6 Modelovanie a simulácia nelineárnych fyzikálnych systémov v prostredí MATLAB

6.1 Modelovanie a simulácia matematického kyvadla v prostredí MATLAB

Úloha: Zostavte matematický a simulačný model pre NDS matematického kyvadla.

Uvažujeme matematické kyvadlo zobrazené na obrázku, ktoré je tvorené závažím o hmotnosti m, zavesenom na šnúre o dĺžke l, ktorej hmotnosť môžeme zanedbať.

a) Modelovanie nelineárneho DR – matematické kyvadlo

Popis parametrov matematického kyvadla na obrázku:

m – hmotnosť závažia

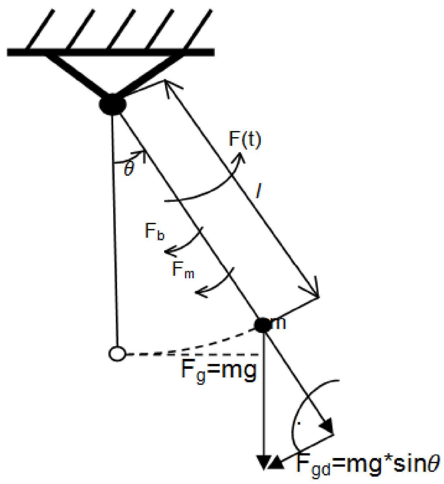
l - dĺžka závesu

g - gravitačná konštanta

B - koeficient tlmenia

Význam skúmanej fyzikálnej veličiny:

θ- uhol vychýlenia kyvadla



Význam síl v dynamickom modeli matematického kyvadla je nasledovný

$F(t)$- budiaca sila (vyvedie dynamický systém z rovnovážnej polohy)
F_m- sila zotrvačnosti
F_b- brzdná sila (odpor vzduchu + tangensiálna zložka tiažovej sily F_{gd})
F_g - tiažová sila, $F_g = m\,.\,g$

Na zostavenie matematického modelu dynamického systému matematické kyvadlo použijeme d'Alembertov princíp:

$$F_m + F_b + F_{gd} - F(t) = 0$$

kde

- $F_m = m * l * \frac{d^2\theta(t)}{dt^2} = m * l * \theta''(t)$ - výpočet sily zotrvačnosti
- $F_b = B * l * \frac{d\theta}{dt} = B * l * \theta'(t)$ - výpočet brzdiacej sily pre obvodovú rýchlosť kyvadla
- $F_{gd} = m * g * sin\theta$ - tangensiálna zložka tiažovej sily
- $F(t) = \frac{M(t)}{l}$ - externá budiaca sila
- $M(t)$ - vonkajší moment – vstup do systému
- $\theta(t)$ - uhol vychýlenia kyvadla – výstup systému

Dosadením jednotlivých síl do d'Alambertov princíp dostaneme nelineárnu diferenciálnu rovnicu 2. rádu s pravou stranou, ktorá reprezentuje matematický model kyvadla

$$m * l * \theta''(t) + B * l * \theta'(t) + m * g * sin\,\theta = \frac{M(t)}{l}$$

Po normovaní uvedenej DR dostanem NDR

$$\theta''(t) + \frac{B}{m}\theta'(t) + \frac{g}{l}\sin\theta = \frac{1}{ml^2}M(t)$$

Prepis do substitučného kanonického tvaru

Substitúcia: $x_1(t) = \theta(t)$,

$$x_1'(t) = x_2(t)$$

$$x_2'(t) = -\frac{B}{m}x_2(t) - \frac{q}{l} * \sin\big(x_1(t)\big) + \frac{1}{ml^2}M(t)$$

b) Simuláciou v MATLAB-e zistite časový priebeh výchylky kyvadla *θ(t)* a jeho rýchlosti *θ´(t)* bez pôsobenia vstupného signálu (počiatočné podmienky : *x_1(0)* = 1, *x_1´(0)* = 0)a pri pôsobení vstupného budiaceho signálu *M(t)*

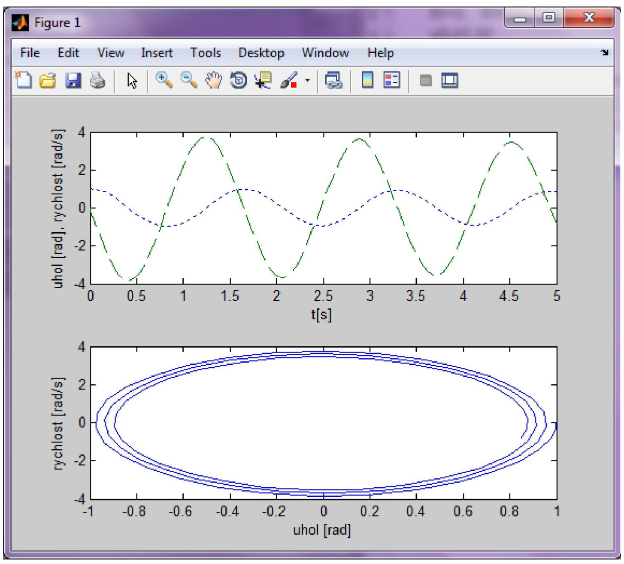
<u>Riešenie v programovom prostredí MATLAB:</u>

funkcia : **kyvadlo.m**

```
% funkcia 'kyvadlo.m' - kmitanie vychyleneho kyvadla s tlmenim

function xdot=kyvadlo(t,x)
l=0.6; B=1e-2;g=9.81;m=0.2038;
xdot=[x(2);-B/m*x(2)-g/l*sin(x(1))];
```

hlavný program : **kyv23sol.m**

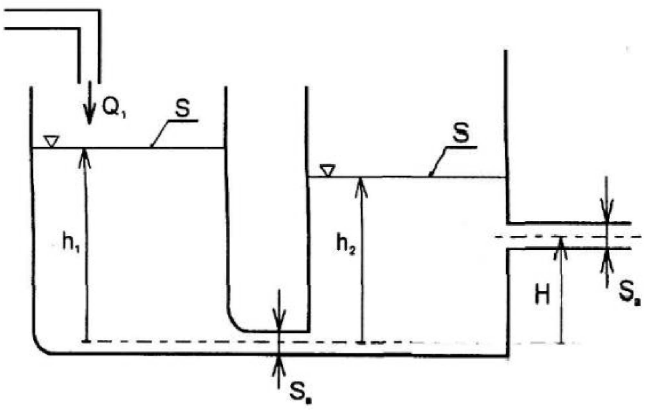
```
% subor 'kyv23sol.m' na riesenie DR kyvadla
t0=0;   tfin=5;
x0=[1 0]';
tol=1e-3; trace=0;
[t,x]=ode45('kyvadlo',t0,tfin,x0,tol,trace);
subplot(211), plot(t,x(:,1),':',t,x(:,2),'--');
xlabel('t[s]'), ylabel('uhol [rad], rychlost [rad/s]');
subplot(212), plot(x(:,1),x(:,2))
title ...
```



Obrázok 6-1 Simulácia pohybu matematického kyvadla – časový priebeh výchylky kyvadla *θ(t)* , *rýchlosti pohybu θ´(t) pri M(t) = 0 a PP = [1,0]*

6.2 Modelovanie a simulácia hydraulického systému v jazyku MATLAB

Úloha:
Zostavte matematický a simulačný model hydraulického systému dvoch nádob v interakcii.



a) Zostavenie matematického modelu hydraulického systému

Sledovanou veličinou hydraulického systému bude zmena výšky hladiny kvapaliny v prvej nádobe h_1(t) od prítoku Q_1(t)do prvej nádoby.
Zmenu výšky hladiny vieme popísať pomocou DR v tvare

$$S * \frac{dh_1(t)}{dt} = Q_1 - S_0 * v_1$$

Pre výtokovú rýchlosť kvapaliny z nádoby platí Toricelliho vzťah $v = \sqrt{2g\left(h_1(t) - h_2(t)\right)}$.

Teda zmena výšky hladiny vody $h_1(t)$ môže byť vyjadrená DR v prvej nádobe bude popísaná diferenciálnou rovnicou

$$h_1'(t) = \frac{1}{S} * Q - \frac{S_0}{S}\sqrt{2g\left(h_1(t) - h_2(t)\right)}.$$

Podobným spôsobom vieme popísať aj zmenu výšky hladiny vody $h_2(t)$ v druhej nádobe:

$$h_2'(t) = \frac{S_0}{S}\sqrt{2g(h_1(t) - h_2(t))} - \frac{S_0}{S}\sqrt{2g(h_2(t) - H)}$$

b) Úloha na samostatné riešenie: Naprogramujte simulačný model hydraulického systému pre získanie časových priebehov výšky hladiny kvapaliny $h_1(t)$ a $h_2(t)$ pri zvolených počiatočných podmienkach a vstupnom prítoku $Q_1(t)$

*Simuláciu vykonajte pre zvolených počiatočných podmienkach:*_ $h_1(0) = 1{,}2$ a $h_2(0) = 0{,}7$.
a parametroch:

$S = 0{,}25m^2$ $S_0 = 0{,}01m^2$ $H = 0{,}2m$

<u>*Pri vstupnom prítoku :*</u>

$$Q = 0{,}025m^3s^{-1}$$

6.3 Príklad na simuláciu nelineárneho dynamického systému van der Polov oscilátor

Zostavte simulačný model nelineárneho dynamického systému van der Polovho oscilátora v jazyku MATLAB na základe matematického modelu, ktorý je vyjadrený nelineárnou DR:

$$y''(t) + A\left(1 - y^2(t)\right)y'(t) - y(t) = u(t)$$

Nelineárna diferenciálna rovnica je druhého rádu potrebujeme,kvôli vytvoreniu simulačného modelu zostaviťsystém NDR prvého rádu a to prepisom NDR oscilátora do substitučného kanonického tvaru.

Prepis do substitučného kanonického tvaru:

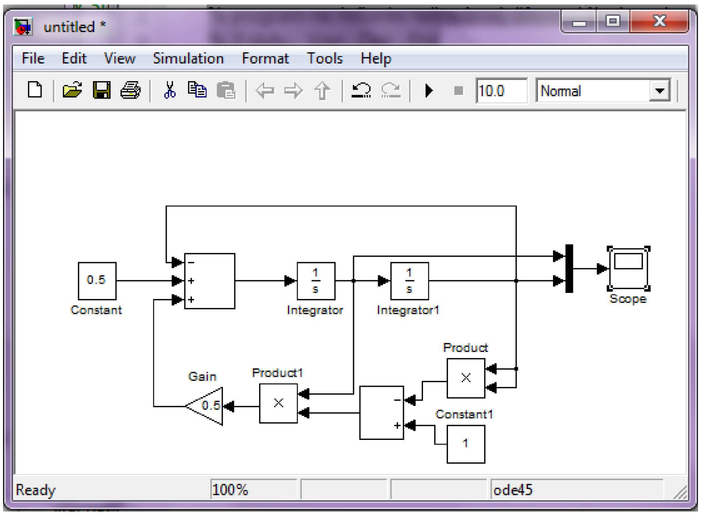
Substitúcia: $y(t) = x_1(t)$

$$x_1'(t) = x_2(t)$$

$$x_2'(t) = -A * \left(1 - x_1^2(t)\right)x_2(t) + x_1(t) + u(t)$$

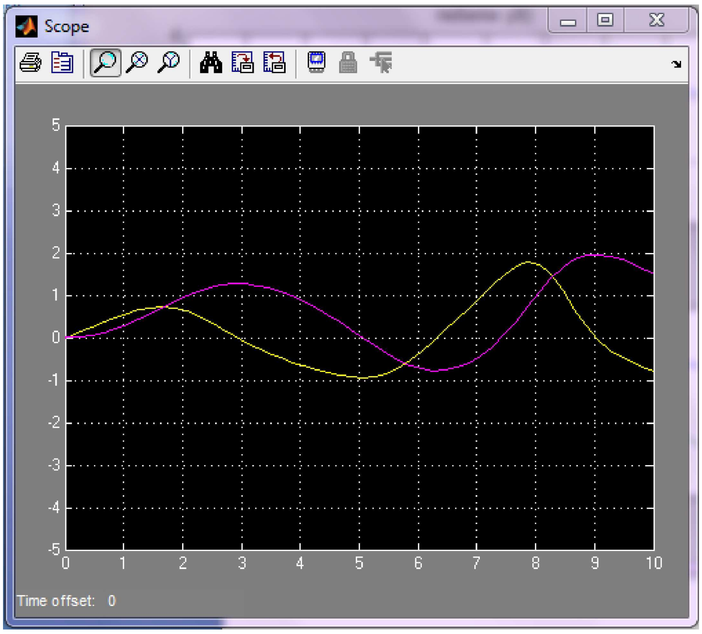
<u>Zostavenie simulačného modelu v programovom prostredí Simulink :</u>

Simulačný model van der-Polovho oscilátora zostavíme v programovom prostredí Simulink, metódou postupného znižovania radu derivácie.
Pre simuláciu modelu použijeme voľbu parametra $A = 0{,}5$ a vstupný budiaci signál $u(t) = 0{,}5$.



Obrázok 6-2 Programová Schéma simulačného modelu – Vander –Polov oscilátor

Pred spustením simulácie užívateľ musí zvoliť metódu pre riešenie NDR (solver: napr. metóda R-K), je nutné zvoliť integračný krok pre výpočet, počiatočné podmienky a začiatočný a koncový čas pre riešenie simulácie.



Obrázok 6-3 Grafické znazornenie riešenia $x_1(t)$ a $x_2(t)$ pre NDR Vander-Polov oscilatora

6.4 Príklad na simuláciu nelineárneho dynamického systému “dravec-korist“

Úloha:
Zostavte simulačný model v prostredí MATLAB pre systém “dravec-korist” (systém Lotka-Volterra) s cieľom študovať efekt interakcie koeficientov α,β , ktorého matematický model je tvorený dvoma NDR prvého rádu

$$y_1'(t) = y_1(t) - \alpha . y_1(t) . y_2(t)$$

$$y_2'(t) = -y_2(t) + \beta . y_1(t) . y_2(t)$$

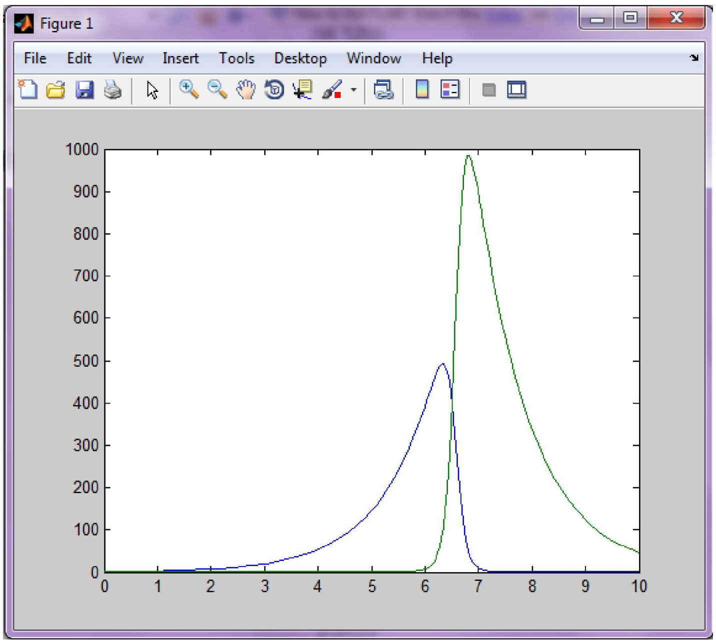
<u>Simulačný model systému Lotka-Volterra v jazyku MATLAB</u>

funkcia: **lotka.m**

```
% funkcia 'lotka.m' - interakcia koeficientov alpha a beta
function yp=lotka(t,y)
global ALPHA BETA
yp = [y(1)-ALPHA*y(1)*y(2); -y(2)+BETA*y(1)*y(2)]
```

hlavná funkcia

```
global ALPHA BETA
ALPHA = 0.01
BETA = 0.02
[t,y]=ode23('lotka',0,10,[1,1]);
plot(t,y)
```



Obrázok 6-4 Časové priebehy $y_1(t)$ a $y_2(t)$ simulačného modelu Lotka-Volterra pri α = 0,01 , β = 0,02

6.5 Zadanie č.3: Riešenie nelineárnej diferenciálnej rovnice v prostredí MATLAB s využitím naprogramovanej metódy Runge-Kutta 4. rádu

ZADANIE:

Riešenie nelineárnej diferenciálnej rovnice (NDR) numericky so zvolenou numerickou technikou a algoritmicky v programovom prostredí MATLAB.

OBSAH ZADANIA:
Zvolenú nelineárnu diferenciálnu rovnicu s konštantnými koeficientmi a s počiatočnými podmienkami riešte v programovom prostredí MATLAB.

Majme nelineárnu diferenciálnu rovnicu 2. rádu tvaru:

$$5y''(t) + 4cos(y'(t))y(t) - \sqrt{y(t)} = sin(2\pi)$$

Prepíšeme do substitučného kanonického tvaru pri substitúcii: $y(t) = x_1(t)$:

$$x_1' = x_2$$

$$x_2' = \frac{sin(2\pi) - 4cos(x_2)x_1 - \sqrt{x_1}}{5}$$

Riešenie nelineárnej diferenciálnej rovnice v programovom prostredí MATLAB s využitím funkcie ode45 a vlastnej funkcie R-K:

funkcia **NDR.m**

```
function xder=NDR(t,x)

xder=[x(2);(sin(2*pi)-4*cos(x(2))*x(1)-sqrt(x(1)))/5]

return
```

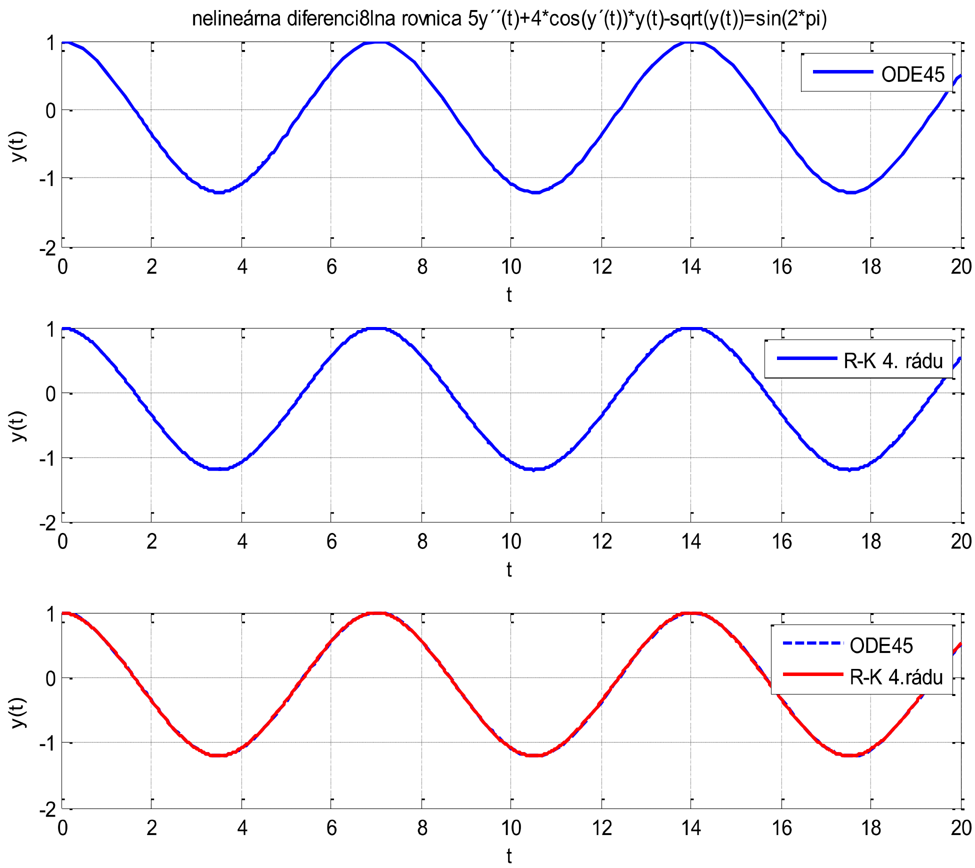
Časť kódu v jazyku MATLAB pre vlastnú funkciu Runge-Kutta 4.rádu

funkcia **rk.m**

```
function [t,y]=rk(f,T,PP)

⋮

for i=1:length(t)
    K1=h.*f(x(i),y(i,:));
    K2=h.*f(x(i)+h./2,y(i,:)+K1'./2);
    K3=h.*f(x(i)+h./2,y(i,:)+K2'./2);
    K4=h.*f(x(i)+h,y(i,:)+K3');

if i~=(length(t))
        y(i+1,:)=y(i,:)+(K1'+2*(K2'+K3')+K4')/6;
end;
end;
```

Časť kódu hlavného programu

```
% hlavnýprogram pre riešenie nelineárnej diferenciálnej rovnice pomocou funkcie ODE45 a vlastnou funkciou R-K 4. rádu
⋮

[t,y]=ode45('NDR',[0 20],[1 0])
[t1,y1]=rk(@NDR,[0 20], [1 0])
subplot(311)
plot(t,y(:,1))
⋮
```



Obrázok 6-5 Riešenie NDR s využitím vstavanej funkcie ode45 a vlastnej funkcie rk 4. rádu a ich vzajomné porovnanie