

T-61.140 Signaalinkäsittelyjärjestelmät

K7. Matlab/tietokonekäsittely 8.-11.3.2005.

Luo ensin Windowsissa alihakemisto Z:\SKJ2005. Jos teet UNIX-koneissa, niin ensin “use matlab”, sitten käynnistys “matlab” ja tee vastaava alihakemisto. Kuulokkeet tulevat äänikortin (vihreään) ulostuloreikään. Palauta lainatut kuulokkeet takaisin eteen tunnin jälkeen.

Matlabin editori aukeaa komennolla `edit`. Kirjoita syntyvä koodi ajokelpoisiin tiedostoihin työhakemistoosi. Komennolla `help` saa apua funktioiden käyttöön.

Portfoliosuoritusta varten tee muistiinpanoja ja kirjoita vastauksia tehtäviin A-E sekä merkitse läsnäolosi assistentin listaan. Lisäpisteitä (arvosanat 2-5) varten julkaistaan muutamia lisäpistetehtäviä, joiden palautus on keskiviikko 16.3.2005.

HUOM! Jos äänen kanssa ongelmia Linux/Matlab-tapauksissa, niin äänitiedostot voi kirjoittaa wav-muotoisiksi komennolla

```
soundsc(x,fs); % jos tämä EI TOIMI tai vaan MÖÖÖRRRIINÄÄÄÄÄ
wavwrite(x, fs, 8, 'ulos.wav');
```

jossa `x` on äänivektori, `fs` näytteenottotaajuus, `8` bittiä per näyte ja lopuksi tiedoston nimi. Tämän jälkeen äänitiedosto kuunnellaan Linuxissa sopivalla ohjelmalla, esim. `play`.

Tehtävät

- Lukujonon diskreetti Fourier-muunnos (DFT) saadaan komennolla `fft` ja käänteismuunnos `ifft`¹. DFT on siis diskreetti myös taajuustasossa. Kun pisteitä otetaan tarpeeksi paljon (N on iso), saadaan “illuusio” diskreettiaikaisesta Fourier-muunnoksesta $X(e^{j\omega})$:

$$X_N[k] = X(e^{j\omega})|_{\omega=2\pi k/N} = \left(\sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n} \right) |_{\omega=2\pi k/N}$$

Tutkitaan lyhyellä esimerkillä, kuinka Fourier-muunnos “kadottaa aikainformaation”.

Tehtävä A: Lataa vektori `signaali.txt` kurssin www-sivulta Matlabiin komennolla `load` (tai `textread`, `fscanf`, ...) ja talleta muuttujaksi `x`. Piirrä ensin koko signaali aikatasossa ja spektrin itseisarvo taajuustasossa tiedoston `K7prob1.m` komennoilla.

Laske alkuosan nopeasti muuttuvan sekvenssin jaksonaika $N_1 = \text{---}$, ja jälkiosan pienempi $N_2 = \text{---}$. Arvioi, pitävätkö $\omega_1 = 2\pi/N_1$ ja $\omega_2 = 2\pi/N_2$ paikkansa spektrikuvassa.

Tehtävä B: Piirrä sitten (a) signaalin alkuosa (indeksit 1:256, `osa1 = x(1:256)`) ja sen spektri, ja erikseen (b) signaalin loppuosa (`osa2 = x(257:512)`) ja loppuosan spektri sekä (c) vielä näiden summasignaali (`osa3 = osa1 + osa2`) ja spektri.

Poikkeavatko alkuperäisen signaalin (512 pitkä) ja `osa3`:n spektrit olennaisesti toisistaan? Voidaanko spektreistä yleisesti päätellä, onko kaksi eri taajuista sekvenssiä yhdistetty vai summattu yhteen?

Fourier-analyysi on perustyökalu signaalin tutkimiseen. Esimerkeistä käynee selville, että ei-stationäärinen signaali kannattaa kuitenkin jakaa pieniin paloihin ennen analyysiä! Näin tehdään aina esimerkiksi puheenkäsittelyssä, jossa aikaikkuna on usein 10-50 millisekuntia.

- Spektrogrammilla tarkoitetaan esitystä, jossa on mukana sekä aika- että taajuustaso.

Matlab-demolla `specgramdemo` voit suoraan kuunnella (Windows-kone) signaalia, jonka aaltomuoto esitetään alaiikkunassa. Signaalin kunkin ajanhetken spektri näkyy vasemmassa laatikossa ja kunkin taajuuskomponentin voimakkuus yläiikkunassa. Isossa keski-ikkunassa on spektrogrammi, jossa x-akselilla siis aika ja y-akselilla taajuus. Väri kertoo kunkin spektrikomponentin voimakkuuden tietyllä ajanhetkellä.

Funktio `specgramdemo` näyttää silloin tällöin kaatavan Matlabin, joten voit käyttää myös kommentoja:

```
yd = y + 0.001*randn(size(y)); % lisätään tahallisesti kohinaa
specgram(yd, [], fs);
colorbar
soundsc(y, fs); % kuunnellaan
```

Spektrogrammissa signaalia käsitellään pienissä ikkunoissa, joiden pituus on N_{win} . Ikkunassa lasketaan DFT-muunnos N_{fft} pisteessä $X_{N_{fft}}[k]$. Spektrogrammi on siis ajan mukana muuttuva sarja lyhyitä spektrejä (“short-time Fourier-spectrum”). Yleensä vierekkäisten ikkunoiden kannattaa myös olla osittain päällekkäin, johon viittaa muuttuja `Nlap`.

Tehtävä C: Generoi alla oleva signaali ja tutki sen spektrogrammia. Jos et ymmärrä, miten spektrogrammia luetaan, ota yhteyttä assistenttiin.

```
N = 400;
Fs = 8192;
t = [1 : N]/Fs;
x1 = cos(2*pi*697*t);
x2 = cos(2*pi*1209*t);
specgramdemo([x1 x2],Fs)
```

- Tutustu ohjelmaan Praat, joka on asennettu ATK-keskuksen Windows-koneisiin. Sen voi halutessaan myös ladata omalle kotikoneelleen osoitteesta <http://www.praat.org>.

Lue esimerkiksi tiedosto `kiisseli.wav` Praatiin. Kun valitset Sound-objektin, voit vaikkapa kuunnella tai piirtää sen. Valitse `Label & Segment - To TextGrid`, ylimmälle riville `foo` ja keskimmäiselle `bar`. Tämän jälkeen valitse sekä `Sound-` että `TextGrid-`objekti samanaikaisesti ja paina `Edit`. Avautuvassa ikkunassa ylhäällä on aaltomuoto ja keskellä spektrogrammi. Voit kuunnella osia signaalista. Koskettamalla rajalla olevaa palloa signaalia segmentoidaan pienempiin osiin.

Tehtävä D: Erottele ääniteitä (suomenkielisestä) puheesta. Pystyykö vierustoverisi tunnistamaan lyhyitä vokaaliääniteitä, jos ne on irrotettu pois kontekstistaan?

- Tutustu Praat-ohjelman demoon ääniväylän mallintamisesta ja äänen tuottamisesta.

Tehtävä E: Valitse `New - Articulatory Synthesis` ja seuraa tutoriaalini ohjeita. Ohjelma syntetisoi sanan “hallo” leukaperiä myöten.

¹itse asiassa samalla diskreettiaikainen Fourier-sarjaesitys