CSC (Nada), KTH, Beatrice Frock

Lösningar till Tentamen 2D1240, Numeriska Metoder gk2, 2007-03-12

1. Bézier kurva.

```
clear, clf

p1=[12 0]; p2=[24 0];
k1=[1/sqrt(2) 1/sqrt(2)];

k2=[0 -1];

a1=20*sqrt(2);

a2=10;

b=p1+a1*k1;
c=p2-a2*k2;

t=(0:0.05:1)';
F3=[(1-t).^3  3*t.*(1-t).^2  3*(1-t).*t.^2  t.^3];
r=F3*[p1;  b;  c;  p2];

plot(r(:,1),r(:,2)), hold on

%plot(b(1),b(2),'*',c(1),c(2),'o')

title('flygplansvinge')

rhalv=(p1+p2)/2+3/8*(a1*k1-a2*k2);

disp('rhalv='),rhalv
```

2. Polynominterpolation.

a. Newtons ansats för interpolationspolynomet är lämplig här

$$P(x) = c_1 + c_2(x-20) + c_3(x-20)(x-22)$$

Tabellvärdena sätts in i uttrycket och ger succesivt
$c_1 = 8$,
$8 + c_2(22-20) = 10$ vilket ger $c_2 = 1$
$8 + (25-20) + c_3(25-20)(25-22) = 11$ vilket ger $c_3 = -2/15$. Polynomet är således

$$P(x) = 8 + (x-20) - \frac{2}{15}(x-20)(x-22)$$

b. Linjär interpolation ger

$$y(24) = 10 + \frac{(11-10)}{(25-22)} \cdot (24-22) = 10.667$$

.

3. Icke-linjärt ekvationssystem. Ekvationssystemet ges av $\mathbf{f}(\mathbf{x}) = 0$, där

$$\mathbf{f}(\mathbf{x}) = \begin{pmatrix} \sin(x+y) - x \\ \cos(x-y) - y \end{pmatrix}$$

Newtons metod för ett ickelinjärt system, givet en startgissning $\mathbf{x} = \mathbf{x_0}$, kan skrivas på följande sätt

$$\mathbf{x}_{n+1} = \mathbf{x}_n - J^{-1}(\mathbf{x}_n)\mathbf{f}(\mathbf{x}_n), \quad \text{för} \quad n = 1, 2, \ldots$$

J är systemets Jacobianmatris och blir i det här fallet

$$J = \begin{pmatrix} \frac{\partial f_1}{\partial x} & \frac{\partial f_1}{\partial y} \\ \frac{\partial f_2}{\partial x} & \frac{\partial f_2}{\partial y} \end{pmatrix} = \begin{pmatrix} \cos(x+y) - 1 & \cos(x+y) \\ -\sin(x-y) & \sin(x-y) - 1 \end{pmatrix}$$

Med startgissningen $x_0, y_0 = (1, 1)$ får vi $f_1 = \sin(2) - 1$, $f_2 = 0$, med lösningen i närheten.

Matlab-kod:

```
clear
disp('mata in startvektorns 2 komponenter  ')
x(1)=input('x(1) '), x(2)=input('x(2) ')

% x(1) motsvarar x och x(2) motsvarar y

x=x'; % transponera till kolumnvektor
iter=0;
hnorm=1; %initiering
while hnorm>0.5e-6 & iter<8,
   f=[sin(x(1)+x(2))-x(1)
      cos(x(1)-x(2))-x(2)];
   J=[cos(x(1)+x(2)-1     cos(x(1)+x(2))
      -sin(x(1)-x(2))     sin(x(1)-x(2))-1];
   h=-J\f;
   x=x+h;
   hnorm=norm(h,inf);
   iter=iter+1;
   disp([x'])
   disp([iter, hnorm]), % för kontroll av kvadratisk konvergens
end, %while
x, iter
```

4. **Initialvärdesproblem.**

a. Inför $u_1 = x$, $u_2 = y$, $u_3 = x'$ och $u_4 = y'$ så erhålls ett system av fyra kopplade första ordningens differentialekvationer.

Funktionsfil som beskriver systemet:

```
% Beatrice, Foucaults pendel
function dudt=fpendel(t,u);
global omega psi k
dudt=[u(3)
      u(4)
      2*omega*sin(psi)*u(4)-k^2*u(1)
      -2*omega*cos(psi)*u(3)-k^2*u(2)];
```

b. Olika alternativ:

```
% Beatrice, Numme gk2
%%%%%%%%%%%%%%%%%%%%%%%%
clear, clf
global omega psi k
omega=7.29e-5; psi=pi/4; g=9.8; l=20; k=sqrt(g/l);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% 4:e ordningens Runge-Kuttas metod

subplot(3,2,1)
disp('======================================')
disp(' RK4:')
disp('======================================')
tslut=300;
u0=[1 0 0 0]';   % initiala villkoren
t=0; tut=[0]; uut=[u0];
n=3e3; h=300/n; u=u0;
while t<tslut-h/2,
      k1=fpendel(t,u);
      k2=fpendel(t+h/2,u+h*k1/2);
      k3=fpendel(t+h/2,u+h*k2/2);
      k4=fpendel(t+h,u+h*k3);
      u=u+h/6*(k1+2*k2+2*k3+k4);
      t=t+h;
      tut=[tut t]; uut=[uut u];
end, % while
plot(uut(1,:), uut(2,:))
title('RK4')
xlabel('x'), ylabel('y')
gtext(['h=' num2str(h)])
subplot(3,2,2)
```

```
plot(tut, uut(1,:))
xlabel('t'), ylabel('x')
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Framåt Eulers metod

subplot(3,2,3)
disp('=======================================')
disp(' Euler:')
disp('=======================================')
n=6e3; h=300/n;
      u0=[1 0 0 0]';      % i.v.
tut2=[0]; uut2=[u0];
t=0; u=u0;
while t<tslut-h/2,
      u=u+h*fpendel(t,u);
      t=t+h;
      tut2=[tut2 t]; uut2=[uut2 u];
end, % while
plot(uut2(1,:), uut2(2,:))
title('Framåt Euler')
xlabel('x'), ylabel('y')
gtext(['h=' num2str(h)])
subplot(3,2,4)
plot(tut2, uut2(1,:))
title('Instabilitet i framåt Euler')
xlabel('t'), ylabel('x')
% gtext(['x\prime']),%  gtext(['y\prime'])
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Lösning med ode45
u0=[1 0 0 0]';
options=odeset('RelTol', 1e-7);
[tout, uout]=ode45('fpendel', [0 300], u0, options);
subplot(3,2,5)
plot(uout(:,1), uout(:,2))
title('Lösning med ode45')
xlabel('x'), ylabel('y')

subplot(3,2,6)
plot(tout, uout(:,1))
xlabel('t'), ylabel('x')
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% utskrift av slutvärdena

disp('slutvärdena av x, y, xprim, yprim, beräknade med ode45')
disp([uout(end,1), uout(end,2), uout(end,3), uout(end,4)])

% utskrift av xbis och ybis
```

```
disp('slutvärdena av xbis resp ybis, beräknade med ode45'),
xbis=2*omega*sin(psi)*uout(end,4)-k^2*uout(end,1)
ybis=-2*omega*cos(psi)*uout(end,3)-k^2*uout(end,2)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```