

**Technická univerzita v Košiciach**  
**Fakulta elektrotechniky a informatiky**  
**Katedra kybernetiky a umelej inteligencie**

**Simulačné systémy**  
(Zadanie č. 4)

### Text zadania

Uvažujte fyzikálno-matematický model dynamického systému, ktorý je popísaný Lineárnou diferenciálnou rovnicou (LDR) 2. a vyššieho rádu.

Navrhňte m-file v jazyku Matlab, ktorý umožní:

1. zadefinovanie prenosovej funkcie opisujúcej LDS v s-oblasti (v polynomilálnom tvare, v tvare póly/nuly) a v stavovom priestore pomocou matíc A, B, C, D.
2. konverziu modelov zo stavového priestoru do tvaru prenosovej funkcie a naopak.
3. analýzu LTI DS v časovej (prechodová charakteristika, impulzná charakteristika, odozva na ľubovoľný vstupný signál) a frekvenčnej oblasti (Nq,Bode, Nichols charakteristiky).
4. vyhodnotenie stability uvažovaného LTI DS na základe získaných odoziev na rôzne typy vstupných signálov.

Zadanie riešte modulárne pomocou príkazov riadiaceho toku príkazov (switch, menu, if).

Zadanie má obsahovať obrázok modelu, popis fyzikálnych veličín, vstupy, výstupy modelu.

### Dodefinovanie zadania

Pre vypracovanie zadania som si vybral nasledovný model tlmiča kola auta.

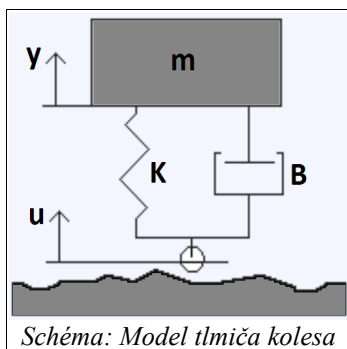


Schéma: Model tlmiča kola

Popis veličín systému:

$B$  – koeficient tlmenia.

$K$  – tuhosť pružiny.

$m$  – hmotnosť auta.

$u$  – vstupná veličina systému.

$y$  – výstupná veličina systému.

Popis modelu pomocou diferenciálnej rovnice:

$$m \cdot \ddot{y} + B \cdot \dot{y} + K \cdot y = B \cdot \dot{u} + K \cdot u$$

### Analýza úlohy

Pre riešenie v s-oblasti prepíšeme rovnicu pomocou laplaceovej transformácie:

$$ms^2 \cdot Y(s) + Bs \cdot Y(s) + K \cdot Y(s) = Bs \cdot U(s) + K \cdot U(s)$$

Rovnicu upravíme:

$$(ms^2 + Bs + K) \cdot Y(s) = (Bs + K) \cdot U(s)$$
$$\frac{Y(s)}{U(s)} = \frac{Bs + K}{ms^2 + Bs + K}$$

Určíme prenosovú funkciu:

$$F(s) = \frac{Bs + K}{ms^2 + Bs + K}$$

Môže byť zapísaná aj v tvare:

$$F(s) = \frac{B}{M} \cdot \frac{\left(s + \frac{K}{B}\right)}{\left(s + \frac{\frac{B}{m} - \sqrt{\left(\frac{B}{m}\right)^2 - 4\frac{K}{m}}}{2}\right) \left(s + \frac{\frac{B}{m} + \sqrt{\left(\frac{B}{m}\right)^2 - 4\frac{K}{m}}}{2}\right)}$$

Odkiaľ už jednoducho získame hodnoty zosilnenia, pólov aj núl.

### Implementácia riešenia

Program naprogramovaný v prostredí simulačného jazyka MATLAB je zostavený zo súborov hlavného programu model.m a funkcií casova.m, frekvencna.m, konverzia.m, odozva.m, stabilita.m a vstup.m

#### main.m

Globálne premenné: sys, a, b, c, d.

Lokálne premenné: volba.

Hlavná časť programu začne odvolaním na funkciu vstup, po ktorej začne v cykle prezentovať menu, z ktorého sa spúšťajú ostatné funkcie, resp. sa ukončuje program.

#### vstup.m

Globálne premenné: sys, a, b, c, d.

Lokálne premenné: volba, B, K, m, opakuj, num, den, poly, nuly, zosilnenie.

Užívateľ si v menu vyberie spôsob určenia vstupov do systému. Funkcia následne zabezpečí načítanie systému (v danom formáte) do premennej sys. Zadefinovanie prenosovej funkcie opisujúcej LDS môže byť v s-oblasti (v polynomilálnom tvare, v tvare póly/nuly) alebo v stavovom priestore pomocou matíc A, B, C, D.

#### casova.m

Globálne premenné: sys, a, b, c, d.

Lokálne premenné: volba.

Užívateľ si v menu vyberie, akú časovú charakteristiku chce graficky vykresliť. Má na výber prechodovú, impulznú, alebo odozvu na ďalší ľubovoľný signál (čo spustí funkciu odozva.m).

#### frekvencna.m

Globálne premenné: sys, a, b, c, d.

Lokálne premenné: volba.

Užívateľ si v menu vyberie, akú frekvenčnú charakteristiku chce graficky vykresliť. Má na výber Nyquistovu, Bodeho, alebo Nicholsovu frekvenčnú charakteristiku.

#### konverzia.m

Vstupné premenné: sys, a, b, c, d.

Lokálne premenné: volba.

Táto funkciu umožňuje zobraziť systém uložený v premennej sys (resp. v stavovom priestore v premenných a, b, c, d) aj v inom formáte, ako bol zadán vo funkcii vstup.m.

### odozva.m

Globálne premenné: sys.

Lokálne premenné: volba, num, den, signal\_type, ton, time, dt, u, t

Funkcia zabezpečuje vykreslenie časového priebehu signálu, ktorého parametre zadá užívateľ.

### stabilita.m

Globálne premenné: sys.

Lokálne premenné: num, den, r, max, test.

Funkcia na základe koreňov charakteristickej rovnice systému vyhodnotí jeho stabilitu a výsledok vypíše.

### Zdrojový kód model.m

```
% Hlavná časť programu Fyzikálno-dynamického modelu.
```

```
global sys a b c d;
```

```
sys=vstup
```

```
volba = 7;
```

```
while volba ~= 5
```

```
    volba = menu ('Hlavne menu', 'Konverzia modelov', 'Casove charakteristiky', ...  
                  'Frekvencne charakteristiky', 'Urcenie stability', 'Koniec');
```

```
    clc;
```

```
    switch volba
```

```
        case 1
```

```
            konverzia(a,b,c,d,sys);
```

```
            pause
```

```
        case 2
```

```
            casova;
```

```
        case 3
```

```
            frekvencna;
```

```
        case 4
```

```
            stabilita;
```

```
            pause
```

```
    end
```

```
    clc;
```

```
end
```

### Zdrojový kód casova.m

```
% Casova analyza
```

```
global sys a b c d;
```

```
volba = menu('Analyza v casovej oblasti', 'Prechodova charakteristika', ...  
             'Impulzna charakteristika', 'Odozva na lubovolny signal');
```

```
switch volba
```

```
    case 1
```

```
        % Prechodova char.
```

```
        step(sys)
```

```
        title('Prechodova casova charakteristika')
```

```
    case 2
```

```
        % Impulzna char.
```

```
        impulse(sys)
```

```
        title('Impulzna casova charakteristika')
```

```
    case 3
```

```
        % Odozva na lubovolny signal
```

```
        odozva;
```

```
end
```

### Zdrojový kód frekvencna.m

```
% Frekvencna analyza

global sys a b c d;

volba = menu('Analyza vo frekvencnej oblasti','Nyquistova charakteristika',...
            'Bodeho charakteristika','Nicholsova charakteristika');
switch volba
    case 1
        % Nyquistova frekvencna char.
        nyquist(sys)
        title('Nyquistova frekvencna charakteristika')
    case 2
        % Bodeho frekvencna char.
        bode(sys)
        title('Bodeho frekvencna charakteristika')
    case 3
        % Nicholsova frekvencna char.
        nichols(sys)
        title('Nicholsova frekvencna charakteristika')
end
```

### Zdrojový kód konverzia.m

```
function []=konverzia(a,b,c,d,sys)

% Konverzia modelu zo stavoveho priestoru do prechodovej funkcie a naopak

volba = menu('Konverzia','Zo SS do Prechodovej funkcie','Z prechodovej funkcie
do SS');

switch volba
    case 1
        disp('Konverzia modelu zo stavoveho priestoru do prechodovej funkcie
pomocou "ss2tf"')
        [num,den]=ss2tf(a,b,c,d,1);
        tf(num,den)
    case 2
        disp('Konverzia modelu z prechodovej funkcie do stavoveho priestoru
pomocou "tf2ss"')
        [num,den]=tfdata(sys,'v');
        [a,b,c,d]=tf2ss(num,den)
end
```

### Zdrojový kód odozva.m

```
% Odozva na lubovolny vstupny signal

global sys;

signal_type=input('Zadajte typ signalu (napr. ''sin''):\n');
ton=input('Zadajte periodu vzorkovania:\n');
time=input('Zadajte celkovu dobu simulacie:\n');
dt=input('Zadajte vzorkovanie casu:\n');

[u,t]=gensig(signal_type,ton,time,dt);
lsim(sys,u,t)
grid;
title('Odozva na lubovolny signal');
```

### Zdrojový kód stabilita.m

```
% Vyhodnotenie stability

global sys
```

```

[num,den]=tfdata(sys,'v');
r=roots(den);
max=size(r);

test = 1;

for i= 1:max(:,1)
    if r(i) >0;
        test=0;
    end
end

if test == 0
    disp('System je nestabilny');
else
    disp('System je stabilny, alebo na hranici stability')
end

```

### Zdrojový kód vstup.m

```

function [sys]=vstup()

% Funkcia pre určenie vstupov do systému.

% Zadeinovanie prenosovej funkcie opisujúcej LDS v s-oblasti
% (v polynomilálnom tvare, v tvare póly/nuly) alebo v stavovom
% priestore pomocou matíc A, B, C, D.

global sys a b c d;

B = 2 ;%koeficient tlmenia
K = 6 ;%tuhosť pružiny
m = 700 ;%hmotnosť auta

opakuj=1;
while opakuj==1
    volba = menu ('Vstupy','V polynomialnom tvare', 'Pomocou polov a nul', 'V
stavovom priestore');
    switch volba
        case 1
            B %koeficient tlmenia
            K %tuhosť pružiny
            m %hmotnosť auta
            num=[B K];
            den=[m B K];
            sys=tf(num,den);
            opakuj=0;
        case 2
            nuly = [-K/B]
            poly = [(-B/m+sqrt((B/m)^2-4*K/m))/2 (-B/m-sqrt((B/m)^2-4*K/m))/2]
            zosilnenie = B/m
            sys=zpk(nuly,poly,zosilnenie);
            opakuj=0;
        case 3
            a=[-0.0029, -0.0086; 1, 0];
            b=[1; 0];
            c=[0.0029, 0.0086];
            d=[0];
            sys=ss(a,b,c,d);
            opakuj=0;
    end
end

return

```

## Príloha: Model v Simulinku

