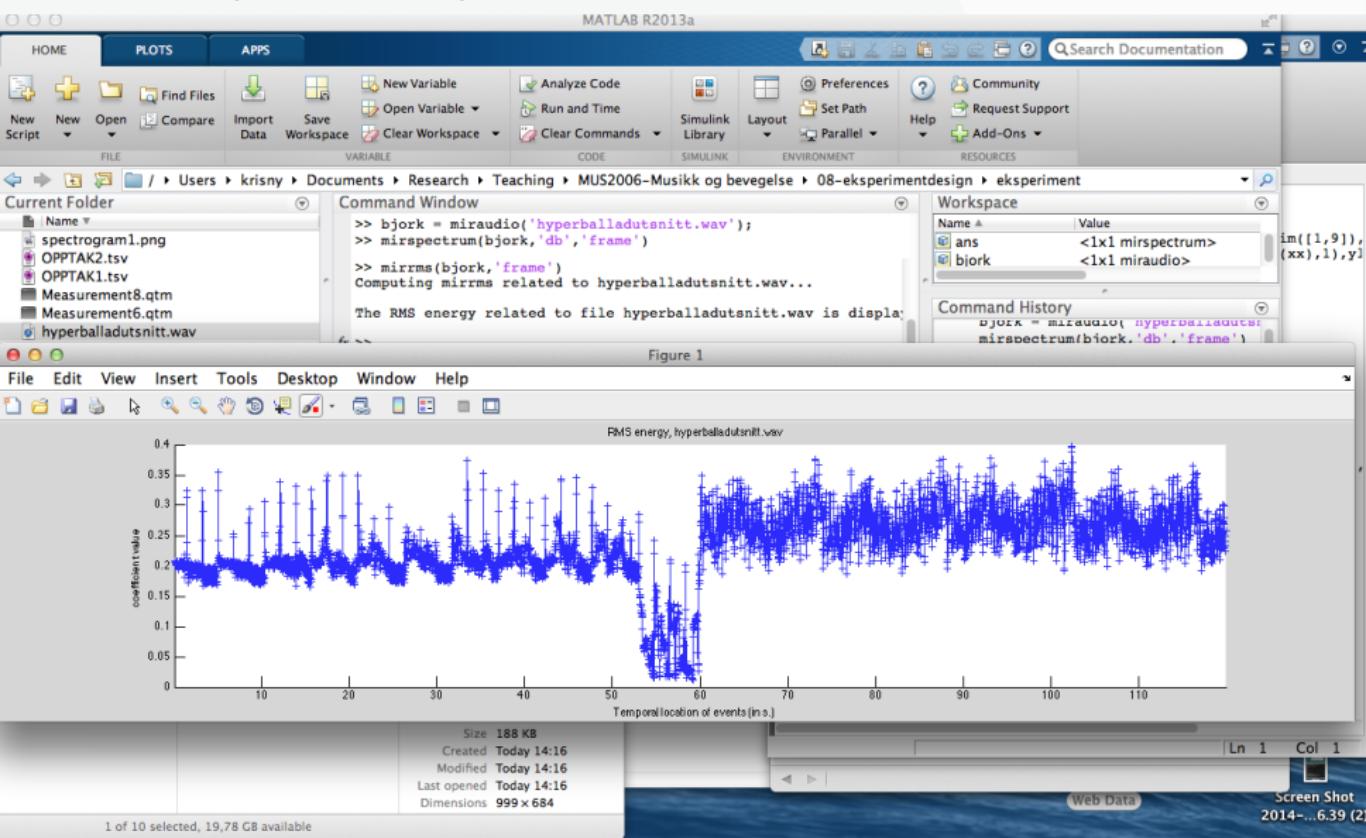
En liten eksemplanalyse av lydfilen vår

Vi kan lage et spektrogram med funksjonen `mirspectrum`:
`mirspectrum(bjork,'db','frame')`



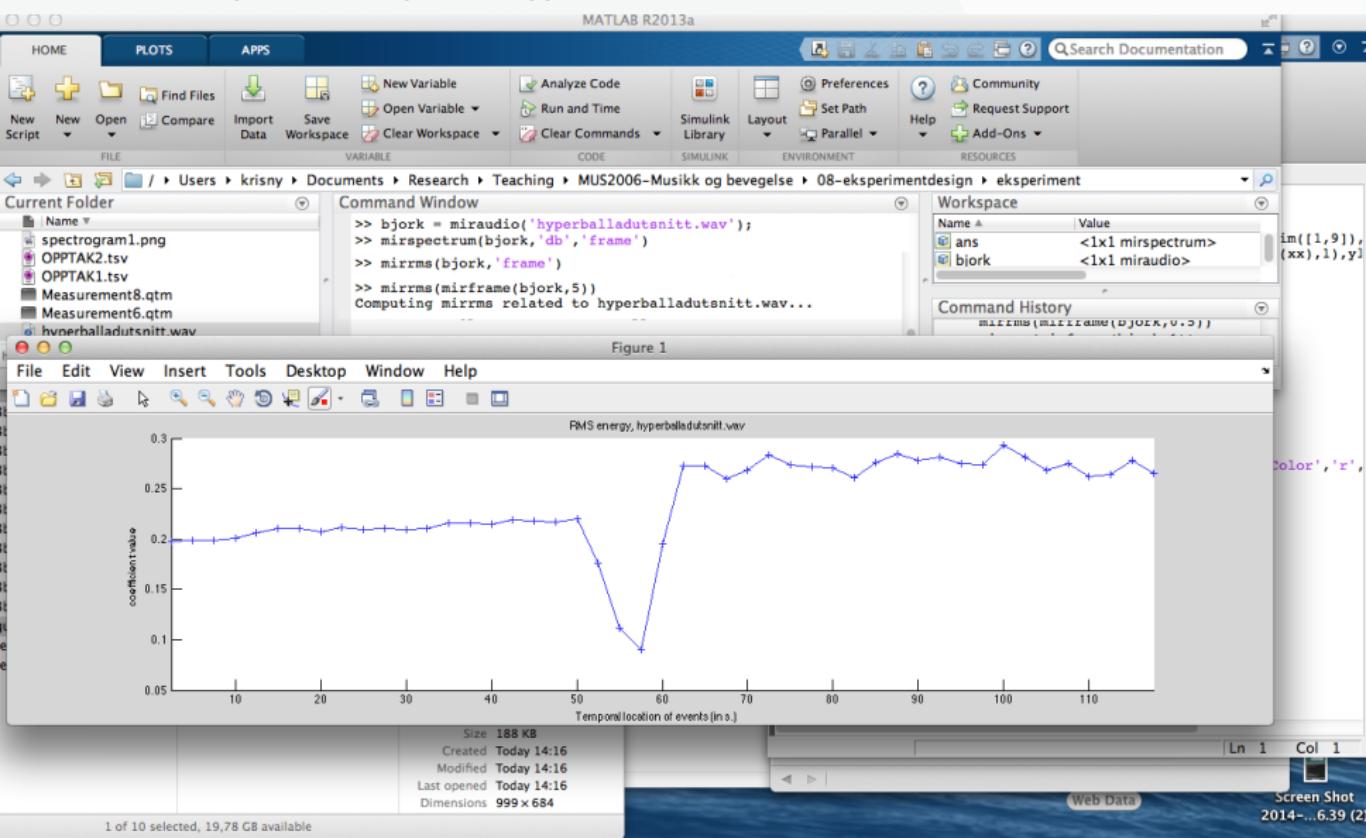
En liten eksemplanalyse av lydfilen vår

Vi kan analysere energien i lydfilen med funksjonen `mirrms`:
`mirrms(bjork,'frame')`



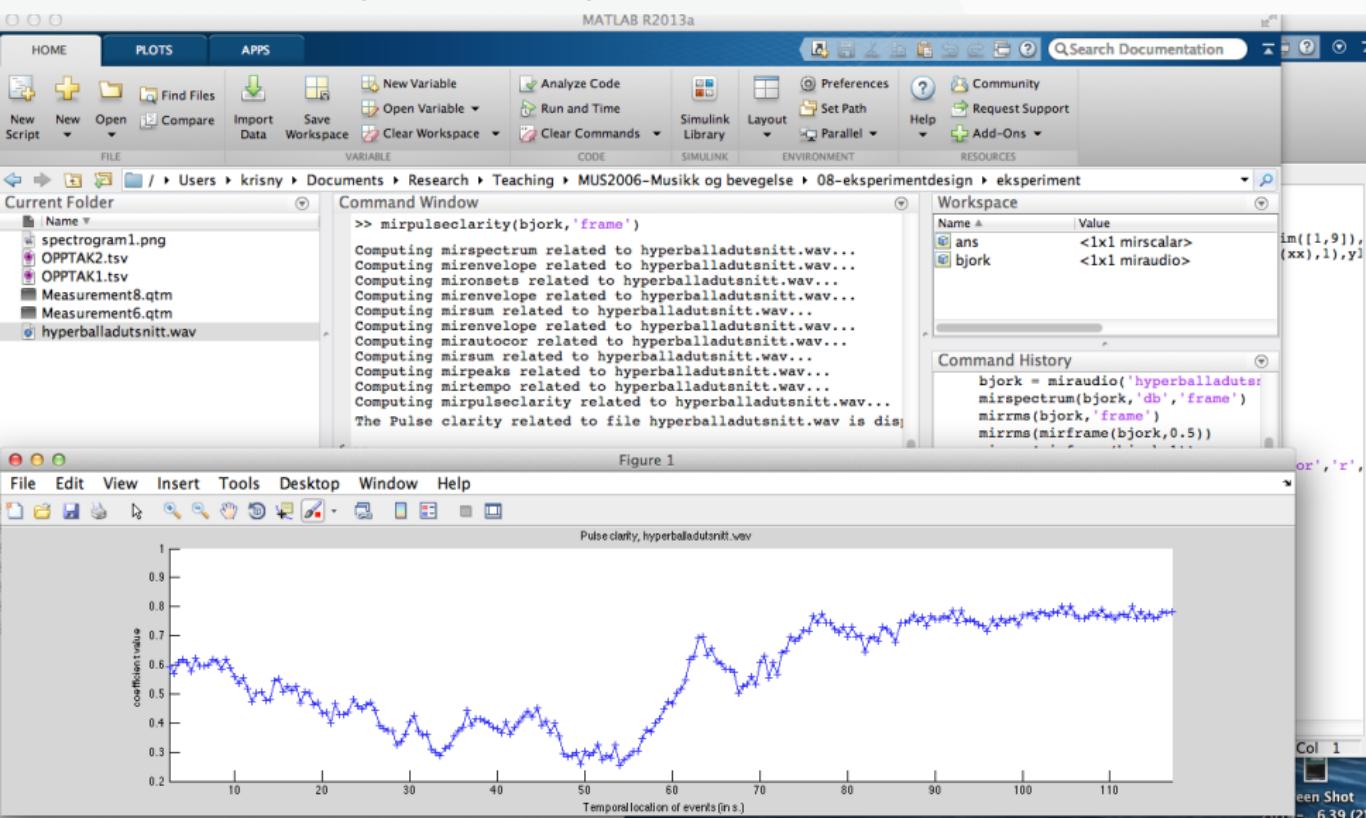
En liten eksemplanalyse av lydfilen vår

Hvis vi vil se på energien i lengre segmenter om gangen kan vi skrive:
mirrms(mirframe(bjork,5)) -ser på energien per 5 sek utsnitt



En liten eksemplanalyse av lydfilen vår

Funksjonen mirpulseclarity gir oss et mål på hvor tydelig rytmen er:
mirpulseclarity(bjork,'frame')



Lydanalyse

I dette kurset er ikke lydanalyse i fokus, men mange av metodene for kvantitativ lydanalyse likner på metodene for kvantitativ bevegelsesanalyse

En annen tilnærming til lydanalysen er å bruke enkle beskrivelser.
F.eks. kan en rask beskrivelse av Bjørk-utsnittet se slik ut:

Tid	Beskrivelse
0:00	Rolig, enkle rytmiske elementer.
0:11	Vokal kommer inn
0:53	Bass slutter. Enda roligere
1:00	Dynamikk opp, bass tilbake, basstromme "four to the floor"
1:14	Hihat kommer inn på offbeat
1:28	Skarptomme på 2 og 4
1:42	Vokaltema: "safe again"
1:57	Skarptomme forsvinner

Tidspunktene kan senere brukes som referansepunkter når vi analyserer bevegelsesdataene.

Bevegelsesanalyse i Mocap Toolbox

Referanseliste over nyttige funksjoner. Detaljer på forelesningen 27. mars.

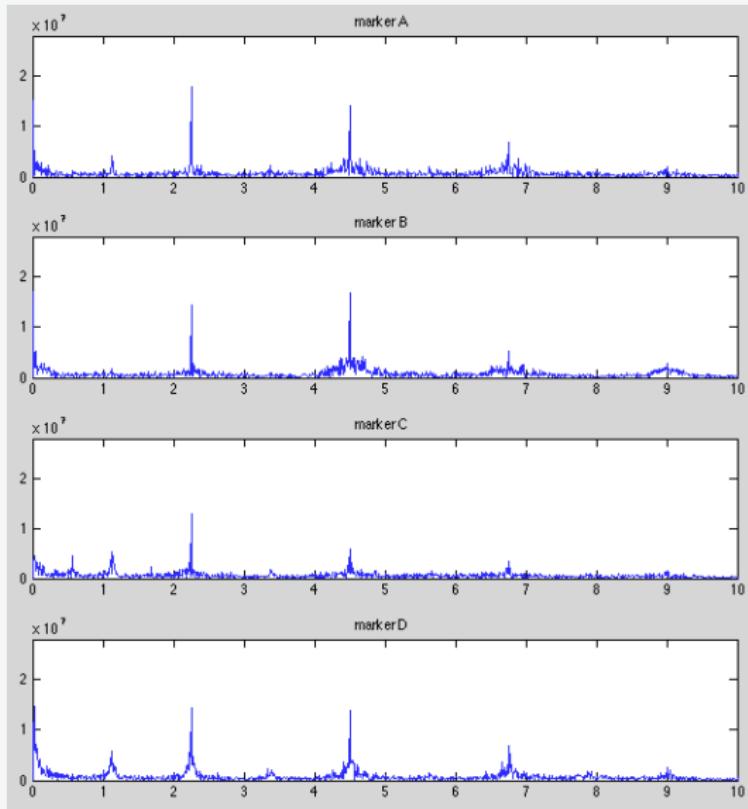
Lese inn en tsv-fil i variabelen a	a = mcread('OPPTAK1.tsv')
Lage en graf over XYZ-bevegelsen til markør 1	mcplottimeseries(a,1,'dim',[1 2 3])
Beregne hastighet (første derivert)	a1 = mctimeder(a) a1 inneholder nå XYZ-hastighet
Norm. Fra 3-dimensjonell data til 1-dimensjonell	a1n = mcnorm(a1) a1n inneholder nå <i>absolutt hastighet</i>
Beregne akselerasjon (andre derivert)	a2 = mctimeder(a,2) a2 inneholder nå XYZ-akselerasjon
Akselerasjon norm. / Absolutt akselerasjon	a2n = mcnorm(a2)
Lage en graf over absolutt akselerasjon for markør 2	mcplottimeseries(a2n,2)
Glatte ut / fjerne støy	aGlatt = mcsmoothen(a)
Plotte data i aGlatt med markørnavn	mcplottimeseries(aGlatt,[1:4],'names',1)

Bevegelsesanalyse — mer avansert

Slå sammen to opptak fra variablene a og b	c = mcconcatenate(a,1:4,b,1:4) 4 markører fra a og 4 markører fra b
Beregne og plotte spektrumet til markørene	[s,f]=mcspectrum(a1n); plot(f,s.data),xlim([0 s.freq/2])
Alternativ måte for i = 1:4 betyr: Repeter dette fire ganger. Første repetisjon er i = 1, neste gang med i = 2, så i=3, og til slutt i=4.	for i = 1:4 subplot(4,1,i); plot(f,s.data(:,i)); xlim([0 15]), ylim([0 max(max(s.data))]); set(gca,'XTick',[0:15]); title(s.markerName(i)); end
Autokorrelasjon	[p, ac, e, lag] = mcwindow(@mcperiod,a2n,8,0.05); for i=1:4 subplot(2,2,i) imagesc([1:size(p,1)],lag,squeeze(ac(:,:,i))); ylim([lag(6),lag(length(lag))]) end

Disse mer avanserte metodene er det ikke forventet at dere skal kunne bruke på dette kurset, men de er nevnt her fordi de er gode måter å visualisere periodiset på.

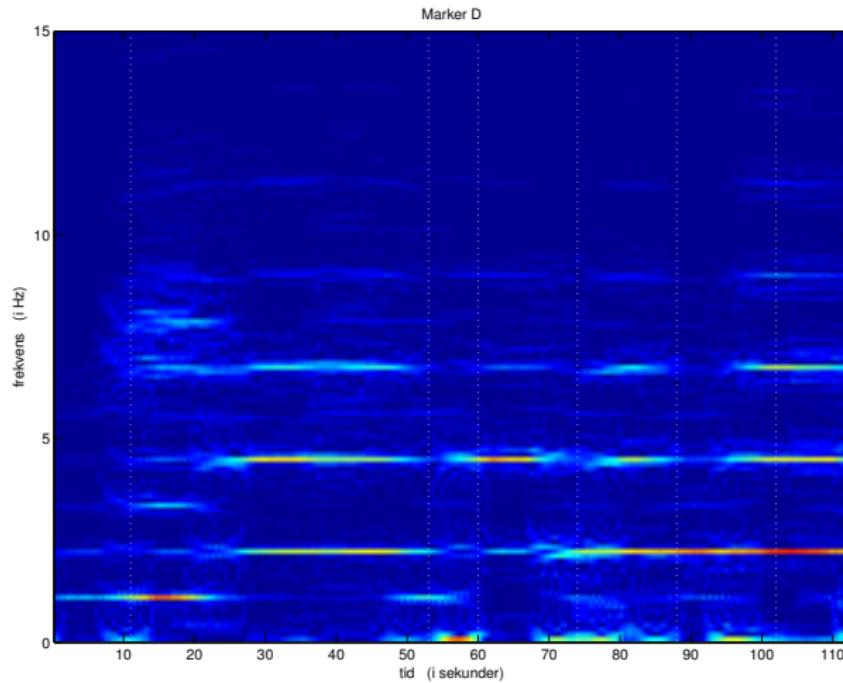
Bevegelsesanalyse — mer avansert



Slik ser **spektrumet** til absolutt akselerasjon for 4 av våre opptak ut. Det er klare peaks på ca 2.2 Hz og 4.5 Hz, som tilsvarer åttendelsnoter og fjerdedelsnoter i musikkseksempelet.

Disse viser altså spektrumet for hele bevegelsesopptaket

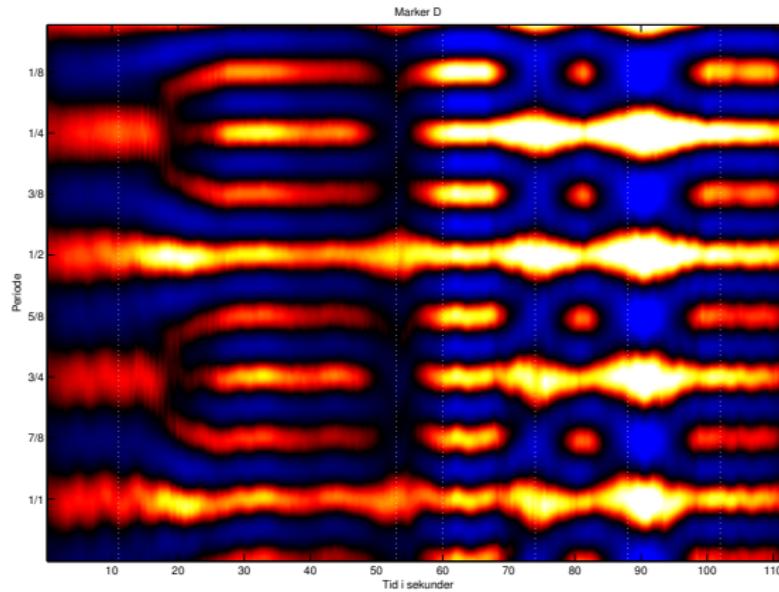
Bevegelsesanalyse — mer avansert



Dette er et **spektrogram** som viser hvordan spektrumet til absolutt akselerasjon for markør D utvikler seg over tid.

Vi ser peaks på ca 2.2 Hz og 4.5 Hz slik vi så i spektrumet på forrige side.

Bevegelsesanalyse — mer avansert



Dette viser **autokorrelasjonskoeffisienten** til absolutt akselerasjon for markør D utvikler seg over tid.

På y-aksen vises hvor ofte akselerasjonsmønsteret repeteres.