Laboration 1, M0039M, VT16

Ove Edlund, Staffan Lundberg
LTU

1 Förberedelser

(1) Gör dig bekant med Matlab-manualen– finns för nedladdning på Fronter.

(2) Läs igenom laborationens teoridel, avsnitt 2 nedan. Kör teoridelens exempel.

2 Teoridel

En ordinär differentialekvation (ODE) av första ordningen med begynnelsevillkor, ett s.k. begynnelsevärdesproblem (BVP), kan skrivas

$$y' = f(x, y), \quad y(x_0) = y_0. \tag{1}$$

Anmärkning Lägg märke till att $y = y(x)$.

Exempel

$$y' = xy, \quad y(0) = 1.$$

Här är $f(x, y) = xy$.

2.1 Eulers metod

Begynnelsevärdesproblem av ordning 1 kan approximeras med Eulers metod, som vi sett under kursens gång. Ett verktyg som Matlab gör beräkningsgången otroligt mycket enklare.

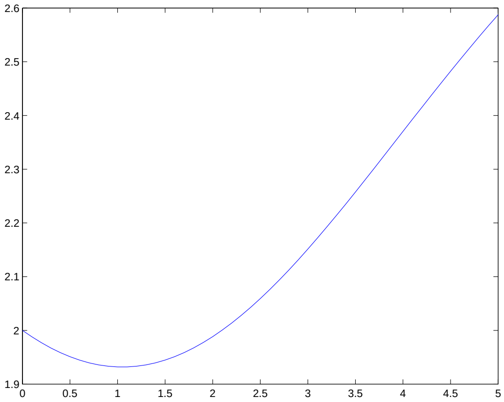
Exempel Lös begynnelsevärdesproblemet

$$y' = 0.12(x - 1)(3 - y), \quad y(0) = 2,$$

i intervallet $0 \leq x \leq 5$. Plotta lösningskurvan.

Detta löses enklast med ett Matlab-script, dvs en räcka kommandon som man effektivitetsskäl sätter i en fil istället för att knappra in dem i kommandofönstret. Steglängden h kan här lätt varieras. Notera hur Eulersteget $y_{k+1} = y_k + h\, f(x_k, y_k)$ skrivs i for-snurran.

```
% eulerscript.m
h = 0.1;
x = 0:h:5;
y(1) = 2; % motsvarar y(0) = 2, 1 ar ett index
          % tillhorande x-varde finns i x(1) vilket ar 0
for k = 1:length(x)-1
   y(k+1) = y(k) + h*0.12*(x(k)-1)*(3-y(k));
end
plot(x,y)
```



2.2 Numerisk approximation av första ordningens BVP med ode45

Eulers metod är dock inte den bästa metoden att göra numerisk approximation av lösningen. Ett bättre alternativ finns i Matlabs funktion `ode45`.

Anmärkning För numerisk lösning av (1) skrivs problemet på s.k. standardform: Vi ska lösa

$$y' = f(x, y) \quad \text{i intervallet } x_0 \le x \le x_1, \text{ där } \quad y = y_0 \quad \text{för } x = x_0. \tag{2}$$

Exempel Lös begynnelsevärdesproblemet

$$y' = 0.07(1 - y)(3 - y), \quad y(0) = 0,$$

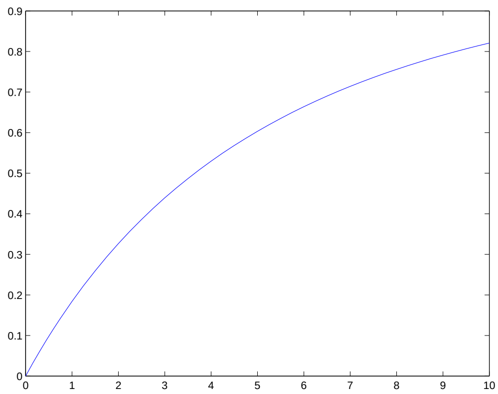
i intervallet $0 \le x \le 10$. Plotta lösningskurvan.

Vi måste definiera funktionen $f(x, y)$ i standardformen (2) i form av en M-fil:

```
% funktion.m
function dy = funktion(x, y);
%%%
% Observera att bade den oberoende variabeln, x,
% och den beroende variabeln, y, maste finnas med som indata till funktionsnamnet.
%%%
dy=0.07*(1-y).*(3-y);
end
```

I Matlabs kommandofönster ger vi kommandona

```
>> [x,y]=ode45(@funktion, [0,10],0);
%%%
% Forklaring av indataparametrarna i ode45(@funktion, [0,10],0)
% [0 10] ar foreskrivet intervall
%  x0 i ekv (1) ovan ar vanster andpkt i foreskrivet intervall
% 0  ar begynnelsevardet y(0)=0, dvs y0 i ekv (1) ovan
%%%
>> plot(x,y);
```



2.3 Numerisk lösning av andra ordningens BVP

För att lösa begynnelsevärdesproblemet

$$y'' + ay' + by = f(t), \quad y(t_0) = A, \quad y'(t_0) = B,$$

måste vi skriva om ekvationen som ett system av första ordningens BVP. Vi sätter $y_1 = y$ och $y_2 = y_1'$, vilket ger systemet

$$\begin{cases} y_1' &= y_2, \quad y_1(0) = A \\ y_2' &= -ay_2 - by_1 + f(t), \quad y_2(0) = B \end{cases}$$

Exempel Lös begynnelsevärdesproblemet

$$y'' + 0.3y' + 0.4y = 0.3\sin(2t) + 0.5\cos(3t), \quad y(0) = 0, \quad y'(0) = 1$$

i intervallet $0 \leq t \leq 30$. Plotta lösningskurvan.

Efter omskrivning får vi

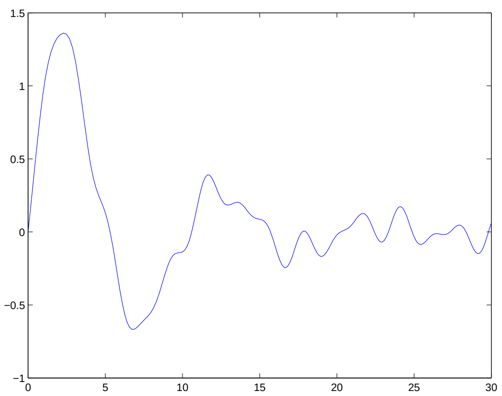
$$\begin{cases} y_1' &= y_2, \quad y_1(0) = 0 \\ y_2' &= -0.3y_2 - 0.4y_1 + 0.3\sin(2t) + 0.5\cos(3t), \quad y_2(0) = 1 \end{cases}$$

Vi måste definiera högerledet $-0.3y_2 - 0.4y_1 + 0.3\sin(2t) + 0.5\cos(3t)$ i en M-fil:

```
% vibration.m
function dy=vibration(t,y);
dy=[y(2);
    -0.3*y(2)-0.4*y(1)+0.3*sin(2*t)+0.5*cos(3*t)];
end
```

I kommandofönstret ger vi kommandona

```
>> [t,y]=ode45(@vibration, [0,30],[0,1]); %  [0,30] ar foreskrivet intervall,
                                          %  [0,1] ar begynnelsevardena
>> plot(t,y(:,1));
%%%
% Kommandot plot(t,y(:,1)) genererar en plott av t mot kol. 1 i vektorn y
%%%
```



2.4 Symbolisk lösning av ODE

Vi ska använda Matlabs Symbolic Toolbox. Kommandot `dsolve(eq,cond)` löser den ordinära differentialekvationen `eq`. Tillval: begynnelsevärdet/-värdena `cond` med avseende på variabeln `t`.

Exempel Lös differentialekvationen

$$t^2 y'' - ty' + y = t$$

symboliskt.

I kommandofönstret ger vi kommandot

```
>> dsolve('t^2*D2y-t*Dy+y=t')
 %
 % Matlab svarar
 %
ans =

(t*log(t)^2)/2 + C7*t*log(t) + C8*t
%
% C7 och C8 ar godtyckliga konstanter
```

Exempel Lös begynnelsevärdesproblemet

$$y'' = -a^2 y, \quad y(0) = 1, \quad y'(\pi/a) = 0$$

symboliskt.

I kommandofönstret ger vi kommandot

```
>> dsolve('D2y = -a^2*y', 'y(0) = 1', 'Dy(pi/a) = 0')

ans =

(1/exp(a*t*i))/2 + exp(a*t*i)/2
```

Exempel Lös begynnelsevärdesproblemet

$$y'' - 2y' + 2y = 0, \quad y(0) = 1, \quad y'(0) = 0$$

symboliskt. Plotta lösningen för $0 \leq t \leq 3$.

Matlabs kommandofönster

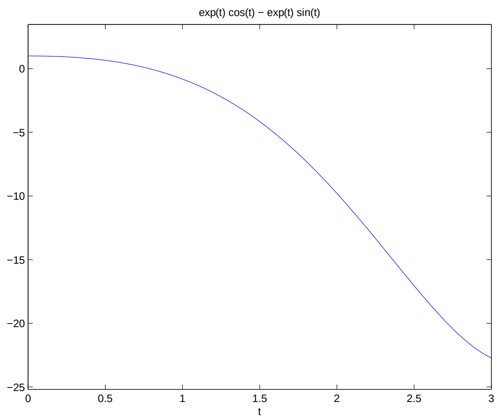
```
>> sol=dsolve('D2y-2*Dy+2*y=0','y(0)=1','Dy(0)=0')

sol =

exp(t)*cos(t) - exp(t)*sin(t)

%
%%% Symboliska uttryck plottas med ezplot
%

>> ezplot(sol,[0,3])
```



3 Uppgiftsdel

3.1 Anvisningar

- Följ anvisningarna i ”PM Laborationsmomentet i M0039M, VT2016”, som du kan ladda ner från Fronter.
- Uppgifterna (I)–(III) är obligatoriska och skall lämnas in senast 19 februari 2016.
- I uppgift (I) skall numerisk lösning med Eulers metod och `ode45` användas, medan uppgift (II) endast skall lösas med `ode45`. I uppgift (III) skall symbolisk lösning användas.

3.2 Laborationsuppgifter

(I) Lös begynnelsevärdesproblemet

$$\frac{dP}{dt} = rP\left(1 - \frac{P}{K}\right), \quad P(0) = P_0,$$

där $r = 0.5$, $K = 10$, $P_0 = 0.1$, i intervallet $0 \leq t \leq 50$, dels med Eulers metod med steglängd $h = 0.1$, dels med `ode45`. Plotta lösningarna.

(II) Lös med `ode45` begynnelsevärdesproblemet

$$y'' + 5y' + 4y = 0, \quad y(0) = 1, \quad y'(0) = 1$$

i intervallet $0 \leq t \leq 4$. Plotta lösningskurvan.

(III) Bestäm med `dsolve` lösningen till följande begynnelsevärdesproblem.

$$\frac{d^2y}{dt^2} + 2\frac{dy}{dt} = \sin(t), \quad y(0) = 1, \quad y'(0) = 0.$$

Plotta lösningen i intervallet $-2 \leq t \leq 4$.

För plottning av symboliska uttryck skall kommandot `ezplot` användas. Med kommandot `doc ezplot` får man mer information om detta plott-kommando.