4 Aplikácie metód numerickej matematiky

4.1 Numerické a algoritmické riešenie určitého integrálu

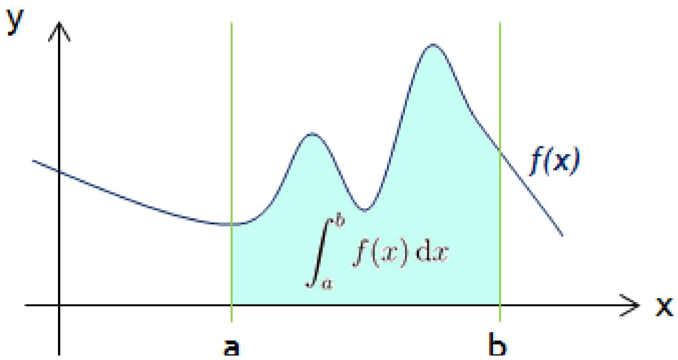
V tejto kapitole sa budeme zaoberať numerickým riešením integrálu. Rozoberieme si dve metódy jeho riešenia a to :

obdĺžnikovú metódu,
lichobežníkovú metódu.

V tejto kapitole sa budeme zaoberať výpočtom integrálu funkcie zadanej tabuľkou.

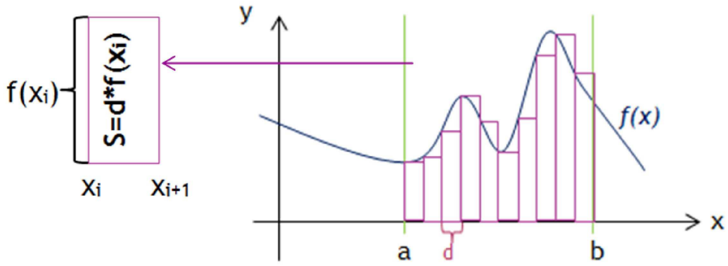
Určitý integrál $\int_a^b f(x)\,\mathrm{d}x$ predstavuje obsah plochy ohraničenej nerovnicami

$a \le x \le b,\ 0 \le y \le f(x)$



Obrázok 4-1 Grafické znázornenia určitého integrálu funkcie *f(x)*

- **Obdĺžniková metóda**



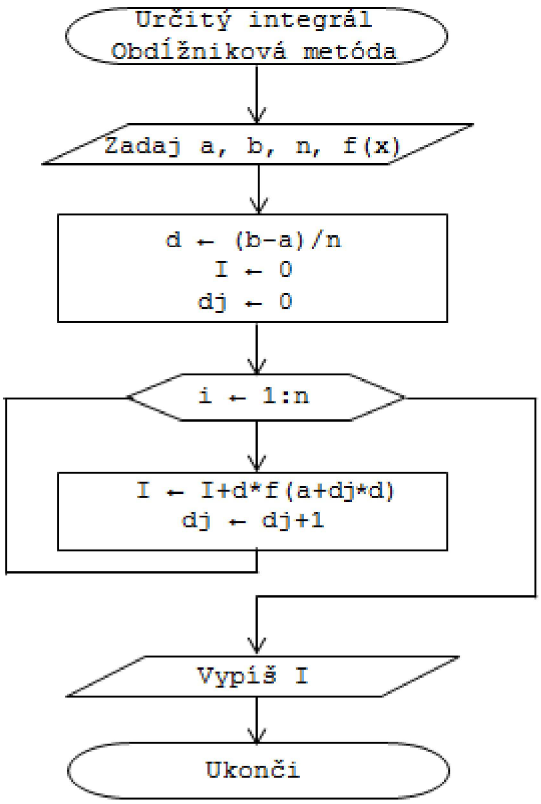
Obrázok 4-2 Obdĺžniková metóda výpočtu určitého integrálu

Podstatou obdĺžnikovej metódy je rozdelenie intervalu <*a,b*> na *n* častí (v našom prípade 10). Plochu rozdelíme na *n* obdĺžnikov, ktorých jednu stranu bude tvoriť premenná $d=(b-a)/n$ alebo $d=x_{i+1}-x_i$ a druhú stranu tvorí funkčná hodnota v bode x_i.
Obsah obdĺžnika potom vieme vypočítať :

$S=d*f(x_i)$

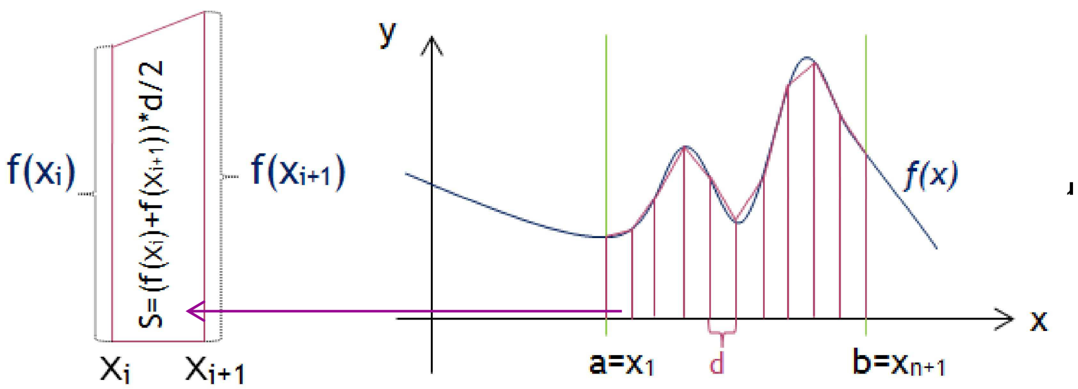
Integrál potom vieme vyjadriť ako súčet obsahov jednotlivých obdĺžnikov.

Algoritmické riešenie určitého integrálu obdĺžnikovou metódou.



Obrázok 4-3 Vývojový diagram výpočtu určitého integrálu obdĺžnikovou metódou

- **Lichobežníková metóda**



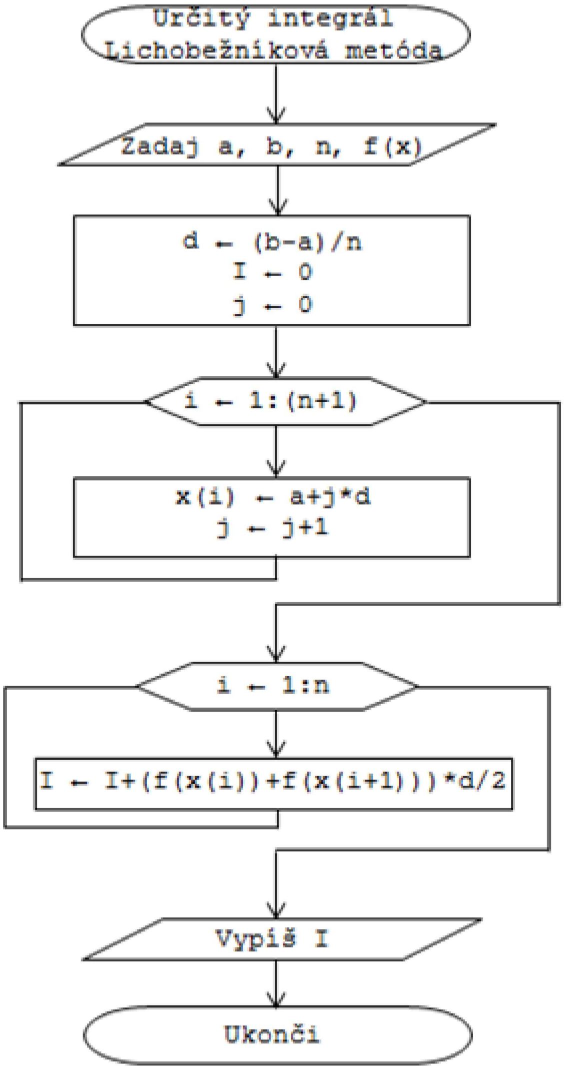
Obrázok 4-4 Lichobežníková metóda výpočtu určitého integrálu

Podstatou lichobežníkovej metódy je rozdelenie intervalu *<a,b>* na *n* častí (v našom prípade 10). Plochu rozdelíme na *n* lichobežníkov, ktorých výšku tvorí $d=(b-a)/n$ alebo $d=x_{i+1}-x_i$. Základňami jednotlivých lichobežníkov sú funkčné hodnoty v bodoch výšky – $f(x_i)$ a $f(x_{i+1})$.
Pre obsah lichobežníka s takto danými stranami platí :

$$S=(f(x_i)+f(x_{i+1}))*d/2$$

Výpočet integrálu sa realizuje sčítaním jednotlivých obsahov lichobežníkov.

Algoritmické riešenie určitého integrálu lichobežníkovou metódou

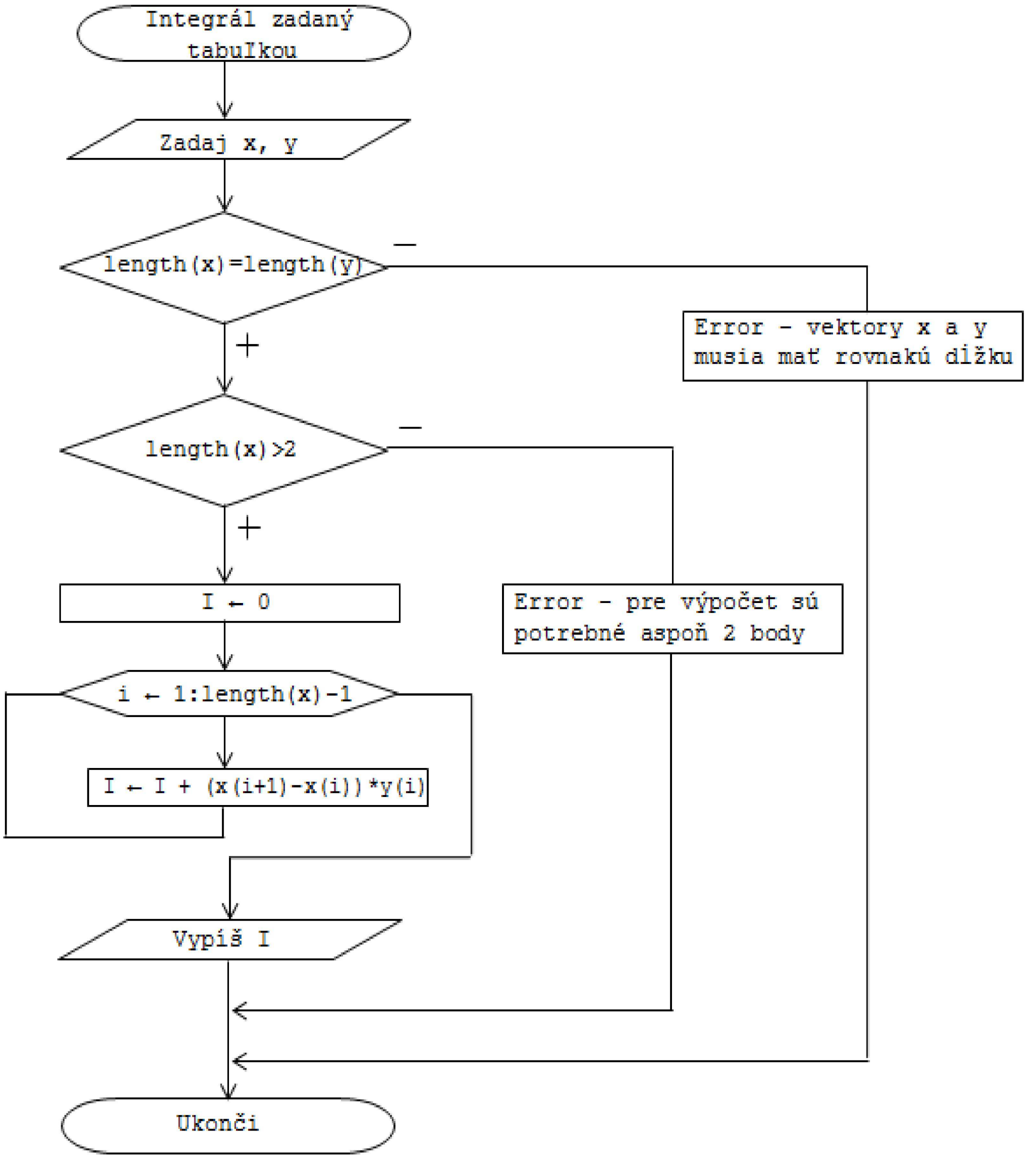


Obrázok 4-5- Vývojový diagram výpočtu určitého integrálu lichobežníkovou metódou

- ## Výpočet integrálu z funkcie zadanej tabuľkou – algoritmické riešenie

V prípade, že predpis funkcie, ktorej integrál máme počítať, nepoznáme a máme len jej body zadané tabuľkou využijeme všetky tieto body *[x_i, y_i] a* výpočet vykonáme po nahradení určitého integrálu súčtom, ktorý vyjadruje nasledujúci vzorec:

$$\sum_{i=1}^{n-1} (x_{i+1} - x_i).y_i$$



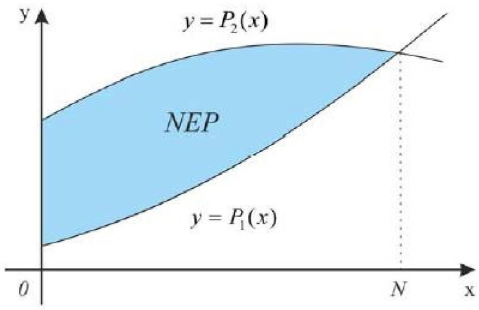
Obrázok 4-6 Vývojový diagram výpočtu určitého integrálu obdĺžnikovou metódou

4.2 Aplikácie určitého integrálu v ekonómii

- **Čistý prebytok zisku**

Majme dva projekty, ktorých rýchlosti ziskov sú dané funkciami $P_1(x)$ a $P_2(x)$, kde x predstavuje počet rokov. Nech funkcia $P_2(x) > P_1(x)$ počas nasledujúcich N rokov.
Čistý prebytok zisku – **NEP** vypočítame nasledovne:

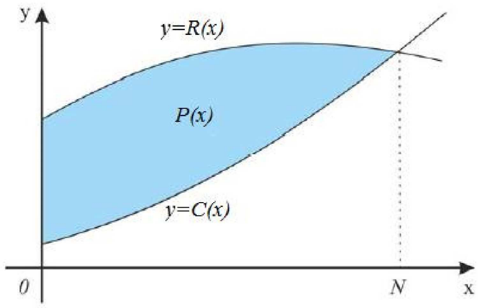
$$NEP = \int_0^N \left[P_2(x) - P_1(x)\right] dx$$



- **Zisk z výrobného zariadenia**

Výrobné zariadenie vytvára príjem rýchlosťou $R(x)$ a náklady na jeho prevádzku rastú rýchlosťou $C(x)$, kde x predstavuje počet rokov. Zariadenie je ziskové, pokiaľ platí nerovnosť $R(x) > C(x)$.
Zisk z výrobného zariadenia ***P***, za N rokov vypočítame podľa vzorca:

$$P(x) = \int_0^N \left[R(x) - C(x)\right] dx$$



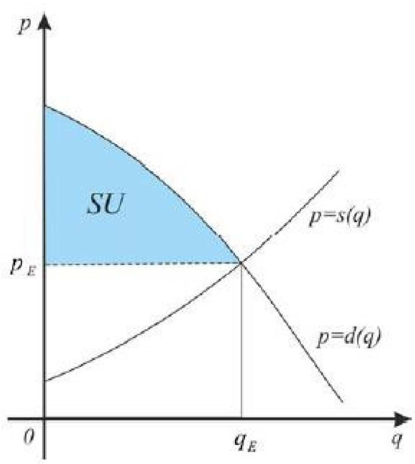
- **Spotrebiteľská úspora, podnikateľský prebytok**

Majme danú funkciu dopytu $\boldsymbol{p = d(q)}$ a funkciu ponuky $\boldsymbol{p = s(q)}$ a súradnice rovnovážneho bodu

Obrázok 4-7 Vývojový diagram výpočtu určitého integrálu pre funkciu zadanú tabuľkou

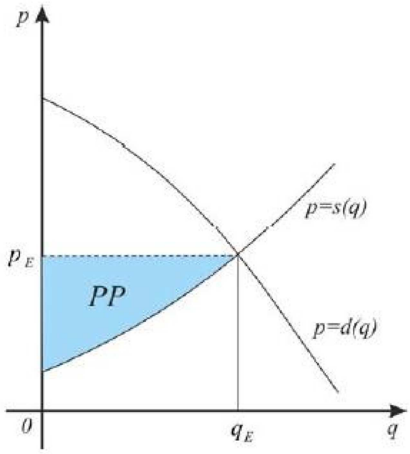
Spotrebiteľská úspora – **SU** predstavuje rozdiel medzi množstvom peňazí, ktoré sú spotrebitelia ochotní minúť za q_E výrobkov a množstvom peňazí, ktoré by minuli pri cene p_E.

$$SU = \int_0^{q_E} d(q)\, dq - p_E q_E$$



Podnikateľský prebytok – **PP** je rozdiel medzi množstvom peňazí, ktoré výrobcovia dostanú za q_E výrobkov predávaných za cenu p_E a množstvom peňazí, za ktoré boli ochotní predať q_E výrobkov.

$$PP = p_E q_E - \int_0^{q_E} s(q)\,dq$$

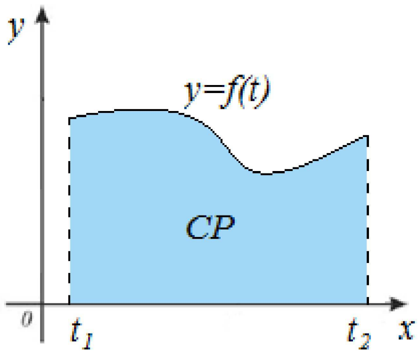


- **Celkový príjem**

Nech je funkcia hustoty toku príjmu *f(t)* v čase *t* spojitá na intervale <t_1,t_2>

Celkový príjem – **CP** v čase <t_1,t_2> určíme podľa vzorca:

$$CP = \int_{t_1}^{t_2} f(t)\,dt$$



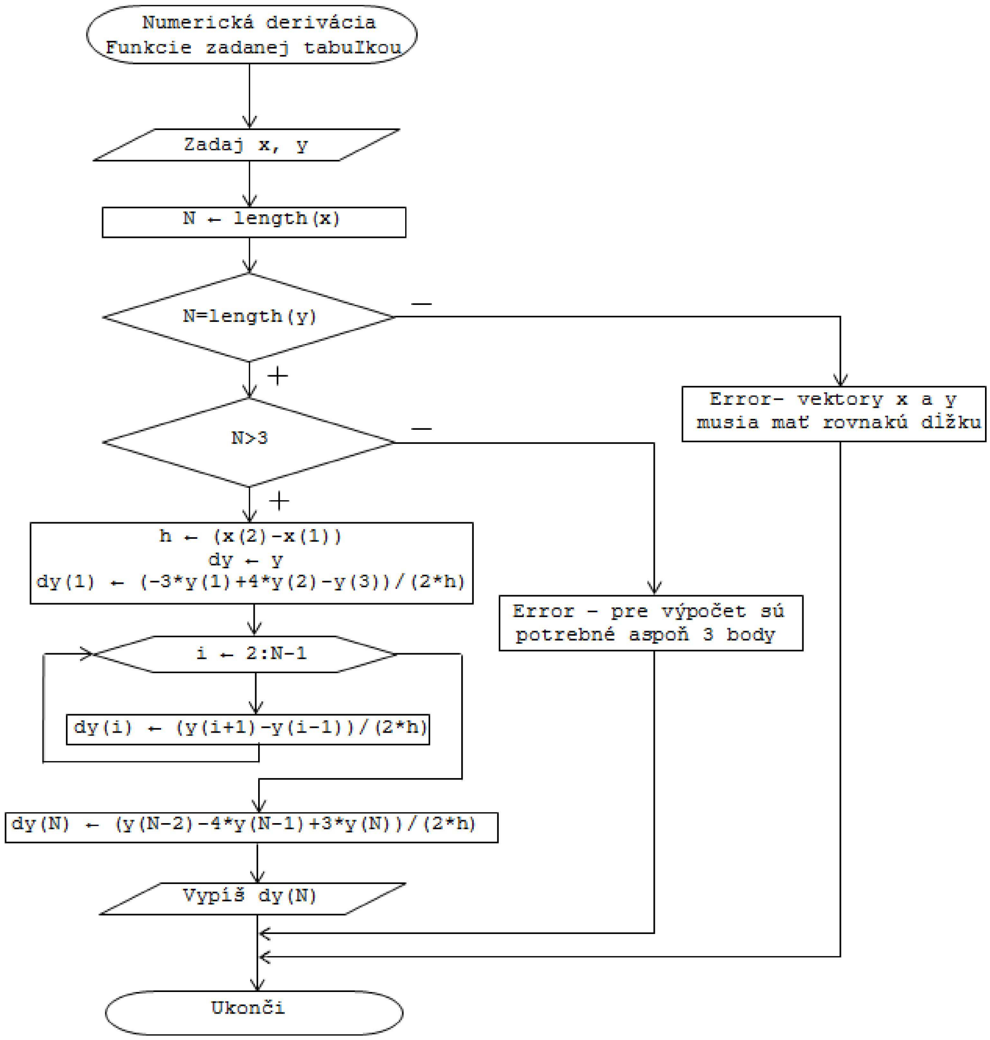
4.3 Algoritmické a numerické riešenie derivácie

Derivácia predstavuje smernicu ku krivke v určitom bode. Pri odhade derivácie funkcie *f(x)* môžeme vychádzať z definície :

$$f'(x) = \lim_{h \to 0}(f(x+h) - f(x))/h$$

kde *h* je z prstencového okolia bodu 0. Po zvolení „malého" *h*, dostaneme odhad derivácie v bode *x*.

Tento postup však nie je možné použiť pri funkciách zadaných tabuľkou.
V prípade, že je funkcia zadaná tabuľkou, s rovnomerným rozdelením bodov $x_1, x_2, x_3, \ldots, x_{n-1}, x_n$, môžeme použiť interpoláciu 2. stupňa a získame vzťahy pre výpočet derivácie v daných bodoch – deriváciu vo vnútorných bodoch $x_2, \ldots, x_{n-1}$ určíme podľa symetrického vzorca a okraje dopočítame z vnútorných bodov.



4.4 Príklady na samostatné riešenie

1. Vypočítajte integrál $\int_{0,1}^{1,5} \frac{\cos(x)}{x^2}\,\mathrm{dx}$ obdĺžnikovou metódou, pre *n* zadané z klávesnice, riešte pomocou funkcií v MATLAB-e a vlastných naprogramovaných funkcií.
2. Vypočítajte integrál $\int_{0,5}^{2,3} \frac{dx}{\sqrt{(1+3x^2)}}$ obdĺžnikovou metódou pre *n*=5, 10 a 15. Výpočty porovnajte.
3. Vypočítajte integrál $\int_{0}^{2\pi} \sin(x)\,dx$, na výpočet použite lichobežníkovú metódu.
4. Majme funkciu danú bodmi uvedenými v tabuľke. Vytvorte funkciu na numerický výpočet určitého integrálu obdĺžnikovou metódou. Aplikujte túto funkciu na výpočet integrálu funkcie danej nasledujúcou tabuľkou

x	1	2.12	3.24	4.36	5.48	6.6	7.72	8.84
y	0.265	0.369	0.985	1.356	1.452	0.874	1.256	2.012

4.5 Pojem funkcie funkcií v jazyku MATLAB

Hľadanie minima funkcie viacerých premenných, numerický výpočet koreň rovnice *f(x)* = 0, riešenie sústavy diferenciálnych rovníc, numerická integrácia sú často vyskytujúce sa problémy pri riešení inžinierskych úloh.
Simulačný jazyk MATLAB s využitím svojich vstavaných funkcií podporuje riešenie napr. týchto problémov:

a) nájdenie nulovej hodnoty funkcie jednej premennej (riešenie nelineárnych rovníc)
b) nájdenie minima funkcie jednej alebo viacerých premenných,
c) numerický výpočet hodnoty určitého integrálu jednej premennej,
d) riešenie sústavy diferenciálnych rovníc (lineárnych,nelineárnych DR)

Funkcie funkcií sú funkcie, ktoré pracujú s funkciami programového prostredia MATLAB v úlohe ich parametra umožňujú prácu s matematickými funkciami namiesto číselných premenných.
Podmienkou pre používanie štandardných vstavaných funkcií simulačného jazyka MATLAB je programatorska zručnosť pri vytváraní ***m-funkcií*** : funkcie funkcií majú ako prvý parameter meno ***novovytvorenej*** funkcie, ďalšie parametre sú dané syntaxou vstavanej funkcie.
Vstavané funkcie jazyka MATLAB pracujúce s matematickými funkciami sú umiestnené v adresári ***funfun***.

funkcia		***popis***	
fmin	→	*minimalizácia funkcie s jednou premennou*	
fmins	→	*minimalizácia funkcie s niekoľkými premennými*	
fplot	→	*zobrazenie priebehu funkcie*	
fzero	→	*nájdenie núl funkcie s jednou premennou*	
ode23	→	*riešenie DR*	$R - K$ 3.*rádu*
ode45			$R - K$ 5.*rádu*
quad			
quad8	→	*numerický integrál*	*v tvare nižšieho rádu*
quadl			*v tvare vyššieho rádu*

- **Zápis matematických funkcií**

V simulačnom jazyku MATLAB môžeme zapísať funkcie pomocou ***m-súborov*** typu ***funkcia*** alebo s využitím priamych objektov ~ s využitím funkcie ***inline.*** Definícia funkcie ***inline*** je dočasné a pri novom spustení programového systému MATLAB je funkcia nedostupná

<u>Príklad 1</u>
Uvažujeme matematickú funkciu s nasledujúcim predpisom

$$f(x) = \frac{1}{(x-0{,}3)^2+0{,}01} + \frac{1}{(x-0{,}9)^2+0{,}04} - 6$$

Táto funkcia môže byť použitá ako vstup pre už vyššie uvedené funkcie. To si vyžaduje vykonať zápis funkcie ***f(x)*** do m-súboru napr. s názvom ***humps.m***:

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

```
function y = humps(x)         %x je vstupná premenná, y je výstupná premenná
y = 1./((x - 0,3).^2 + 0,01) + 1./((x / 0,9).^2 + 0,04) / 6;
```

<u>Príklad 2</u>
S využitím funkcie ***inline*** vytvorte v simulačnom jazyku Matlab zápis funkie ***f(x)*** z príkladu 1.

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

```
>> f=inline('1./((x - 0.3).^2 + 0.01)+ 1./((x - 0.9).^2 + 0.04)- 6')

f =
      Inline function:

      f(x) = 1./((x-0.3).^2+0.01)+1./((x-0.9).^2+0.04)-6
```

Pri výpočte funkčnej hodnoty v danom bode je postup rovnaký nezávisle na spôsobe, ako bola funkcia zadefinovaná (m-súbor resp. inline).

Príklad 3
Vypočítajte hodnotu funkcie ***f(x)*** pre *x* = 2.

Riešenie v programovom prostredí MATLAB:

```
>> f(2)

ans =

      -4.8552
```

Vytvorenie funkcie viacerých premenných prebieha pomocou zápisu:

$$\boldsymbol{inline('funkcia','arg1','arg2',\dots)}$$

kde:

funkcia – reťazec znakov vyjadrujúcich funkciu (vstavanú alebo zadefinovaú užívateľom)
arg1, **arg2** – argumenty – parametre funkcie

Príklad 4
S využitím MATLAB funkcie ***inline*** vytvorte funkciu viacerých premenných

$$f(x,y) = y * sin(x) + x * cos(y).$$

Riešenie v programovom prostredí MATLAB:

```
>> f=inline('y*sin(x)+x*cos(y)', 'x', 'y')

f =
      Inline function:
      f(x,y) = y*sin(x)+x*cos(y)
```

- ## Grafické zobrazenie funkcií

Funkcia ***fplot*** umožňuje zobraziť matematické funkcie, ktoré sú buď vstavané funkcie jazyka MATLAB alebo funkcie vytvorené užívateľom.

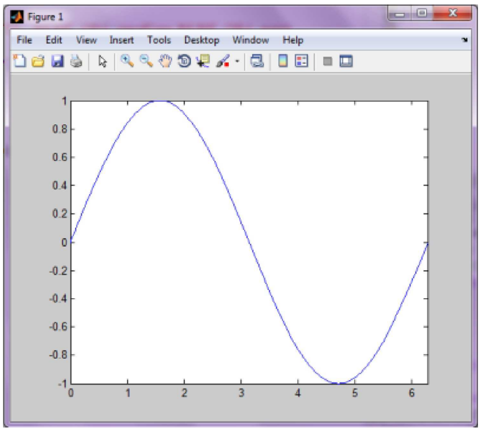
$$\boldsymbol{fplot('fmeno',limit)}$$

kde:

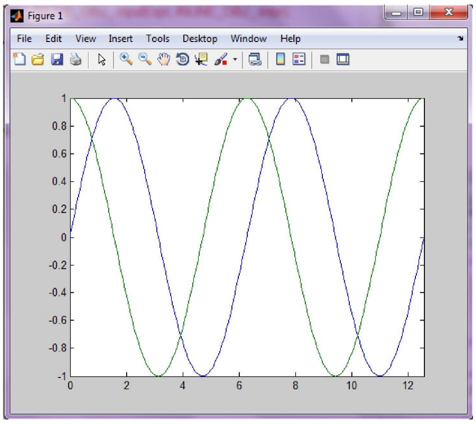
fmeno – názov funkcie v jazyku MATLAB
limit – interval na ktorom budeme zobrazovať funkciu s názvom ***fmeno***

Príklad 5
Príklad s využitím funkcie ***fplot***. Graficky znázornite goniometrické funkcie na definovanom intervale.

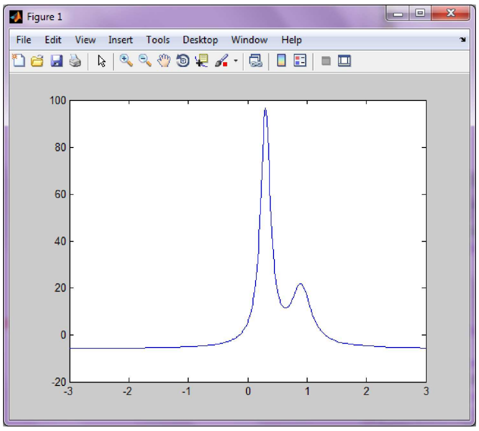
```
>> fplot('sin',[0,2*pi]);
```




```
>> fplot('[sin(x),cos(x)]',[0,4*pi]);
```

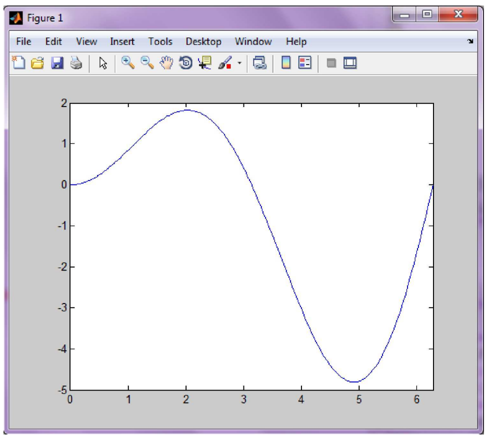



```
>> fplot('humps',[-3,3]);
```



Podobne je možné použiť ako vstupný argument funkcie ***plot*** aj funkciu definovanú ako ***inline*** objekt

```
>> f=inline('x.*sin(x)','x');
>> fplot(f,[0,2*pi])
```



• Minimum funkcie a hľadanie nulových bodov – funkcia *fminbnd*

Na minimalizáciu zadefinovanej funkcie simulačný jazyk MATLAB využíva vstavané funkcie ktoré umožňujú:
- minimalizácia funkcie s jednou premennou;
- minimalizáciu funkcie s viacerými premennými;
- hľadanie nulového bodu funkcie s jednou premennou:

Minimalizácia funkcie s jednou premennou :

$$fminbnd('fun', x_1, x_2, options)$$

kde:

fun – reťazec znakov, pomocou ktorého je zadefinová funkcia alebo názov premennej, v ktorej je funkcia zadefinová pomocou príkazu ***inline***
x_1,x_2 **–** začiatok a koniec intervalu na ktorom hľadáme minimum
options **–** voľby pre hľadanie minima

Príklad 6
Nájdite minimum funkcie sin na intervale <0,2;π>, využite pritom vstavanú funkciu ***fminbnd.***

Riešenie v programovom prostredí MATLAB:

```
>> k=fminbnd('sin',0,2*pi)
k =
    4.7124
>> min=sin(k)
min =
   -1.0000
```

• Práca s rovnicami s jednou premennou – funkcia *fzero*

Na riešenie koreňov algebraických rovníc môžeme využiť funkciu ***roots*** (ľavá strana polynómu), avšak na hľadanie numerického riešenia rovnice, ktorá nie je algebraická, používame funkciu ***fzero*** **-** nájdenie núl funkcie(nulových bodov), ktorú voláme nasledovne:

$$fzero('fun', x_0)$$

kde :

fun – zápis funkcie, ktorá predstavuje ľavú stranu funkcie
x_0 – počiatočný odhad riešenia, alebo interval kde sa ma nachádzať koreň

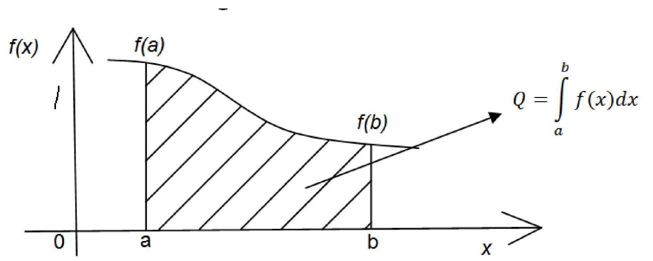
Príklad 7
Nájdite riešenie rovnice $\cos(2x) * \sin(3x)$ = 0 s využitím funkcie ***fzero***

```
>> x=fzero(inline('cos(2*x)*sin(3*x)'),2)
x =
    2.0944
>> x=fzero(inline('cos(2*x)*sin(3*x)'),[1.5 2.2])
x =
    2.0944
```

Samostatne riešte nasledujúce rovnice s využitím funkcie ***fzero***, s ich úpravou na tvar *f(x)* = 0

1. $x * e^x = 1$
2. $x^2 + x + 1 = 0$
3. $\sin(x) = \frac{x}{10}$

- **Numerická integrácia**



Na numerické integrovanie funkcie jednej premennej v tvare $\int_a^b f(x)dx$ ponúka MATLAB dve štandardné vstavané funkcie ***quad, quadl.***

- ***quad*** – implementuje adaptívne Simpsonovo pravidlo

Pre výpočet Simpsonovým pravidlom rozdelíme interval <*a*,*b*> na párny počet podintervalov a v každom intervale vykonávame náhradu pôvodnej funkcie parabolou. Vzorec pre výpočet Simpsonovho pravidla je:

$$\int_a^b f(x)dx \approx \frac{h}{3}\left(f(x_0) + f(x_{2n}) + 4\left(f(x_1) + f(x_3) + \ \dots\ + f(x_{2n-1})\right) + 2\left(f(x_2) + f(x_4) + \cdots + f(x_{2n-2})\right)\right).$$

- ***quadl*** – implementuje adaptívne Lobattovo pravidlo

Funkcie quad,quadl majú analogickú syntax:

$$\boldsymbol{hodnota_{int} = quad('fname', dolna\_hr, horna\_hr);}$$

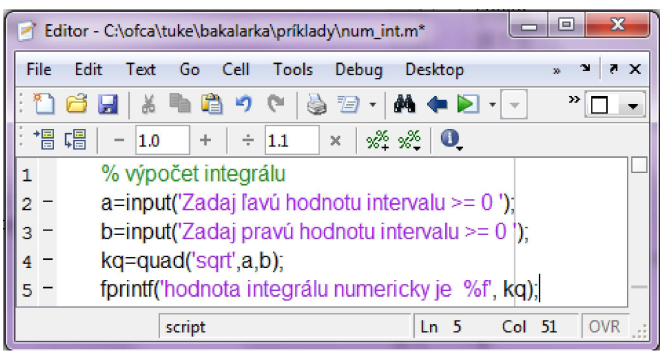
kde:

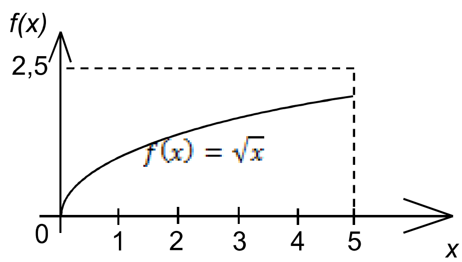
fname – meno užívateľom vytvorenej funkcie alebo aj meno funkcie programového prostredia MATLAB, (*built-in-funkcie)*, ktoré sú napríklad: sin, sqrt ...
dolna_hr, horna_hr – hranice intervalu, ktorých počítame integrál z definovanej funkcie

Príklad 8

Vypočítajte s využitím funkcie ***quad*** v simulačnom jazyku MATLAB hodnotu integrálu $\int_0^5 \sqrt{x}\, dx$; na intervale <0,5> .

Riešenie v programovom prostredí MATLAB:





Medzi ďalšie volitelné parametre pre funkciu ***quad*** :

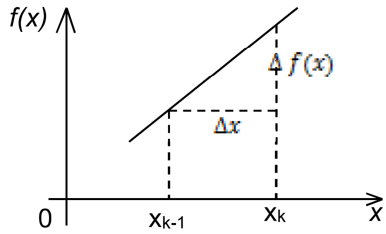
$$\boldsymbol{Q = quad('fun', a, b, tol, trace)}$$

kde:

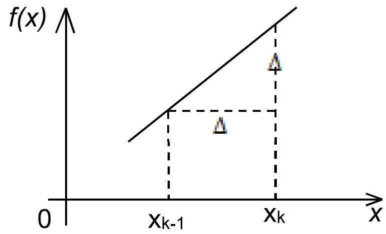
tol- presnosť integrálu ($1e^{-6}$)
trace – nenulový výpis výpočtovej rekurzie

- **Numerická derivácia - funkcia diff**

Funkcia ***diff*** vypočíta diferencie medzi hodnotami vo vektore a vytvára nový vektor:



$f'(x_k) = \frac{f(x_k)-f(x_{k-1})}{x_k-x_{k-1}}$ spätná diferencia



$f'_{KM}(x_k) = \frac{f(x_{KM})-f(x_K)}{x_{KM}-x_K}$ dopredná diferencia

Príklad 9
Predpokladajme, že funkcia *f(x)* má tvar polynómu:

$f(x) = x^5 - 3x^4 - 11x^3 + 27x^2 + 10x - 24$

Vypočítajte deriváciu tejto funkcie na intervale [-4,5] s využitím funkcie ***diff***.

Riešenie v programovom prostredí MATLAB:

```
>> x=-4:0.1:5;%generovanie hodnôt nezávislej premennej
>> f=x.^5-3*x.^4-11*x.^3+27*x.^2+10*x-24;
>> df=diff(f)./diff(x);
```

Príklad na samostatné riešenie:

Vypočítajte integrál $\int_0^{3\pi} \sqrt{4\cos(2t)^2 + \sin(t)^2 + 1}\, dt$ pomocou vytvorenia funkcie ako ***m-súbor*** a ako objekt.

[t,y]=**solver**('odefun', cas_int, poc_pod)

kde***:***

solver – riešiteľ , napríklad vstavané funkcie *ODE45, ODE23* ...
'odefun' – meno ***m-súboru*** funkcie , ktorá obsahuje definovaný predpis systému DR

.

4.6 Analytické a numerické riešenie diferenciálnych rovníc

Fyzikálne systémy (elektrické, mechanické, tepelné, hydraulické) ,ktoré sú vo všeobecnosti popísané systémom lineárnych alebo nelineárnych diferenciálnych rovníc (lineárne dynamické systémy sú popísané LDR s konštantnými koeficientmi).

Analytické riešenie LTI (Linear Time Invariant) systému DR v časovej oblasti získame:

- riešením DR bez a s pravou stranou a sčítaním homogenného a partikulárneho riešenia
- riešením v Laplaceovej transformácii a časový originál riešenia získame spätnou Laplaceovou transformáciou (L^{-1}).

DR môžeme riešiť **numerickými metódami**. Funkciu, ktorú dostávame po každom kroku pri numerickom výpočte nazývame aproximácia analytického riešenia.

Pri riešení diferenciálnych rovníc v sa budeme vžívať dva postupy a to :

- na riešenie DR použijeme vstavanú funkciu **ode45 (využíva metódu Runge-Kutta 4.stupňa),**
- a vlastnú funkciu využívajúcu algoritmus metódy **Runge – Kutta.**

Je nutné si uvedomiť, že simulačný jazyk MATLAB neumožňuje riešiť DR vyššieho rádu ako 1. rádu, t.j. DR s definovanou počiatočnou podmienkou v tvare

$$\frac{dy(t)}{dt} = f(t, y), \quad PP: y(0) = y_0$$

Z tohto dôvodu je pri oboch postupoch potrebné vykonať transformáciu DR *n*-tého rádu na systém *n* diferenciálnych rovníc 1. rádu. (prepísanie DR do substitučného kanonického tvaru). Princíp spočíva v vysvetlený na konkrétnych príkladoch DR.

- **Prepis diferenciálnej rovnice na substitučný kanonický tvar**

Prepis lineárnej DR 2.rádu do substitučného kanonického tvaru:

Uvažujme DR, ktorá ma nasledujúci tvar:

$$y''(t) + 3y'(t) + y(t) = 1$$

Na prepis do substitučného kanonického tvaru zvolíme substitúciu :

$$y = x_1(t), \; y' = x_2(t)$$

$$\boxed{\begin{aligned} y'(t) &= x_1'(t) = x_2(t) \\ y''(t) &= x_2'(t) = 1 - 3x_2(t) + x_1(t) \end{aligned}} \longrightarrow \boxed{\text{Substitučný kanonický tvar}}$$

V každom kroku výpočtu je dôležité poznať hodnoty stavových premenných ktoré sú reprezentované vektorom [*x1; x2*]**.** V prostredí MATLAB si vytvoríme funkciu, ktorú nazveme ***„dif.m“***. Jej výstupný parameter (***xout***) predstavuje v každom kroku derivácie stavových premenných: **xout =** [*x1´; x2´*].

```
function [ xout ] = dif(t,x)
x1=x(2);
x2=x(1)-3*x(2)+1;
xout=[x1;x2];
end
```

Prepis lineárnej DR 3.rádu do substitučného kanonického tvaru:

Uvažujme všeobecnú DR 3.rádu s pravou stranou.

$$y'''(t) + a_2 y''(t) + a_1 y'(t) + a_0 y(t) = b_0 * u(t)$$

Definujme stavové premenné: $y = x_1(t)$, $y' = x_2(t)$, $y'' = x_3(t)$

Prepis do substitučného kanonického tvaru:

$$y'(t) = x_1'(t) = x_2(t)$$
$$y''(t) = x_2'(t) = x_3(t)$$
$$y'''(t) = x_3'(t) = b_0 u(t) - a_0 x_1 - a_1 x_2 - a_2 x_3$$

DR n = 3 je prepísaná do substitučného kanonického tvaru troch DR 1.rádu.

- **Riešenie DR s využitím vstavanej funkcie *ode45* jazyka MATLAB**

Funkcia ***ode45*** vypočíta riešenie DR *n*-tého rádu, ktorá bola zapísana do substitučného kanonického tvaru(súbor ***„dif.m“***), funkcia ***ode45*** používa názov funkcie ***„dif“*** ako parameter. Funkcie na riešenie diferenciálnych rovníc teda považujeme za ***„funkcie funkcií“***. Základná syntax funkcie ode45 je nasledovná:

```
[t,x] = ode45(@dif, [t0,tf], [P])
```

kde:

`dif` – funkcia, ktorá obsahuje definovaný predpis riešenej DR v tvare systému diferenciálnych rovníc 1. rádu

`[t0,tf]` – začiatočná a koncová hodnota intervalu, na ktorom vykonávame riešenie DR

`[P]` – vektor počiatočných podmienok

Výstup z funkcie ***ode45***:

t – stĺpcový vektor časových bodov
y – matica riešenia (jednotlivé stĺpce obriešenie DR a jeho derivácie)

Výsledok riešenia funkcie ***ode45*** je možné vykresliť pomocou funkcie ***plot*** nasledovne :

```
plot(t,y(:,1))          %riešenie DR x1(t)
plot(t,y (:,2))         %derivácia riešenia DR x2(t)
```

Pozn. Namiesto funkcie ode45 je možné použiť aj iné solvery (riešitele), napr. ode23 a

Príklad
Uvažujeme nelineárnu diferenciálnu rovnicu 2. rádu ,ktorú reprezentuje Van-der-Polov oscilátor. Namodelujte riešenie tejto DR v simulačnom jazyku MATLAB.

$$y''(t) = g(t, y, y') = y'(1 - y^2) - y$$

$$y''(t) - y'(t) * \left(1 - y^2(t)\right) + y(t) = 0$$

Substitúcia: $y = x_1(t)$, $y' = x_2(t) y(t) = x_1(t)$

Vytvorenie substitučného kanonického tvaru:

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

vander.m:

```
% Matematicky zápis DR2.rádu prepísaný
%do stavového priestoru
function xder = vander(t,x)
xder=[x(2);x(2).*(1-x(1).^2)-x(1)];
return
```

Pozn. `xder = [x(2);x(2).*(1 - x(1).^2) / x(1)]` - maticový zápis

Hlavný program :

difrov2r.m :

```
% program na riešenie nelineárnej diferenciálnej rovnice
% 2.rádu -  Van - Der - Pol
% oscilátor - zadaný v stavovom priestore
x0=[0 0.25]'; % inicializácia počiatočných podmienok
t0=0; tf=10;  % definícia časového intervalu
[t,x]=ode23('vander',[t0,tf],[0;0.25]);
[t,x]=ode45('vander',[t0,tf],[0;0.25]);
subplot(211); plot(t,x(:,1)); ...
title('riešenie y(t)'); xlabel('t'); grid; ...
subplot(212); plot(t,x(:,2)); ...
title('prvá derivácia y(t)'); xlabel('t'); grid; ...
plot(t,x); % dva grafy v jednom obrázku
title('Van-der-Pol rovnica - časová história ');
pause
```

```
plot(t,x(:,1),'k-',t,x(:,2),'k--')          %využitie viacparametrovej funkcie na grafický výstup
```

- **Metóda Runge – Kutta 4.rádu**

Metóda Runge – Kutta je jednou z najznámejších numerických metód na riešenie DR. Podstata spočíva v aproximácii lineárnych kombinácií funkčných hodnôt v zvolených bodoch. Vyjadrenie nového stavu systému pomocou predchádzajúceho stavu je dané vzťahmi :

$$k_1 = T * f\,(t_n,\, x_n)$$

$$k_2 = T * f\,(t_n + ½T,\, x_n + ½Tk_1)$$

$$k_3 = T * f\,(t_n + ½T\,, x_n + ½Tk_2)$$

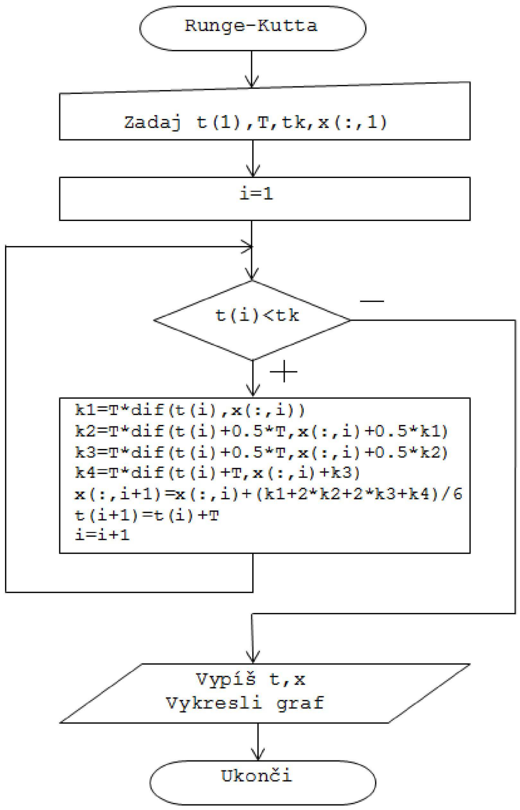
$$k_4 = T * f\,(t_n + T,\, x_n + Tk_3)$$

$$x_{n+1} = x_n + (k_1+2k_2+2k_3+k_4)/6$$

pre n = 1, 2, 3...

Zavedieme si premenné:

t_k – hodnota času t, pri ktorej chceme výpočet ukončiť
T – integračný krok, s ktorým sa bude výpočet realizovať
t(1) – počiatočná hodnota t, pri ktorej sa výpočet odštartuje ak i = 1
x(:,1) – počiatočný vektor x závislej premennej



Obrázok 4-8 Vývojový diagram pre metódu Range – Kulta – vlastná naprogramovaná funkcia

4.7 Riešenie lineárnych diferenciálnych rovníc analyticky a s využitím vstavaných funkcií v jazyku MATLAB

Diferenciálne rovnice nazývame rovnice, v ktorých sa vyskytujú funkcie a ich derivácie. K diferenciálnym rovniciam nás vedú fyzikálne a matematické úlohy.

- **Druhy diferenciálnych rovníc**

a) **Obyčajné diferenciálne rovnice** – ako neznáma vystupuje reálna funkcia jednej reálnej premennej a tiež derivácie tejto funkcie.

- Hovoríme, že diferenciálna rovnica je *n*-tého rádu, ak v nej vystupuje *n*-tá derivácia neznámej funkcie *y* a ak už v nej nevystupuje žiadna jej derivácia vyššieho rádu ako *n*.
    - Lineárne diferenciálne rovnice
    - Nelineárne diferenciálne rovnice

b) **Parciálne diferenciálne rovnice** – ako neznáma vystupuje funkcia dvoch alebo viac premenných a v ktorej vystupujú parciálne derivácie tejto neznámej funkcie.

- **Lineárne diferenciálne rovnice 1.rádu**

V úlohách, ktoré vedú na riešenie diferenciálnych rovníc 1. rádu je obyčajne vopred známa hodnota hľadanej funkcie v nejakom bode t_0.

Úlohou je nájsť medzi všetkými riešeniami diferenciálnej rovnice $y' = f(t, y)$ také riešenie $y = y(t)$, ktoré spĺňa počiatočnú podmienku $y(t_0) = t_0$, kde y_0 je dané číslo. Takýto typ úlohy sa nazýva *Cauchyho úloha.*

- **Lineárne diferenciálne rovnice n-tého rádu**

Homogénna lineárna diferenciálna rovnica (DR bez pravej strany) je DR tvaru

$$a_n y^{(n)}(t) + a_{n-1} y^{(n-1)}(t) + \cdots + a_1 y'(t) + a_0 y(t) = 0,$$

kde a_i, i =0,1, ..., *n* sú koeficienty DR.

Nehomogénna lineárna diferenciálna rovnica (DR s pravou stranou) je DR tvaru

$$a_n y^{(n)}(t) + a_{n-1} y^{(n-1)}(t) + \cdots + a_1 y'(t) + a_0 y(t) = u,$$

kde a_i, $i = 0,1, \dots, n$ sú koeficienty DR a u je budiaca funkcia.

Všeobecné riešenie y DR je súčtom homogenného riešenia $\bar{y}$ a partikulárneho riešenia y^* , tj. $y = \bar{y} + y^*$.

- **Riešenie LDR Laplaceovou transformáciou**

Laplaceova transformácia nám ponúka jednoduché riešenie lineárnych diferenciálnych rovníc s konštantnými koeficientami.

1) DR v časovej oblasti pretransformujeme do LT so zohľadnením počiatočných podmienok
2) Vypočítame obrazový prenos systému $G(s) = \frac{Y(s)}{U(s)}$
3) Z obrazového prenosu systému vyjadríme obraz riešenia DR *Y(s)*

Y(s) = G(s) . U(s)

kde

$U(s)$ predstavuje vstup do systému
$Y(s)$ predstavuje výstup zo systému

4) Použitím vzorcov pre spätnú LT získame riešenie DR v časovej oblsati.

Y(s) → y(t)

PRÍKLAD 1

Majme LDR tvaru:

$$y''(t) - 2y'(t) - 3y(t) = 7 * \cos(3t).$$

Nájdite všeobecné riešenie LDR pre nulové počiatočné podmienky analytickým výpočtom a s využitím vstavanej funkcie ode45 v jazyku MATLAB.

Riešenie DR v časovej oblasti:

1. Najprv vyriešime homogénnu DR
   $y''(t) - 2y'(t) - 3y(t) = 0$
   Pre výpočet homogénnej DR potrebujeme vypočítať charakteristickú rovnicu a jej korene:

   $s^2 - 2s - 3 = 0$
   $(s+1)(s-3) = 0$
   $s_1 = -1$
   $s_2 = 3$
   Na základe koreňov charakteristickej rovnice všeobecné riešenie $\bar{y} = C_1 * e^{-t} + C_2 * e^{3t}$

2. Následne vypočítame partikulárne riešenie DR s uvažovaním špeciálnej pravej strany
   V našom prípade je pravá strana $f(t) = 7 * \cos(3t)$.
   Z všeobecného partikulárneho riešenia $y^* = x^k * e^{ax} * (A * \cos(bx) + B * \sin(bx))$ odhadneme
   s = 0, *a* = 0, *b* = 3.
   Keďže komplexne združené číslo $a + ib = 3i$ nie je v tomto prípade žiadnym koreňom, preto
   k = 0.
   A teda partikulárne riešenie tejto LDR je
   $y^* = x^0 * e^{0t} * (A * \cos(3t) + B * \sin(3t)) = A * \cos(3t) + B * \sin(3t)$

3. Získané partikulárne riešenie zderivujeme:

   $(y^*)' = -3A * \sin(3t) + 3B * \cos(3t)$
   $(y^*)'' = -9A * \cos(3t) - 9B * \sin(3t)$

4. Dosadíme partikulárne riešenia do pôvodnej LDR

   $(y'' - 2y' - 3y = 7 * \cos(3t))$
   $-9A * \cos(3t) - 9B * \sin(3t) - 2 * (-3A * \sin(3t) + 3B * \cos(3t)) -$
   $-3 * (A * \cos(3t) + B * \sin(3t)) = 7 * \cos(3t)$

Upravíme:

$$-9A * \cos(3t) - 9B * \sin(3t) + 6A * \sin(3t) - 6B * \cos(3t) - 3A\cos(3t) - 3B * \sin(3t) = 7 * \cos(3t)$$
$$(-12A - 6B) * \cos(3t) + (-12B + 6A) * \sin(3t) = 7 * \cos(3t)$$

$\cos(3t)$: $\quad -12A - 6B = 7 \qquad A = -\frac{7}{15}$

$\sin(3t)$: $-12B + 6A = 0 \qquad B = -\frac{7}{30}$

Partikulárnym riešením DR je $\quad y^* = -\frac{7}{15} * \cos(3t) - \frac{7}{30} * \sin(3t)$

5. Všeobecné riešenie danej nehomogénnej LDR hľadáme v tvare $y = \bar{y} + y^*$

$$y = C_1 * e^{-t} + C_2 * e^{3t} - \frac{7}{15} * \cos(3t) - \frac{7}{30} * \sin(3t)$$

6. Určenie integračných konštanty z počiatočných podmienok:

$y(0) = 0$:

$$y(t) = C_1 * e^{-t} + C_2 * e^{3t} - \frac{7}{15} * \cos(3t) - \frac{7}{30} * \sin(3t)$$

$$y'(t) = -C_1 * e^{-t} + 3C_2 e^{3t} + \frac{7}{5}\sin(3t) - \frac{7}{10}\cos(3t)$$

$$C_1 * e^{-1*0} + C_2 * e^{3*0} - \frac{7}{15} * \cos(3*0) - \frac{7}{30} * \sin(3*0) = 0$$

$$C_1 + C_2 - \frac{7}{15} + 0 = 0$$

$$C_1 = \frac{7}{15} - C_2$$

$$C_1 = \frac{7}{40}$$

$y'(0) = 0$:

$$-C_1 * e^{-t} + 3C_2 e^{3t} + \frac{7}{5}\sin(3t) - \frac{7}{10}\cos(3t) = 0$$

$$-C_1 + 3 * C_2 + 0 - \frac{7}{10} = 0$$

$$4 * C_2 = \frac{7}{6}$$

$$C_2 = \frac{7}{24}$$

7. Celkové analytické riešenie DR :

$$y = \frac{7}{40} * e^{-t} + \frac{7}{24} * e^{3t} - \frac{7}{15} * \cos(3t) - \frac{7}{30} * \sin(3t)$$

Nájdenie riešenia DR pomocou Laplaceovej transformácie

$$y''(t) - 2 * y'(t) - 3 * y(t) = 7 * cos(3t), PP: y(0) = y'(0) = 0$$

1. Prevedieme LDR do Laplaceovej transformácie

$$s^2Y(s) - sy(0) - y'(0) - 2 * (sY(s) - y(0)) - 3Y(s) = \frac{7s}{s^2 + 3^2}$$

$$s^2Y(s) - 2sY(s) - 3Y(s) = \frac{7s}{s^2 + 3^2}$$

2. Vyjadríme obraz riešenia DR v LT : Y (s) :

$$Y(s) * (s^2 - 2s - 3) = \frac{7s}{s^2 + 3^2}$$

$$Y(s) = \frac{7s}{(s^2 + 3^2) * (s^2 - 2s - 3)}$$

3. Rozklad prenosovej funkcie na parciálne zlomky

$$\frac{7s}{(s^2+3^2)*(s^2-2s-3)}=\frac{A}{s+1}+\frac{B}{s-3}+\frac{Cs+D}{s^2+9}$$

$$7s=A*(s^2+9)*(s-3)+B*(s^2+9)*(s+1)+(Cs+D)*(s-3)*(s+1)$$

$$\begin{array}{ll} s^3: & 0=A+C+D \\ s^2: & 0=-2A+B-3C+D \\ s^1: & 7=-3A-2B+9C+9D \\ s^0: & 0=-3B-27C+9D \end{array}$$

$$\left(\begin{array}{cccc|c} 1 & 0 & 1 & 1 & 0 \\ -2 & 1 & -3 & 1 & 0 \\ -3 & -2 & 9 & 9 & 7 \\ 0 & -3 & -27 & 9 & 0 \end{array}\right) => \quad \begin{array}{l} A=\frac{7}{40} \\ B=\frac{7}{24} \\ C=-\frac{7}{15} \\ D=-\frac{7}{10} \end{array}$$

$$\frac{7s}{(s^2+3^2)*(s^2-2s-3)}=\frac{\frac{7}{40}}{s+1}+\frac{\frac{7}{24}}{s-3}-\frac{\frac{7}{15}s+\frac{7}{10}}{s^2+9}$$

4. Následne použijeme spätnú laplaceovú transformáciu, aby sme získali analytické riešenie DR y(t) v časovej oblasti.

$$\frac{7}{40}*\frac{1}{s+1}\div\frac{7}{40}*e^{-t}$$

$$\frac{7}{24}*\frac{1}{s-3}\div\frac{7}{24}*e^{3t}$$

$$\frac{-\frac{7}{15}s-\frac{7}{10}}{s^2+9}=-\frac{7}{15}*\frac{s}{s^2+3^2}-\frac{7}{10}*\frac{1}{s^2+3^2}$$

$$-\frac{7}{10}*\frac{1}{s^2+3^2}=-\frac{7}{30}*\frac{3}{s^2+3^2}\div-\frac{7}{30}*\sin(3t)$$

$$-\frac{7}{15}*\frac{s}{s^2+3^2}\div-\frac{7}{15}*\cos(3t)$$

5. Riešenie y(t) pomocou spätnej Laplaceovej transformácie

$$y=\frac{7}{40}*e^{-t}+\frac{7}{24}*e^{3t}-\frac{7}{15}*\cos(3t)-\frac{7}{30}*\sin(3t)$$

Riešenie v jazyku MATLAB :

Pri riešení funkciami DR v simulačnom jazyku MATLAB je nutné si uvedomiť, že MATLAB neumožňuje riešiť DR vyššieho rádu, ale iba systém diferenciálnych rovníc 1.rádu. Preto používame transformáciu diferenciálnych rovníc *n*-tého rádu na diferenciálne rovnice 1-ho rádu.

Prepis do substitučného kanonického tvaru

$$y(t)=x_1$$

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

$$y''(t)=x_2'=\frac{1}{a_2}*(7\cos(3t)-a_1*x_2-a_0*x_1)=7\cos(3t)+2x_2+3x_1$$

Program v simulačnom jazyku MATLAB

difrov.m

```
function xder = difrov(t,x)
%Zápis diferenciálnej rovnice 2.radu pomocou 2rovnic 1.radu
global  a0 a1 a2;
xder=[x(2) ; (7*cos(3*t)-a1.*x(2)-a0.*x(1))./a2];
return
```

analyt.m

```
%analytické riešenie
function d=analyt(t)
d=(7/40)*exp(-t)+(7/24)*exp(3*t)-(7/15)*cos(3*t)-(7/30)*sin(3*t);
return
```

chyba.m

```
%odhad chyby:
function chyba = rozdiel(d,y)
rozdiel = abs(d-y(:,1));
chyba = max(rozdiel);
fprintf('Maximálna odchýlka = %d \n', chyba)
return
```

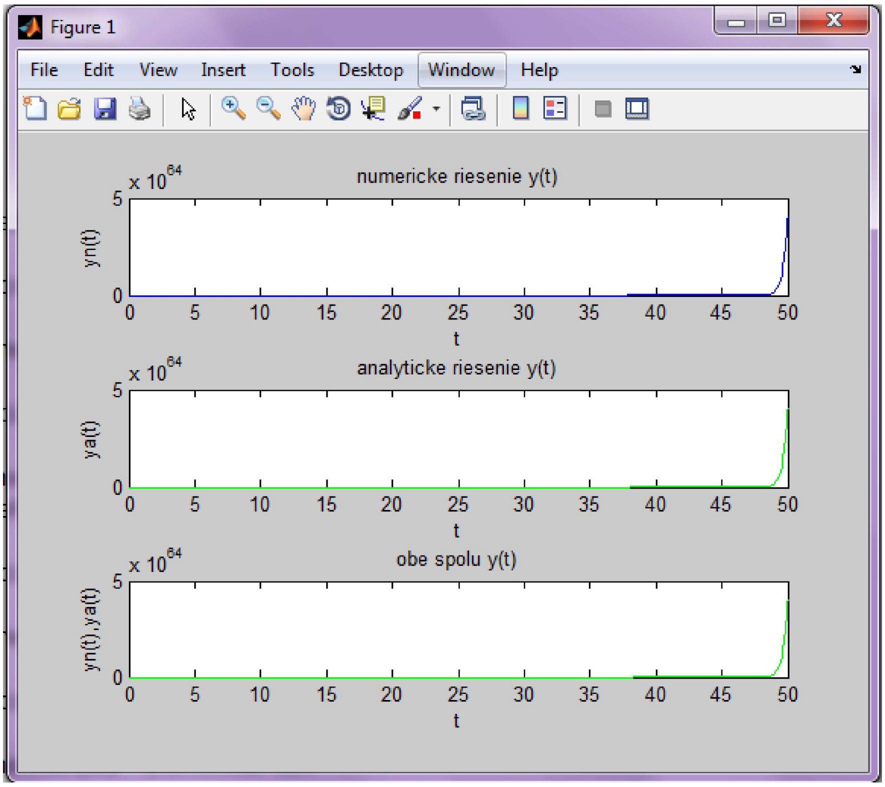
Hlavný program

```
global a;
a = input('Zadaj koeficienty LDR: [a(1) a(2) a(3)]\n');
T(1) = input('Zadaj počiatočnú hodnotu časového intervalu pre riešenie LDR:
 \n');
T(2)  = input('Zadaj  konečnú  hodnotu  časového  intervalu  pre  riešenie  LDR:
 \n');
PP = input('Zadaj počiatočné podmienky: [P(0) P(1)]\n');

 %riešenie LDR pomocou funkcie ode45:
[t,y]=ode45('difrov',T,PP);

%analytické riešenie:
d = analyt(t);

%odhad chyby:
chyba = rozdiel(d,y);
%vykreslenie vyriešených priebehov:
subplot(3,1,1)
plot(t,y(:,1))
title('numerické riešenie y(t)'),xlabel('t'),ylabel('yn(t)')
subplot(3,1,2)
plot(t,d,'g')
title('analytické riešenie y(t)'),xlabel('t'),ylabel('ya(t)')
subplot(3,1,3)
plot(t,y(:,1),t,d,'g')
title('obe spolu y(t)'),xlabel('t'),ylabel('yn(t),ya(t)');
```



Obrázok 4-9 Porovnanie analytického a numerického riešenia LDR v prostredí MATLAB s využitím funkcie odr45

PRÍKLAD 2

$y''(t) + 2y'(t) + y(t) = t \qquad PP: y(0) = y'(0) = 0$

Riešenie homogénnej diferenciálnej rovnice

$y'' + 2y' + y = 0$

1. Charakteristická rovnica: $s^2 + 2s + 1 = 0$
   $(s + 1)(s + 1) = 0$

Korene charakteristickej rovnice : $s_1 = -1$ $\qquad s_2 = -1$

Všeobecné riešenie vyplývajúce z charakteristickej rovnice: $\bar{y} = C_1 e^{-t} + C_2 t e^{-t}$

2. Riešenie špeciálnej pravej strany :

   $f(t) = t$

   Partikulárne riešenie : $y^*(t) = A_1 t + A_0$, pričom

   $A_0 = -2 \qquad A_1 = 1$

3. Všeobecné riešenie DR $y = \bar{y} + y^*$ má tvar:

$$y = C_1 e^{-t} + C_2 t e^{-t} + t - 2$$

4. Výpočet integračných konštánt C_1,C_2 z počiatočných podmienok:

$$y(0) = 0: \qquad C_1 = 2$$

$$y'(0) = 0: \qquad C_2 = 1$$

Celkové riešenie lineárnej DR získane analyticky má tvar:

$$y(t) = t - 2 + 2e^{-t} + te^{-t}$$

Riešenie lineárnej DR pomocou Laplaceovej transformácie

$$y''(t) + 2y'(t) + y(t) = t \text{ , počiatočné podmienky : } PP: y(0) = y'(0) = 0$$

1. Prepis do LT LDR: $s^2Y(s) + 2sY(s) + Y(s) = \frac{1}{s^2}$

2. Lapl. obraz riešenia lineárnej DR Y(s) má tvar: $Y(s) = \frac{1}{s^2 * (s^2+2s+1)}$

3. Rozklad Laplacovho obrazu riešenia DR (prenosovej funkcie) Y(s) na parciálne zlomky

$$\frac{1}{s^2 * (s+1)^2} = \frac{A}{s^2} + \frac{B}{s} + \frac{C}{(s+1)^2} + \frac{D}{s+1}$$

Výpočtom získame : *A* = 1 , *B* = -2 , *C* = 1 , *D* = 2

4. Celkové riešenie DR v časovej oblsati získame spätnou Laplaceovou transformáciou :

$$y(t) = t - 2 + t * e^{-t} + 2 * e^{-t}$$

Príklad:
Vytvorte funkciu v prostredí MATLAB, ktorá numericky vyrieši zadanú diferenciálnu rovnicu s využitím vstavanej funkcie ***ode45***. Uvažovanú DR prepíšte do substitučného kanonického tvaru.

4.8 Zadanie č.2: Riešenie LDR (*n* = 2) s konštantnými koeficientami analyticky a algoritmicky v prostredí MATLAB

ZADANIE:

Riešenie LDR 2. alebo 3. rádu s konštantnými koeficientami analyticky a algoritmicky v programovom prostredí MATLAB.

OBSAH ZADANIA:

A. *Analyticky vyriešiť a nájsť celkové riešenie LDR v časovej oblsati.*
B. *Analyticky vyriešiť celkové riešenie LDR v Laplaceovej transformácií s uvažovaním nájdenia časového originálu riešenia y(t) aplikovaním inverznej Laplaceovej transormácie.*
C. *Programové riešenie v simulačnom jazyku MATLAB (s využitím vstavanej funkcie **ode45** a vlastnej funkcie RK).*

ÚLOHA:

A. Riešenie v časovej oblasti:
Nájdite celkové riešenie LDR s využitím metódy špeciálnej pravej strany pre DR tvaru :

$y''(t) - 2y'(t) - 3y(t) = 3e^{2t}$ *s počiatočnými podmienkami:* $y(0) = y'(0) = 0$

<u>Riešenie LDR so špeciálnou pravou stranou:</u>

1. Charakteristická rovnica:

$s^2 - 2s - 3 = 0$
$(s - 3) * (s + 1) = 0$

Korene charakteristickej rovnice : $s_1 = 3$ $s_2 = -1$

Celkové riešenie vyplývajúce z charakteristickej rovnice:

$$\bar{y} = C_1 e^{-t} + C_2 e^{3t}$$

2. Partikulárne riešenie predpokladáme v tvare $y^* = Ae^{2t}$ nakoľko prava strana DR má tvar :

$$f(t) = 3e^{2t}$$

Derivovaním partikulárneho riešenia a následným výpočtom získame : $A = -1$

3. Celkové riešenie LDR : $y = C_1 e^{-t} + C_2 e^{3t} - e^{2t}$

4. Výpočet integračných konštánt C_1,C_2 z počiatočných podmienok: $C_1 = \frac{1}{4}$, $C_2 = \frac{3}{4}$

5. Celkové analytické riešenie LDR má tvar:

$$y = \frac{1}{4} e^{-t} + \frac{3}{4} e^{3t} - e^{2t}$$

B. Riešenie LDR pomocou Laplaceovej transformácie:

Nájdite celkové riešenie LDR pomocou priamej a inverznej LT:

$y''(t) - 2y'(t) - 3y(t) = 3e^{2t}$ $\qquad PP: y(0) = y'(0) = 0$

1. Zápis LDR do LT: $\quad s^2Y(s) - 2sY(s) - 3Y(s) = \frac{3}{s-2}$

2. Riešenie Y(s) vyjadrenie v Laplaceovej transformácií: $\quad Y(s) = \frac{3}{(s-2)(s-3)(s+1)}$

3. Rozklad prenosovej funkcie Y(s) (riešenia LDR v LT) na parciálne zlomky:

$$\frac{3}{(s-2)(s-3)(s+1)} = \frac{-1}{s-2} + \frac{\frac{3}{4}}{s-3} + \frac{\frac{1}{4}}{s+1}$$

4. Použitím inverznej LT získame výsledné riešenie DR v časovej oblasti:

$$y(t) = -e^{2t} + \frac{3}{4}e^{3t} + \frac{1}{4}e^{-t}$$

C. Riešenie v programovom prostredí MATLAB:

1. Prepis do kanonického substitučného tvaru LDR:

$$a_2y'' + a_1y' + a_0y(t) = u(t) \quad \rightarrow \quad y''(t) - 2y'(t) - 3y(t) = 3e^{2t}$$

Zvolíme substitúciu : $\quad y(t) = x_1$

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

$$y''(t) = x_2' = \frac{1}{a_2} * (3e^{2t} - a_1 * x_2 - a_0 * x_1) = 3e^{2t} + 2x_2 + 3x_1$$

2. Riešenie v jazyku MATLAB

difrov.m

```
function xder = difrov(t,x)
%Zápis diferenciálnej rovnice 2.radu pomocou 2rovnic 1.radu
global a0 a1 a2;
xder=[x(2); (3*exp(2*t) - a1*x(2)- a0*x(1))./a2];
return
```

analyt.m

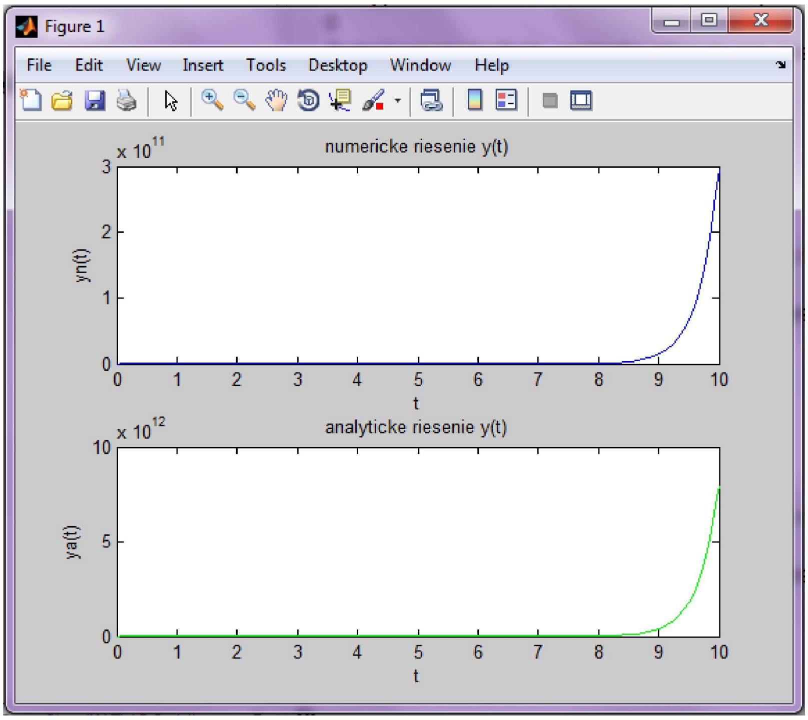
```
%analytické riešenie
function d = analyt(t)
d=(1/4)*exp(-t)+(3/4)*exp(3*t)-exp(2*t);
return
```

Úloha

Naprogramujte hlavný program, ktorého výsledkom bude :

a) funkcia na výpočet riešenia LDR numericky pomocou funkcie ***ode45*** a vlastnej naprogramovanej funkcie R-K,
b) funkcia na výpočet chyby medzi analytickým riešením LDR a jej numerickým riešením získaným metódou R-K (***ode45***/R-K),
c) riešenie LDR numerické a analytické znázornite graficky

Výstupom hlavného programu je nasledujúci obrázok :



Obrázok 4-10 Porovnanie analytického a numerického riešenia LDR v MATLAB-e