Modsim

ØV 4

Sigurd Hellesvik
sigahe

Vår 19

MTTK

Modsim 4

1a) Relative tolerance to stabilize

1 round: $1e^{-3}$

2 rounds: $1e^{-5}$

3 rounds: $1e^{-8}$

---

b) ~~0,1~~ 0,0005

is a step length that gives accurate solutions for several rounds.

---

c) If RK45 misses on a fast diff in the graph, it will be off forever. RK5 will however be right all the time, IF the step-length is sufficient for the extremes. It will be slower though.

Modsim 4

2 a) $b_1 + b_2 = 1$ $\quad b_2 c_2 = \frac{1}{2}$

$a_{21} = c_2$

$a_{21} = t$

$\Rightarrow c_2 = t$

$\Rightarrow b_2 = \dfrac{1}{2t}$

$\Rightarrow b_1 = 1 - \dfrac{1}{2t}$

Modsim4

2b)

0		
c_2	a_{21}	
	b_1	b_2

$$y_{n+1} = R y_n \qquad \dot{y} = \lambda y$$

$$k_1 = f(y_n, t) \quad , k_2 = f(y_n + h a_{21} k_1 , t_n + h c_2)$$

$$y_{n+1} = y_n + h(b_1 k_1 + b_2 k_2)$$

$$y_{n+1} = y_n + h\left(b_1 f(y_n, t) + b_2 f(y_n + h a_{21} f(y_n, t), t_n + h c_2)\right)$$

$$= y_n + h(b_1 \lambda y_n + b2 \lambda y + b_2 h a_{21} \lambda y)$$

λy

$$= y_n (1 + h\lambda b_1 + h\lambda b_2 + h^2 \lambda b_2 a_{21}) = y_n R(h\lambda)$$

$$|R(h\lambda)| \leq 1 \Rightarrow |1 + h\lambda b_1 + h\lambda b_2 + h^2 \lambda b_2 a_{21}| \leq 1$$

import from a: $$\left|1 + h\lambda - \frac{h\lambda}{2t} + \frac{h\lambda}{2t} + \frac{h^2 \lambda t}{2t}\right| \leq 1$$

$$\Rightarrow \left|1 + h\lambda + \frac{h^2 \lambda}{2}\right| \leq 1$$

t forsvant, kun polene λ og tidssteg h påvirker stabiliteten.

Modsim øv 4

3a) $\ddot{x} + g\left(1 - \left(\frac{x_d}{x}\right)^k\right) = 0$

3d) $E = \frac{mg}{k-1} \frac{x_d^k}{x^{k-1}} + mgx + \frac{1}{2}m\dot{x}^2$

$\dot{E} = mg\dot{x} - mg\left(1 - \left(\frac{x_d}{x}\right)^k\right)$

$\dot{E} = 0 \Rightarrow$ konstant energi

Svart ~~Blå~~ = Explicit Euler

Grønn = Implicit —"—

Rød = Midpoint

Scanned by CamScanner

Modsim Øv 4

4.1 a) $z = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix}$

$$\dot{z} = \begin{bmatrix} -z_3 + 4z_4^3 \\ -z_1 + 2z_2 \end{bmatrix} \qquad 0 = \begin{bmatrix} -z_1 + z_4^3 - z_2 + q_1 \\ -z_1 + z_4 + z_3 - q_2 \end{bmatrix}$$

men

$z_2 = x_1 \quad z_3 = x_2, \quad z_1 = y_1 \quad z_4 = y_2$

$u_1 = q_1 \quad u_2 = q_2$

$$\Rightarrow \quad \dot{x} = \begin{bmatrix} -x_2 + 4y_2^2 \\ 2x_1 - y_1 \end{bmatrix}$$

$$0 = \begin{bmatrix} y_2^3 - x_1 - y_1 + u_1 \\ -y_1 + x_2 - y_2 - u_2 \end{bmatrix} = g$$

b) $\frac{dg}{dt} = \begin{bmatrix} 3\dot{y}_2 - \dot{x}_1 - \dot{y}_1 \\ -\dot{y}_1 + \dot{x}_2 - \dot{y}_2 \end{bmatrix} \Rightarrow$ index = 1

Scanned by CamScanner

4.2 a) $z_2' = q_1 - z_1$

$z_3' = q_2 - (1+a) z_2 - at(q_1 - z_1)$

$q_3 = at z_2 + z_3$

$z_2 = x_1 \quad z_3 = x_2 \quad z_1 = y_1 \quad z_4 = y_2$

$q_1 = u_1 \quad q_2 = u_2 \quad q_3 = u_3$

$$\dot{x} = \begin{bmatrix} -y_1 + u_1 \\ -(1+a)x_1 + at y_1 + u_2 - at u_1 \end{bmatrix}$$

$0 = at x_1 + x_2 - u_3 = g$

---

b) $\dot{g} = at\dot{x}_1 + \dot{x}_2 = at y_1 + at u_1$

$-(1+a)x_1 + at y_1 + u_2 - at u_1$

$\ddot{g} = -at\dot{y}_1 - (1+a)\dot{x} + at\dot{y}_1 \Rightarrow \text{index} = 2$

4.3 a) $\dot{q} = v - G^T \eta$

$$M\dot{v} = Fq - G^T \lambda$$

$$0 = Gv, \quad r = Gq$$

$q = x_1$, $v = x_2$, $\eta = y_1$, $\lambda = y_2$,
$r = y_3$

$\frac{1}{2}$

$$\dot{x} = \begin{bmatrix} x_2 - G^T y_1 \\ M^{-1}F x_1 - G^T y_2 \end{bmatrix}$$

$$0 = \begin{bmatrix} G x_2 \\ G x_1 - y_3 \end{bmatrix} = g$$

---

b) $$\dot{g} = \begin{bmatrix} G\dot{x}_2 \\ G\dot{x}_1 - \dot{y}_3 \end{bmatrix} = \begin{bmatrix} G(x_2 - G^T y_1) \\ G(M^{-1}F x_1 - G^T y_2) - \dot{y}_3 \end{bmatrix}$$

$$\ddot{g} = \begin{bmatrix} G(\dot{x}_2 - G^T \dot{y}_1) \\ G(M^{-1}F\dot{x}_1 - G^T \dot{y}_2) - \ddot{y}_3 \end{bmatrix} \Rightarrow \text{index} = 2$$

This dosent work!

4.4 a) $m_1 \ddot{x}_1 = k(x_2 - x_1 - x_0) + F$

$$m_2 \ddot{x}_2 = -k(x_2 - x_1 - x_0)$$

$$x_2 = r$$

$a_1 = x_1 \qquad \mathring{a}_1 = x_2$

$a_2 = \dot{x}_1 \qquad \dot{\mathring{a}}_2 = \dot{x}_2$

$$\dot{a}_2 = \frac{k}{m_1}(\mathring{a}_1 - a_1 - x_0) + F$$

$$\dot{\mathring{a}}_2 = \frac{k}{m_2}(\mathring{a}_1 - a_1 - x_0)$$

$$\dot{a}_1 = a_2, \quad \dot{\mathring{a}}_1 = \mathring{a}_2$$

$$\mathring{a}_1 = r$$

$\Rightarrow a_2 = x_1, \ \mathring{a}_2 = x_2, \ a_1 = x_3, \ \mathring{a}_1 = x_4$

$x_0 = u_1 \qquad F = y_1$

$r = u_2$

$$\Rightarrow \dot{x} = \begin{bmatrix} \frac{k}{m_1}(x_4 - x_3 - u_1) + y_1 \\ \frac{k}{m_2}(x_4 - x_3 - u_1) \\ x_1 \\ x_2 \\ u_2 \end{bmatrix}$$

$$0 = r - x_4 = g$$

4.4b) $\dot{g} = -\dot{x}_4 = x_2$

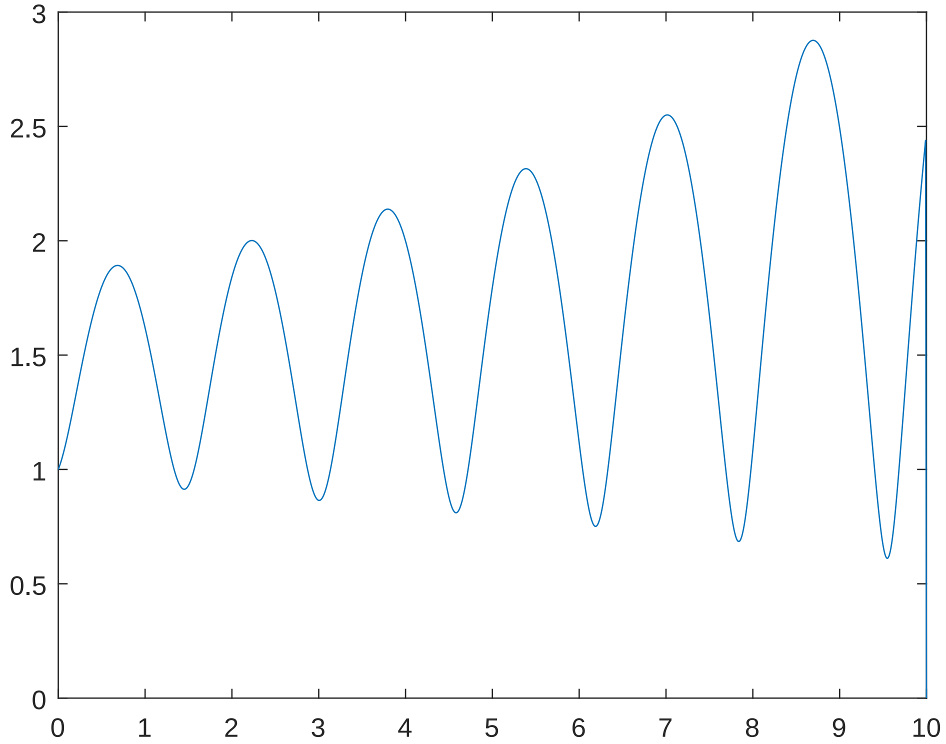
$$\ddot{g} = \dot{x}_2 = \frac{k}{m_2}(x_4 - x_3 - u_1)$$

$$\dddot{g} = \frac{k}{m_2}(\dot{x}_4 - \dot{x}_3) = \frac{k}{m_2}(x_2 - x_1)$$

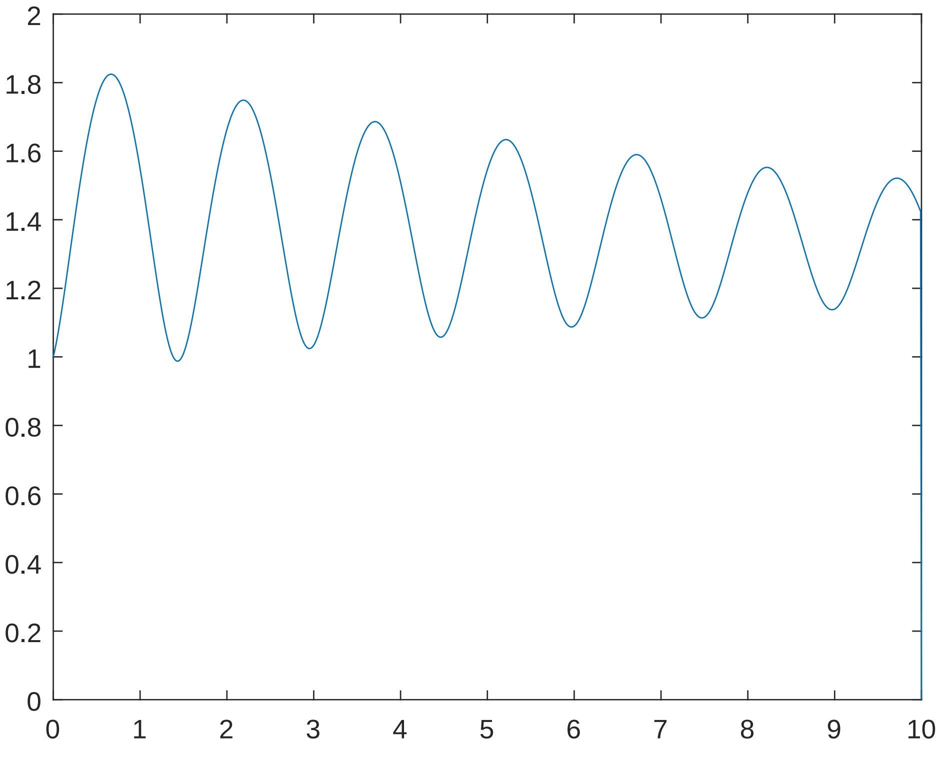
$$g^{(4)} = \frac{k}{m_2}(\dot{x}_2 - \dot{x}_1) = \frac{k}{m_2}\left(\frac{k}{m_2}(x_4 - x_3 - u_1) - \frac{k}{m_1}(x_4 - x_3 - u_1) + y_1\right)$$

$$g^{(5)} = \frac{k}{m_2}\left(\frac{k}{m_2}(\dot{x}_4 - \dot{x}_3) - \frac{k}{m_1}(\dot{x}_4 - \dot{x}_3) + \dot{y}_1\right)$$

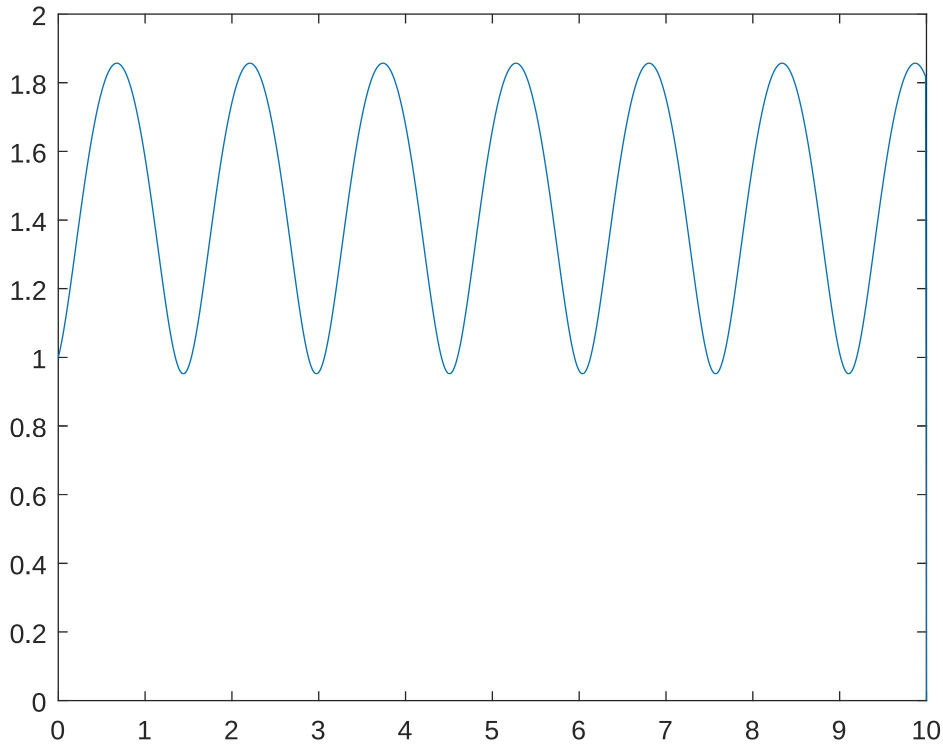
$\Rightarrow$ <u>index = 5</u>



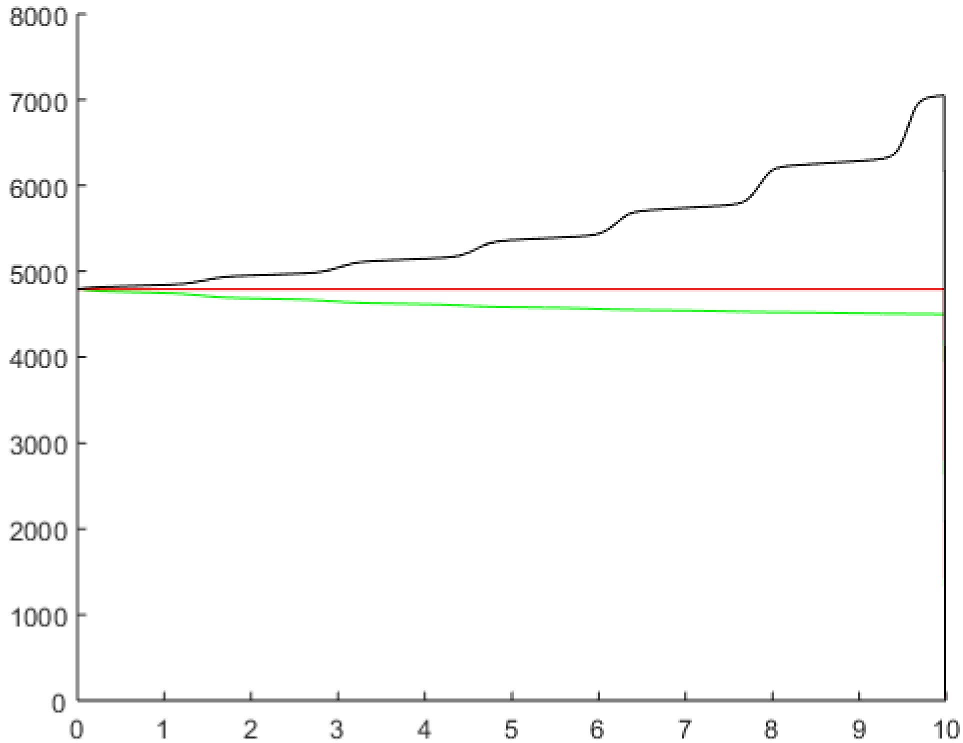
0
0.5
1
1.5
2
2.5
3
0
1
2
3
4
5
6
7
8
9
10



2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
0
1
2
3
4
5
6
7
8
9
10



2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
0
1
2
3
4
5
6
7
8
9
10



8000
7000
6000
5000
4000
3000
2000
1000
0
0 1 2 3 4 5 6 7 8 9 10

Editor - oppg3a.m

Variables - y

oppg3b.m | oppg3c.m | oppg3a.m | +

This file can be opened as a Live Script. For more information, see Creating Live Scripts.

```
%% Implicit Euler
%y(n+1)=y(n)+h(f(n),t(n))+h^2*f(y(n),t(n))/dt+O(h^3)
 x_d = 1.32;
 K = 2.40;
 g = 9.81;
    m=200;

t=10; h=0.01;
N = round(t/h,0);
time=0:h:t
y0 = [2;0];
sigma = size(y0) + 1;

f = @(y,t) [ y(2); -g*(1-(x_d/y(1))^K) ];

y = zeros(size(y0,1),size(time,2));
opt = optimset('Display','off','TolFun',1e-8);

E=zeros(size(y,2));


y(:,1)=1;

for i = 1:N-1

     y(:,1+i) = (y(:,i) + h*feval(f, y(:,i), time(i)));
     E(i)=(m*g/(K-1)) *(x_d^K/(y(1,i)^(K-1)))+m*g*y(1,i)+1/2*m*y(2,i)^2;
end


%plot(time,y(1,:))

plot(time,E,'black')
```

```
%% Implicit Euler
%y(n+1)=y(n)+h(f(n),t(n))+h^2*f(y(n),t(n))/dt+O(h^3)
 x_d = 1.32;
 K = 2.40;
 g = 9.81;

t=10; h=0.01;
N = round(t/h,0);
time=0:h:t
y0 = [2;0];
sigma = size(y0) + 1;

f = @(y,t) [ y(2); -g*(1-(x_d/y(1))^K) ];

y = zeros(size(y0,1),size(time,2));
opt = optimset('Display','off','TolFun',1e-8);



y(:,1)=1;


for i = 1:N-1
     %y(:,1+i) = (y(:,i) + h*feval(f, y(:,i), time(i)));
    r = @(ynext) (y(:,i) + h*feval(f, ynext, time(i+1)) - ynext);
    y(:,i+1) = fsolve(r, y(:,i), opt);
    E(i)=(m*g/(K-1)) *(x_d^K/(y(1,i)^(K-1)))+m*g*y(1,i)+1/2*m*y(2,i)^2;
end

hold on;
plot(time,E,'green')
```

oppg3c.m

This file can be opened as a Live Script. For more information, see Creating Live Scripts.

```
%% Implicit Euler
%y(n+1)=y(n)+h(f(n),t(n))+h^2*f(y(n),t(n))/dt+O(h^3)
 x_d = 1.32;
 K = 2.40;
 g = 9.81;

t=10; h=0.01;
N = round(t/h,0);
time=0:h:t
y0 = [2;0];
sigma = size(y0) + 1;

f = @(y,t) [ y(2); -g*(1-(x_d/y(1))^K) ];

y = zeros(size(y0,1),size(time,2));
opt = optimset('Display','off','TolFun',1e-8);



y(:,1)=1;


for i = 1:N-1
    r = @(ynext) (y(:,i) + h*feval(f, (ynext+y(:,i))/2, time(i+1)+h/2) - ynext);
    y(:,i+1) = fsolve(r, y(:,i), opt);
      E(i)=(m*g/(K-1)) *(x_d^K/(y(1,i)^(K-1)))+m*g*y(1,i)+1/2*m*y(2,i)^2;
end

hold on;
plot(time,E,'red')
%plot(time,y(1,:))

```