Teknillinen korkeakoulu Apiola
Matematiikka

Mat-1.415 Matematiikan peruskurssi V3 syksy 2000
`http://www.math.hut.fi/teaching/v/3/H/`

Laskuharjoitus 5 (viikko 42 , 17–19.10)

Täksi viikoksi tehtäviä on vain torstain harjoitukseen. Tiistaina kommentoidaan välikoeratkaisuja sekä tehdään ohjatusti torstain tehtäviä samoin kuin keskiviikkonakin.

Matlab- ja Maple-apuvälineitä:

```
www.math.hut.fi/teaching/v/3/L/L6.html  Matlab-skripti trajektoriparviin
www.math.hut.fi/teaching/v/3/L/Lmaple6.html  Maple-työskentelyä
www.math.hut.fi/teaching/v/3/H/ratk/harj4av.html  -- '' --
www.math.hut.fi/teaching/k3/luentomateriaali/L20maple.html
```

Tehtävät torstaiksi

1. Harj. 3 loppuviikon tehtävä 5 ("hauska pikku projekti")
2. Harj. 4 Loppuviikon tehtävä 5 (BdiP s. 390 teht. 29, jousisyst.)
3. (Puhdas käsinlaskutehtävä) Osoita luennolla hahmotellulla tavalla (tai muuten), että alla sanottu matriisieksponenttifunktion määritelmä on tosiaan kunnossa. Tarvitset kahta asiaa:
   1) $e^x = 1 + x + \frac{x^2}{2!} + \ldots + \frac{x^n}{n!} + \ldots$ suppenee kaikilla $x \in \mathbb{R}$.
   2) A^2 ei kasva tolkuttomasti.

   Kasvun tutkimisessa voit rajoittua vakiomatriisiin, koska sillekin asian tulee päteä ja toisaalta mielivaltaisen matriisin tapauksessa saadaan yläraja-arvio korvaamalla kaikki matriisin alkiot itseisrvoltaan suurimmalla.

   Nyt homma onkin helppo hoitaa!
4. Laske kynää ja paperia ja tarvittaessa esim. Maplea matriisilaskimena käyttäen: e^{At} matriiseille
   a) $A = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}$ b) $A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$ c) $A = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$

   Kommentteja: a) on ääliömäisen helppo, b) on "nilpotentti", korota potenssiin, niin näet, mitä tarkoittaa, c):ssä näkyy tietty jaksollisuus.

   Tarkista Maplella ja kokeile lisäksi tyyliä

   ```
   seq(evalm((t*A)&^k)/k!,k=0..10); map(series,eat,t=0,10);
   ```

   olevia temppuja joidenkin matriisien kohdalla. (Huomaat, että matriisieksponenttifunktio ei ole ollenkaan kummallinen olio, ehkä siihen kehittyy riippuvuus!)
5. Ratkaise $\mathbf{x}' = A\mathbf{x}$, $\mathbf{x}(0) = [0, 1, 1]$ kun
   $$A = \begin{bmatrix} -14 & -160 & -40 \\ 181 & 5 & 2 \\ 96 & 84 & 18 \end{bmatrix}$$
   Käytä matlab-hakemiston `linsys`-funktiota (jonka toiminta perustuu Matlabin sisäänrakennettuun e^{At}-funktioon nimeltään `expm`, kts. myös `...L/L_ldys.html`).

   Piirrä ratkaisukäyräparvi (aikariippuvuuskuva) sekä faasiavaruuskuva. Piirrä samaan kuvaan reaalinen ominaisvektori ja kompleksisen ominaisvektorin re- ja imosien muodostamat vektorit. Matlabissa *real, imag* . Näiden "ominaisakselien" piirto voisi sujua seuraavaan tyyliin:

   ```
   sopivakerroin=3 ; % tms.
   aks=sopivakerroin*v1;plot3([0,aks(1)],[0,aks(2)],[0,aks(3)],'r')
   ...
   ```

   Kokeile myös sellaisilla alkuarvoilla, jotka sijaitsevat kahden ominaisvektorin (Re/Im-osien) määräämillä tasoilla.
6. Luennolla osoitettiin, että epähomogeenisen systeemin
   $$\mathbf{x}' = A\mathbf{x} + \mathbf{g}(t), \quad \mathbf{x}(0) = \mathbf{x}^{(0)}$$
   ratkaisu on
   $$\mathbf{x}(t) = e^{At}\mathbf{x}^{(0)} + \int_0^t e^{A(t-s)}\mathbf{g}(s)ds.$$
   Vektorifunktion integrointi tarkoittaa yksinkertaisesti integrointia komponenteittain.

   Ratkaise tämän avulla systeemi
   $$\mathbf{x}' = \begin{bmatrix} 1 & 4 \\ -1 & -3 \end{bmatrix}\mathbf{x} + \begin{bmatrix} t \\ 1 \end{bmatrix}, \quad \mathbf{x}(0) = (1, -1)$$
   Voit katsoa mallia tästä:
   `www.math.hut.fi/teaching/k3/luentomateriaali/L20maple.html`

   Mallista poiketen saat tässä oikaista niin, että käytät valmista `linalg[exponential]`-funktiota (tätäpä ei näy uudesta `LinearAlgebra`)-kirjastosta löytyvän.)

Matriisieksponenttifunktio e^{At}

Jokaiselle neliömatriisille A voidaan määritellä eksponenttifunktio $e^A := \sum_{k=0}^{\infty} \frac{A^k}{k!}$. Diagonalisoituvalle matriisille se voidaan laskea suoraan soveltamalla diagonalisointiesityksen D-matriisin diagonaalialkioihin tavallista exp-funktiota. Joskus voi olla kätevää soveltaa sopivaa reaalifunktion sarjakehitelmää, joskus taas yllä mainittu määritelmä johtaa katkeavaan sarjaan yllättävästi syystä, jota lukujen maailmassa ei esiinny, nimittäin matriisin jokin potenssi voi olla nolla (nilpotentti matriisi).

Miten sitten toimitaan yleisesti, kun ei kuitenkaan haluta käyttää sarjakehitelmää? Silloin etsitään ns. yleistetyt ominaisvektorit, mikä vastaa sitä tapaa, joka esitettiin 2×2-matriiseille. Yleisessä muodossa esitettynä se voidaan muotoilla matriisihajotelmaksi, joka on "lähes diagonalisointi", eli ns. *Jordanin muoto*. Sitä emme tällä kurssilla lähemmin opiskele, kiinnostuneet löytävät sen mm. kelpo Laode-kirjasta.

Algoritmi voidaan esittää ilman Jordan-hajoitelma-terminologiaa (vaikka samasta asiasta on kyse).

Itse asiassa pätee ns. *Cayley-Hamiltonin* lause: Jokainen neliömatriisi toteuttaa oman karakteristisen yhtälönsä. (Tarkoittaa, että jos lausekkeessa $p(\lambda) = \lambda^n + \ldots$ sijoitetaan λ:n paikalle A, niin tuloksena on nollamatriisi.) Tästä päädytään siihen, että matriisi "käyttäytyy nilpotentisti" yleistettyjen ominaisvektorien edustamilla suunnilla. (Kts. tarkemmin alla mainituilta sivuilta).

Helpoimmat tapaukset ovat ääritapaukset:
a) Jos ominaisvektoreita on riittävästi (n kpl) tai
b) jos niitä on vain yksi (jolloin ominaisrvojakin on vain yksi).
Tällöinhän yllä mainitusta *Cayley-Hamiltonista* seuraa, että jos λ_0 on tuo ominaisarvo, niin $(A - \lambda_0)^n = 0$. Koska $e^{At} = e^{\lambda_0 t} e^{(A-\lambda_0)t}$, saadaan $(A - \lambda_0 I)t$:n potenssien mukainen sarja, joka katkeaa n:nnessä termissä.

2×2-matriisien käsittely on erityisen helppoa, koska niillä joko on riittävästi (2) ominaisvektoreita tai $(A - \lambda_0)^2 = 0$

Algoritmi on selitetty tarkemmin luentosivuilla: (Emme sitä kuitenkaan kokeissa vaadi osattavaksi.)

`www.math.hut.fi/teaching/v/3/L/L_ldys.html` ja tämän viitteissä:
`www.math.hut.fi/teaching/k3/luentomateriaali/expA.html` ja erit.
`www.math.hut.fi/teaching/k3/luentomateriaali/eat8.gif`
`www.math.hut.fi/teaching/k3/luentomateriaali/eat9.gif`

Käytännössä voidaan turvautua ohjelmistoihin, joihin tuo Jordan-muoto on kätketty. MAPLE:ssa on `linalg[exponential]` ja MATLAB:ssa `expm`.