Numeriska metoder, grundkurs II
Övning 6 för I2

Övningsgrupp 1

Johannes Hjorth
hjorth@nada.kth.se
Rum 163:006, Roslagstullsbacken 35
08 - 790 69 00

Kurshemsida:
`http://www.csc.kth.se/utbildning/kth/kurser/2D1240/numi07`

Material utdelat på övningarna:

`http://www.csc.kth.se/~hjorth/teaching/numi07`

Dagens program

- Högre ordningens differentialekvationer
  Omskrivning till system av första ordningen
- Randvärdesproblem
  Inskjutningsmetoden, bandmatrismetoden
- Teorital och extentor
  Inför tentan

JORGE CHAM ©THE STANFORD DAILY

Högre ordningens diff-ekvationer, exempel 7.9

Vi har en andra ordningens differentialekvation

$$\frac{d^2x}{dt^2} = -g\mu_0(1 - \alpha x^\beta)$$

Genom att göra en variabelsubstitution

$$\begin{aligned} u_1 &= x \\ u_2 &= x' \end{aligned}$$

kan vi skriva om problemet som ett system av första ordningens differentialekvationer

$$\begin{aligned} u_1' &= u_2 \\ u_2' &= -g\mu_0(1 + \alpha u_1^\beta) \end{aligned}$$

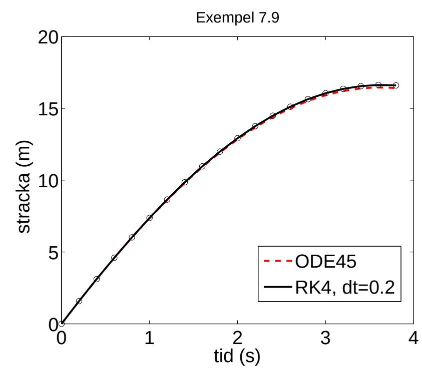
```
clear all, clf

u0 = [0; 8];

u = u0; dt = 0.2; t = 0;

while(u(2,end) > 0) % RK4 tills hastigheten är noll

  f1 = funk(u(:,end));
  f2 = funk(u(:,end) + f1*dt/2);
  f3 = funk(u(:,end) + f2*dt/2);
  f4 = funk(u(:,end) + f3*dt);

  u(:,end+1) = u(:,end) ...
           + (f1 + 2*f2 + 2*f3 + f4)*dt/6;
  t(end+1) = t(end) + dt;

end

% Vi löser även med matlabs ODE45
tol = odeset('RelTol',1e-9);
[T,U] = ode45(@funkODE, [0 t(end)], u0, tol);

Y = U(:,1); y = u(1,:);
p = plot(T,Y,'--r',t,y,'-k',t,y,'ok');

title('Exempel 7.9')
xlabel('tid (s)'), ylabel('sträcka (m)')
legend(p, 'ODE45', 'RK4, dt=0.2')
```

Funktion för beräkning av derivatan

```
function dudt = funk(u)

my0 = 0.1; alpha = 0.15; beta = 0.9; g = 9.81;

dudt = [u(2); - my0*g*(1+alpha*u(1)^beta)];
```

Matlabs ode45 kräver t som inparameter, så vi är diplomatiska och lurar den!

```
function dudt = funkODE(t,u)

dudt = funk(u);  % Varför göra extraarbete? ;)
```



Inskjutningsmetoden, exempel 7.15

Vi har fått diff ekvationen

$$\frac{d^2y}{dx^2} + \frac{2}{x}\frac{dy}{dx} = g(y)$$

där

$$g(y) = y \cdot e^{\frac{4(1-y)}{1+0.2(1-y)}}$$

samt randvärdena $\frac{dy}{dx}(0) = 0$ och $y(1) = 1$.

Vi tänker använda inskjutningsmetoden, och behöver då en funktion för att evaluera derivatan.

```
function f = funkODE(x,u)

y = u(1); g = y*exp(4*(1-y)/(1+0.2*(1-y)));

if(x == 0)
  f = [u(2); g/3];
else
  f = [u(2); g - 2/x*u(2)];
end
```

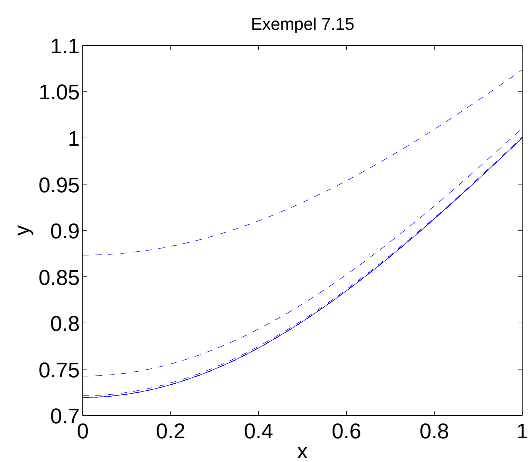
Observera att $x = 0$ måste specialbehandlas.

```
clear all, close all, format compact

s(1) = 0; s(2) = 1; tol = 0.5e-7;

[t,y] = ode45(@funkODE,[0 1], [s(1) 0]);
f(1) = y(end,1)-1;

[t,y] = ode45(@funkODE,[0 1], [s(2) 0]);
f(2) = y(end,1)-1;

% Vi använder sekantmetoden

while(abs(f(end)-f(end-1)) > tol)
  ds = - f(end)*(s(end) - s(end-1))/(f(end) - f(end-1));
  s(end+1) = s(end) + ds;

  [t,y] = ode45(@funkODE,[0 1], [s(end) 0]);
  f(end+1) = y(end,1)-1;

  plot(t,y(:,1),'--'), hold on
end

plot(t,y(:,1))
title('Exempel 7.15'), xlabel('x'), ylabel('y')

eta = 3*y(end,2)
```

Kör vi koden ser vi hur sekantmetoden styr in lösningen mot den korrekta.

```
>> tal715
eta =
    1.3550
>>
```

Här behövs sex iterationer för att nå den önskade noggrannheten för randvillkoret $y(1) = 1$.



Bandmatrismetoden, exempel 7.16

Vi har fått ett *randvärdesproblem* som ska lösas

$$y'' - (2 - 2x + x^2)y' + y = 0$$

där $y(1) = 0$ och $y(2) = 1$.

Följande derivataskattningar kommer till användning när vi diskretiserar problemet ovan

$$\begin{aligned} y'(x_n) &= \frac{y_{n+1} - y_{n-1}}{2h} \\ y''(x_n) &= \frac{y_{n+1} - 2y_n + y_{n-1}}{h^2} \end{aligned}$$

Om vi sätter in dessa skattningar ovan får vi

$$\frac{y_{n+1} - 2y_n + y_{n-1}}{h^2} - (2 - 2x_n + x_n^2)\frac{y_{n+1} - y_{n-1}}{2h} + y_n = 0$$

Nästa steg är att gruppera termerna

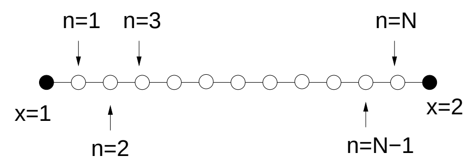
$$y_{n-1}\left(\frac{1}{h^2} + \frac{2 - 2x_n + x_n^2}{2h}\right) + y_n\left(1 - \frac{2}{h^2}\right) + y_{n+1}\left(\frac{1}{h^2} - \frac{2 - 2x_n + x_n^2}{2h}\right) = 0$$

Ekvationen ovan gäller för alla inre punkter, men för randpunkterna har vi till vänster

$$0 \cdot \left(\frac{1}{h^2} + \frac{2 - 2x_1 + x_1^2}{2h}\right) + y_1\left(1 - \frac{2}{h^2}\right) + y_2\left(\frac{1}{h^2} - \frac{2 - 2x_1 + x_1^2}{2h}\right) = 0$$

och till höger

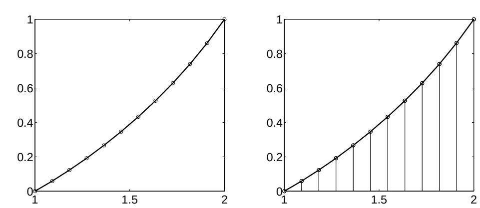
$$y_{N-1}\left(\frac{1}{h^2} + \frac{2 - 2x_N + x_N^2}{2h}\right) + y_N\left(1 - \frac{2}{h^2}\right) + y_{N+1}\left(\frac{1}{h^2} - \frac{2 - 2x_N + x_N^2}{2h}\right) = 0$$



Sen använder vi matlab och glesa matriser för att lösa det tridiagonala systemet.

```
clear all, close all, format compact

N = 10;

X = linspace(1,2,N+2)';
h = X(2)-X(1); % Fråga: Vad blir steglängden?
x = X(2:end-1);

g = 2 - 2*x + x.^2;

sup = [0; 1/h^2 - g(1:end-1)/(2*h)];
dia = (1-2/h^2)*ones(size(x));
sub = [1/h^2 + g(2:end)/(2*h); 0];

A = spdiags([sub dia sup], -1:1,N,N);
b = sparse(N,1,-(1/h^2-g(end)/(2*h)));

y = A\b;
Y = [0; y; 1];

T = h*(sum(Y) - 0.5*(Y(1) + Y(end)))

subplot(1,2,1)
plot(X,Y,'ok',X,Y,'-k')
subplot(1,2,2)
plot(X,Y,'ok',X,Y,'-k'), hold on
stem(X,Y,'-k')
```

```
>> tal716
T =
    0.4250
```



Från spdiags hjälp:

```
A = SPDIAGS(B,d,m,n) creates an m-by-n sparse matrix from the
    columns of B and places them along the diagonals specified by d.

Some elements of B, corresponding to positions "outside" of A, are
not actually used.  They are not referenced when B is an input and
are set to zero when B is an output.  If a column of B is longer than
the diagonal it's representing, elements of super-diagonals of A
correspond to the lower part of the column of B, while elements of
sub-diagonals of A correspond to the upper part of the column of B.
```

Förresten, lite smygreklam...


Hur blir zebror randinga och leoparder prickiga?

Vi kan modellera hur självorganisation i biologiska system går till med diff-ekvationer.

Den som är intresserad hänvisas till kursen Modellering av cellbiologiska processer (2D1436).

När ni trodde det var slut på reklam...

Kan en dator lära sig backa med släp?

Nyckelordet är *lära sig*.

Med genetiska algoritmer kan man efterlikna evolutionen, de bästa programmen korsas med varandra, och de dåliga kandidaterna tas bort.

Den här och andra metoder kan till och med hitta lösningar en människa kanske inte skulle tänkt på.

Den nyfikna hänvisas till Maskininlärning (2D1431).