Lyhyt Matlab-opas

Matriisit

Matlabin keskeinen rakenne on matriisi ja esimerkiksi luvut Matlab käsittelee 1×1-matriiseina. Matriisien syöttäminen tapahtuu rivi kerrallaan, esimerkiksi

`A=[1,1,1,1,1;1,2,3,4,2;1,-1,1,-1,2;1,-1,-1,-1,2].`

määrittelee 4×5-matriisin A. Tärkeä syöttökomento on myös `x=[a:d:b]`, joka muodostaa matriisin `x`, jonka alkiot ovat `a`, `a+d,a+2d,...,a+kd`, missä k on suurin luku, jolle $a + kd \leq b$. Esimerkiksi `x=[0:0.01:4]` muodostaa 1×401-matriisin (vaakavektorin) `[0,0.01,0.02, ...,3.99,4]`. Jos `d` puuttuu, on oletusarvona `d=1`. Kun komennon perään kirjotetaan puolipiste (`x=[0:0.01:4];`) ei Matlab tulosta matriisia `x`.

Matriisit yhdistetään seuraavasti: Jos `A=[1,1;0,1]` ja `B=[0,1;1,1]` (2×2-matriiseja), on `C=[A B]` 2×4-matriisi ja `D=[A;B]` 4×2-matriisi. `A(i,j)` viittaa matriisin `A` alkioon rivillä i ja sarakkeella j. Sarakkeen j k ensimmäistä alkiota taas saadaan muodossa `A(1:k,j)` ja esimerkiksi vaakavektorin `x` viisi ensimmäistä alkiota saadaan kirjoittamalla `x(1:5)`. `size(A)` kertoo matriisin koon (tuloste on kaksi lukua 1×2-matriisissa). `sum(A)` laskee matriisin `A` alkioiden summan.

Matriisien `A` ja `B` yhteenlasku lasketaan komennolla `A+B`, kertolasku `A*B`, ja matriisin `A` transpoosi `A'`. Neliömatriisin determinantti saadaan komennolla `det(A)` ja kääntyvän matriisin käänteismatriisi komennolla `inv(A)`. Komento `rref(A)` muuntaa matriisin A redusoituun porrasmuotoon.

Komento `format rat` asettaa Matlabin laskemaan desimaaliesitysten sijaan murtoluvuilla.

Huom: Matlabiin ohjelmoidut funktiot, kuten sin kohdistuvat matriisin jokaiseen alkioon erikseen. Esimerkiksi `x=[0:0.1:2*pi];` tuottaa 1×63-matriisin `x=[0,0.1000,0.2000, ..., 6.1000,6.2000]`, ja samankokoinen on myös `y=sin(x)=[0,0.0998,0.1987, ...,-0.1822,-0.0831]`.

Sen sijaan matriiseille määritellyt operaatiot kertolasku `*` ja potenssiin korotus ˆ (haluttaessa syöttää ˆ Windows-käyttöliittymässä pitää ˆ kirjoittaa kaksi kertaa!) Matlab pyrkii toteuttamaan matriisikertolaskuna. Tämä voidaan kieltää kirjoittamalla `.*` ja `.ˆ` (huom piste). Esimerkiksi matriisin `A` toinen potenssi lasketaan seuraavasti: `Aˆ2`, mutta matriisin `A` *alkioiden* toinen potenssi taas seuraavasti: `A.ˆ2`

Esimerkki: Lasketaan yhteen luvut $(99/100)^{10}$, $(99/100)^{11}$, $\ldots$ $(99/100)^{100}$. Tämä voidaan tehdä seuraavasti: `n=[10:100]`, `A=(99/100).ˆn` ja `sum(A)`.

Kuvaajan piirtäminen

Jos `x` ja `y` ovat samanpituisia vaakavektoreita, piirtää `plot(x,y)` piirtää kuvaajan, jossa pisteet `(x(i),y(i))` kaikilla i:n mahdollisilla arvoilla ovat liitettyjä janalla toisiinsa. Paraabelin $y = x^2$ kuvaaja välillä $[-1, 3]$ saadaan esimerkiksi seuraavasti: `x=[-1:0.01:3]`, `y=x.^2`, `plot(x,y)`.

Matlabin `plot`-komento on luontevin parametriesityksen yhteydessä: Jos on piirrettävä käyrän $\{(x(t)), y(t) \mid t \in I\}$ kuvaaja, muodostetaan I riittävän tihein välein parametrin arvoja esittävään vaakavektoriin `t`, minkä jälkeen lasketaan `x(t)` ja `y(t)`. Lopuksi `plot(x,y)` tuottaa kuvaajan. Esimerkiksi `t=[0:0.01:2*pi]`, `x=cos(t)`, `y=sin(t)` ja `plot(x,y)` piirtää yksikköympyrän.

Huom: `plot`-komennossa on myös mahdollista, että vektorit `x` ja `y` ovat pystyvektoreita, tai että toinen on pysty- ja toinen vaakavektori. Lisäksi on mahdollista, että `x` on vektori, mutta `y` on matriisi. Esimerkiksi `x=[-3:0.01:3]`, `y=[2*x;x.^2]`, `plot(x,y)` piirtää kaksi kuvaajaa.

Polynomit

Polynomit syötetään antamalla niiden kertoimet vaakavektorina. Esimerkiksi polynomi $p(x) = x^3-2x+1$ annetaan asettamalla `p=[1,0,-2,1]`. Polynomim arvo lasketaan komennolla `polyval(p,x)`, missä `p` on polynomi ja x on piste, missä polynomin arvo lasketaan (voi myös olla vaakavektori). Polynomien `p` ja `q` kertolasku lasketaan komennolla `conv(p,q)`.

Omien funktioiden määrittely

Funktiot kirjoitetaan Matlabissa ns. M-tiedostoiksi (file → new → M-file). Esimerkki:

```
function y = esim(x)
% ESIM on paloittain määritelty funktio,
% lineaarinen, kun x>0 ja neliöllinen muutoin.
if (x>0)
     y=x+1
else y=x^2
end;
```

Differentiaaliyhtälöiden numeerisesta ratkaisemisesta

Ratkaistaan esimerkiksi demotehtävässä 16.2. esiintynyt differentiaaliyhtälö

$$\begin{aligned} & 0,01 \cdot I'' + 20 \cdot I' + 10^4 I = 32500\pi \cos(100\pi t) \\ \Longleftrightarrow \quad & I'' + 2 \cdot 10^3 \cdot I' + 10^6 I = 325 \cdot 10^4 \pi \cos(100\pi t) \end{aligned}$$

numeerisesti välillä $[0, 0, 1]$. Muutetaan ensin differentiaaliyhtälö DY-ryhmäksi, jossa esiintyy vain ensimmäisen kertaluvun derivaattoja. Tätä varten otetaan käyttöön uusi funktio $J = I'$, jolloin saadaan DY-pari

$$\left\{ \begin{array}{lcl} I' & = & J \\ J' & = & -2 \cdot 10^3 J - 10^6 I + 325 \cdot 10^4 \pi \cos(100\pi t) \end{array} \right. .$$

Tämän numeeriseen ratkaisemiseen voidaan käyttää seuraavanlaista Matlab-koodia:

```
function demo1602
[t,tulos]=ode23(@RCL,[0 0.1],[0 0]);
plot(t,tulos(:,1))
return

%Yllä olevat rivit käyttävät ode23-ohjelmaa DY-parin likimääräiseen
%ratkaisemiseen ja kuvaajan piirtämiseen. tulos(:,1) poimii
%matriisin tulos 1. sarakkeen.

function arvo=RCL(t,IJ)
I=IJ(1);
J=IJ(2);
arvo(1,1)=J;
arvo(2,1)=-2*10^3*J-10^6*I+325*10^4*pi*cos(100*pi*t);
return

%Yllä oleva määrittelee funktion RCL. Funktion arvo on 2-pituinen
%pystyvektori, joka Matlabissa voidaan esittää 2*1-matriisina:
%arvo(1,2) on 1. alkio ja arvo(2,1) pystyvektorin 2. alkio.
```