DN1212 Numeriska metoder och grundläggande programmering FN6 09-03-17

Hedvig Kjellström
hedvig@csc.kth.se

Denna föreläsning

- Repetition av FN5 (GNM kap 6.1-2B)
  - Differentialekvationer
  - Standardform för begynnelsevärdesproblem
  - Stegmetoder (Finita differensmetoder)
  - Enstegsmetoder: Eulers metoder
- Begynnelsevärdesproblem forts. (GNM kap 6.1G-2G)
  - Stabilitet och instabilitet
  - Flerstegsmetoder: Runge-Kutta

Repetition av FN5 (GNM kap 6.1-2B)

Differentialekvationer

- Viktigt verktyg för att beskriva naturvetenskapliga, tekniska processer
- Uppdelning 1:
  - Ordinär differentialekvation (ODE)
    $$\frac{du}{dt} = sin(t) - 2u, u(0) = 0$$
  - Partiell differentialekvation (PDE) (vågekvationen i 1 dim)
    $$\frac{\partial^2 u}{\partial t^2} = \nu^2 \frac{\partial^2 u}{\partial x^2}$$
  - Vi studerar bara ODE i denna kurs

Differentialekvationer

- Idag koncentrerar vi oss på **begynnelsevärdesproblem**
- Uppdelning 2:
  - Linjär ODE (kan lösas analytiskt)
    $$\frac{du}{dt} = sin(t) - 2u, u(0) = 0$$
  - Ickelinjär ODE (kan oftast inte lösas analytiskt)
    $$\frac{du}{dt} = -u^2 + t^2, u(1) = 1$$
  - Vi ska studera numerisk lösning av ickelinjära ODE

Differentialekvationer

- Uppdelning 3:
  - Första ordningens ODE
    $$\frac{du}{dt} = sin(t) - 2u, u(0) = 0$$
  - Högre ordningens ODE

$$m\frac{d^2u}{dt^2} + c\frac{du}{dt} + ku = Asin(\omega t), u(0) = 0, \frac{du}{dt}(0) = 0$$

- I fortsättningen:

$$\frac{d}{dt} \equiv \cdot, \frac{d^2}{dt^2} \equiv \ddot{}, \frac{d}{dx} \equiv \ ', \frac{d^2}{dx^2} \equiv \ '', \frac{d^n}{dx^n} \equiv \ ^{(n)},$$

Standardform för begynnelsevärdesproblem

- Första ordningens ODE med begynnelsevillkor
$$y' = f(x, y), y(a) = c$$
- Eller system av ODE
$$\begin{cases} y' = f(x, y, z, \ldots), y(a) = c \\ z' = g(x, y, z, \ldots), z(a) = d \\ \ldots \end{cases}$$
$$\Leftrightarrow$$
$$\mathbf{u}' = \mathbf{f}(x, \mathbf{u}), \mathbf{u}(a) = \mathbf{c}$$

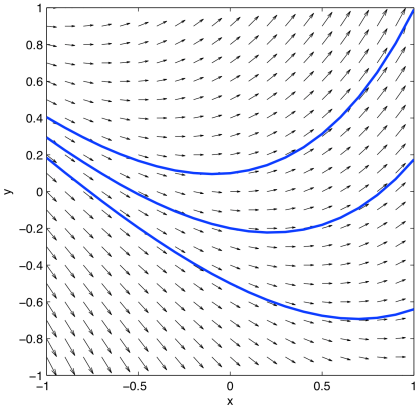
Lösningar

- Lösning – en funktion
- ODE $y' = -xy$
- …har lösningen $y = Ae^{-x^2/2}$
- Detta är en **lösningsskara**
- Begynnelsevillkor definierar lösning ur skaran:

$$y(0) = 1 \Rightarrow A = 1$$

Riktningsfält

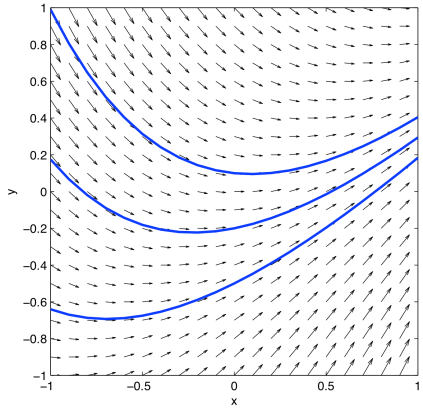
- Grafisk teknik för att bilda sig en uppfattning om lösningsskaran



$$y' = x + y$$

Riktningsfält

- Ger bild av systemets **kondition** ≈ fel i slutet / fel i början



$$y' = x - y$$

Olinjära ODE

- Kan lösas numeriskt
  - Skrivs om på standardform $\mathbf{u}' = \mathbf{f}(x, \mathbf{u}), \mathbf{u}(a) = \mathbf{c}$
- Approximativt (som alltid)
- Approximation = hoppar mellan funktioner i lösningsskaran

Stegmetoder (finita differensmetoder)

- Skriv om ekv på standardform

$$\mathbf{u}' = \mathbf{f}(x, \mathbf{u}), \mathbf{u}(a) = \mathbf{c}$$

- Starta i begynnelsepunkten
- Ta små steg i derivatans riktning
- Trunkeringsfel = hopp mellan kurvor i lösningsskaran
  - **Lokalt trunkeringsfel** = felet vid ett steg
  - **Globalt trunkeringsfel** = det ackumulerade lokala felet
  - Mer om kondition, indatafel, trunkeringsfel och stabilitet i FN6

Euler framåt (Explicit Euler)

- Enklaste stegmetoden

$$\begin{cases} y_{i+1} = y_i + h \cdot f(x_i, y_i), & y_0 = c \\ x_{i+1} = x_i + h, & x_0 = a \end{cases}$$

- Används för
  - Grova uppskattningar
  - Förståelse
  - …men inte för att lösa diffekv noggrant

```
x = a;
u = c;
while x < b
  u = [u, u(end) + h*f(x(end), u(end))];
  x = [x, x + h];
end
```

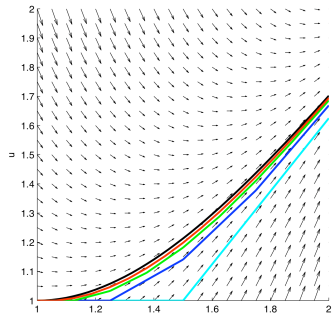
- Noggrannhetsordning p = 1

Euler framåt (Explicit Euler)

- Matlabexempel

$$\dot{u} = -u^2 + t^2, u(1) = 1$$

- Plot:



- Utskrift:

```
   h        E_h    E_2h/E_h=2
0.5000   7.689e-02   13.006
0.2500   3.303e-02    2.328
0.1250   1.589e-02    2.078
0.0625   7.788e-03    2.041
Linjärt avtagande trunkeringsfel - noggrannhetsordning p = 1
```


Euler bakåt (Implicit Euler)

- Problem med Euler framåt : **instabil** (mer i FN6)
- Formulera istället steg som

$$\begin{cases} y_{i+1} = y_i + h \cdot f(x_{i+1}, y_{i+1}), & y_0 = c \\ x_{i+1} = x_i + h, & x_0 = a \end{cases}$$

- Måste skriva om ekv (1) som

$$\begin{cases} y_{i+1} = g(x_i, y_i, h), & y_0 = c \\ x_{i+1} = x_i + h, & x_0 = a \end{cases}$$

- Noggrannhetsordning p = 1

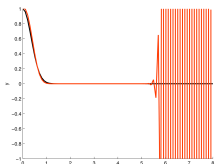
Euler bakåt (Implicit Euler)

- Matlabexempel

$y' = -8xy, y(0) = 1$, lösn $y = e^{-4x^2}$

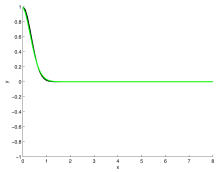
- Euler framåt

```
x = 0; y = 1; h = 0.1;
while x < 8
  y = y + h*(-8*x*y);
  x = x + h;
end
```



- Euler bakåt

```
x = 0; y = 1; h = 0.1;
while x < 8
  y = y/(1 + 8*h*(x + h);
  x = x + h;
end
```

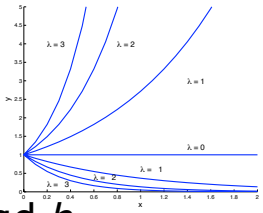


Begynnelsevärdesproblem forts. (GNM kap 6.1G-2G)

Stabilitet

- Varför är Euler bakåt stabilare än Euler framåt för samma steglängd *h*?
- Betrakta testekvationen

$y' = \lambda \cdot y, y(0) = 1,$ lösning $y = e^{\lambda x}$



- Stegmetod med steglängd *h*
  - **Stabil** om $|y_{i+1}| \leq |y_i|$
  - **Instabil** om $|y_{i+1}| > |y_i|$ eller **svagt stabil** om $|y_{i+1}| > |y_i|,\ \text{sign}(y_{i+1}) \neq \text{sign}(y_i)$

Stabilitet

- Euler framåt

$y_{i+1} = y_i + h\lambda \cdot y_i = y_i \cdot (1 + h\lambda)$

  - Dvs stabil om $|1 + h\lambda| \leq 1$ dvs $-2 \leq h\lambda \leq 0$

- Euler bakåt

$y_{i+1} = y_i + h\lambda \cdot y_{i+1}$

$y_{i+1} - h\lambda \cdot y_{i+1} = y_i$

$y_{i+1} = \dfrac{y_i}{1 - h\lambda}$

  - Dvs stabil om $\dfrac{1}{|1 - h\lambda|} \leq 1$ dvs $h\lambda \leq 0$ eller $h\lambda \geq 2$

Mycket större område! …men ett problem att lösning *avtagande* för stora positiva λ.

Runge-Kuttas metoder

- Eulers metoder, noggrannhets-ordning *p* = 1
- Specialfall (1:a ordn) av R-K
- Nästa specialfall (2:a ordn) är **Heuns metod**

$$k_1 = hf(x_i, y_i)$$
$$k_2 = hf(x_i + h, y_i + k_1)$$
$$y_{i+1} = y_i + (k_1 + k_2)/2$$
$$x_{i+1} = x_i + h$$

- Noggrannhetsordning *p* = 2

Runge-Kuttas metoder

- R-K av 4:e ordn är det som kallas **Runge-Kuttas metod**

$$k_1 = hf(x_i, y_i)$$
$$k_2 = hf(x_i + h/2, y_i + k_1/2)$$
$$k_3 = hf(x_i + h/2, y_i + k_2/2)$$
$$k_4 = hf(x_i + h, y_i + k_3)$$
$$y_{i+1} = y_i + (k_1 + 2k_2 + 2k_3 + k_4)/6$$
$$x_{i+1} = x_i + h$$

- Noggrannhetsordning $p = 4$

Vad innebär noggrannhetsordningen?

- Matlabexempel $\dot{u} = -u^2 + t^2, u(1) = 1$
- $R_h = u_h(2)$, $E_h \approx |R_{2h} - R_h|$
- Utskrift:

Inte samma som konvergensordning, lab 4.1
Ger samma info som loglogplot, lab 4.3

	Euler		Heun		Runge-Kutta	
h	Rh	E2h/Eh=2	Rh	E2h/Eh=4	Rh	E2h/Eh=16
0.5000	1.625000e+00		1.823256e+00		1.711293e+00	
0.2500	1.668863e+00		1.715914e+00		1.702201e+00	
0.1250	1.685998e+00	2.560	1.704284e+00	9.230	1.701903e+00	30.527
0.0625	1.694102e+00	2.114	1.702396e+00	6.159	1.701890e+00	22.964
0.0312	1.698035e+00	2.060	1.702007e+00	4.849	1.701889e+00	19.254
0.0156	1.699972e+00	2.031	1.701918e+00	4.374	1.701889e+00	17.564
0.0078	1.700933e+00	2.016	1.701896e+00	4.176	1.701889e+00	16.766
0.0039	1.701412e+00	2.008	1.701891e+00	4.085	1.701889e+00	16.379
0.0020	1.701651e+00	2.004	1.701890e+00	4.042	1.701889e+00	16.191
0.0010	1.701770e+00	2.002	1.701890e+00	4.021	1.701889e+00	16.093
0.0005	1.701830e+00	2.001	1.701889e+00	4.010	1.701889e+00	16.311

Trapetsmetoden

- ”Medelvärde” mellan Euler framåt och Euler bakåt

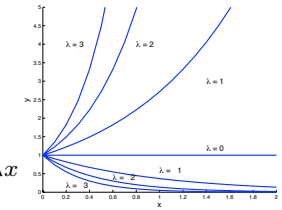
$$\begin{cases} y_{i+1} = y_i + \frac{h}{2} \cdot (f(x_i, y_i) + f(x_{i+1} + y_{i+1})), & y_0 = c \\ x_{i+1} = x_i + h, & x_0 = a \end{cases}$$

- Implicit, kräver att första ekv stuvas om

$$\begin{cases} y_{i+1} = g(x_i, y_i, h), & y_0 = c \\ x_{i+1} = x_i + h, & x_0 = a \end{cases}$$

- Noggrannhetsordning $p = 2$
  - Heter trapetsmetoden eftersom f integreras över $[x_i, x_{i+1}]$ och integralen approximeras enligt trapetsregeln
  - Lik Heuns metod – men Heun explicit

Trapetsmetodens stabilitetsområde?



- Testekvationen

$$y' = \lambda \cdot y, y(0) = 1, \text{ lösning } y = e^{\lambda x}$$

- Stabil lösning om $|y_{i+1}| \leq |y_i|$

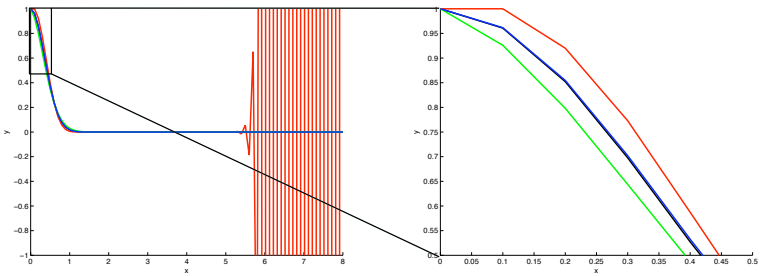
$$y_{i+1} = y_i + \frac{h}{2}(\lambda \cdot y_i + \lambda \cdot y_{i+1})$$
$$y_{i+1} - \frac{h}{2}\lambda \cdot y_{i+1} = y_i + \frac{h}{2}\lambda \cdot y_i$$
$$y_{i+1} = \frac{1 + \frac{h}{2}\lambda}{1 - \frac{h}{2}\lambda} \cdot y_i$$

  - Dvs stabil om $\frac{1 + \frac{h}{2}\lambda}{1 - \frac{h}{2}\lambda} \leq 1$ dvs $\lambda \leq 0$

Steget h kan vara hur stort som helst om detta är uppfyllt! (Trunkeringsfelet blir dock större med h.)

Trapetsmetodens stabilitetsområde?

- Matlabexempel $y' = -8xy, y(0) = 1$
- Plot:



Flerstegsmetoder

- Hittills enstegsmetoder
  - Indata (x_i, y_i)
  - Utdata (x_{i+1}, y_{i+1})
- Man kan dock använda flera punkter som indata
- Tex, **mittpunktsmetoden**
  - Utgår från **centraldifferensapproximation** (FN4)

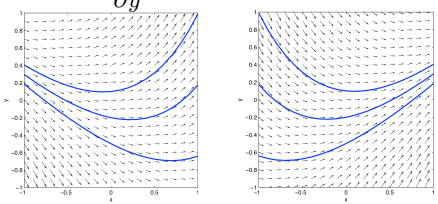
$$y'(x_i) = f(x_i, y_i) \approx \frac{y_{i+1} - y_{i-1}}{x_{i+1} - x_{i-1}} = \frac{y_{i+1} - y_{i-1}}{2h}$$

$$y_{i+1} = y_{i-1} + 2h \cdot f(x_i, y_i)$$

  - Indata $(x_{i-1}, y_{i-1}), (x_i, y_i)$
  - Noggrannhetsordning p = 2

Hur "svår" är själva differential-ekvationen? (Nivå I)

- Olika system olika känsliga för störningar – diffekvationslösning **stabil** om $\frac{\partial f}{\partial y} < 0$ för växande x

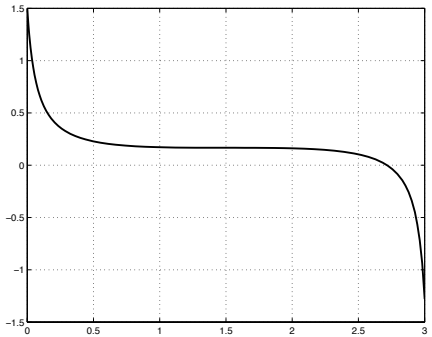


- **Styva** problem
  - Tex:
  - Term 3 << term 1, 2 när x ökar

$$y' = -100(y - \sin(x)), y(0) = 0, \text{ lösning}$$

$$y = \frac{\sin(x) - 0.01\cos(x) + 0.01e^{-100x}}{1.0001}$$

Matlabs adaptiva metoder



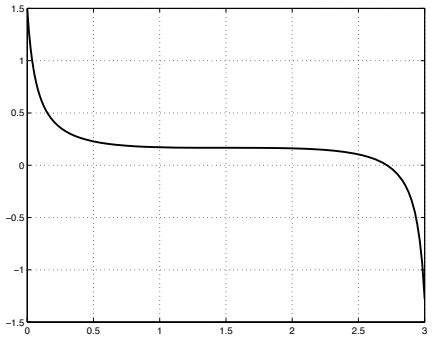
$$y' = -(3y - x/3)^2$$
$$y(0) = 1.5$$

- Vill anpassa steget h efter funktionen!

Matlabs adaptiva metoder

- ode23, ode45, ode15s, …
- Siffror ≈ noggrannhet
- ”s” ≈ specialisering för styva problem
- Adaptiv: kan inte lita på noggrannheten
  - Se FN4, integrering med quad och quadl
  - Ange lägre noggrannhet än vad som behövs, tex 10^{-6} istf 10^{-4}

Matlabs adaptiva metoder




```
f = @(x, y) -(3*y - x/3)^3;
tolerans = ...
  odeset('RelTol', 1e-4);
[x, y] = ...
 ode45(f, [0 3], 1.5, tolerans);

figure(1)
clf
plot(x, y, 'k', 'LineWidth', 2)
grid on
```


Eget arbete

- Extra datorpass on 18/3 kl 17-20
- Till nästa övning (fredag):
  - Läs GNM kap 4, 6.1-2
  - Förbered eventuella frågor om lab 4
- Till nästa föreläsning (23/3):
  - Läs GNM kap 6.3
  - Ta med GNM
- På hemsidan:
  www.csc.kth.se/DN1212/numpm09, Utdelat i menyn
  - Föreläsningsanteckningar
  - Övningstal
  - Läsanvisning till GNM
  - Labbar