

UNIVERZITA PALACKÉHO V OLOMOUCI
PŘÍRODOVĚDECKÁ FAKULTA
KATEDRA MATEMATICKÉ ANALÝZY A APLIKACÍ MATEMATIKY

DIPLOMOVÁ PRÁCE

Diferenční metody pro diferenciální rovnice
a inkluze



Vedoucí diplomové práce:
RNDr. Jitka Machalová, Ph.D.
Rok odevzdání: 2011

Vypracovala:
Hana Jašková
MPM, II. ročník

Prohlášení

Prohlašuji, že jsem vytvořila tuto diplomovou práci samostatně za vedení RNDr. Jitky Machalové, Ph.D. a že jsem v seznamu použité literatury uvedla všechny zdroje použité při zpracování práce.

V Olomouci dne 31. března 2011

Poděkování

Ráda bych na tomto místě poděkovala vedoucí diplomové práce RNDr. Jitce Machalové, Ph.D. za obětavou spolupráci i za čas, který mi věnovala při konzultacích. Dále si zasluží poděkování můj počítač, že vydržel moje pracovní tempo, a typografický systém $\text{\TeX}$, kterým je práce vysázena.

Obsah

Použité značení	4
Úvod	5
1 Přípravná kapitola	6
1.1 Symbolika O	6
1.2 Mnohoznačné zobrazení	6
2 Obecná jednokroková metoda pro diferenciální rovnice	8
2.1 Analýza stability obecné jednokrokové metody při pevném kroku h	11
2.2 Eulerova metoda	16
2.3 Metody Runge–Kutta	24
2.4 Obecná jednokroková metoda pro soustavu diferenciálních rovnic	31
3 Diferenciální inkluze	34
3.1 Eulerova metoda pro diferenciální inkluze	37
3.2 Strategie výběru selekce	46
3.2.1 Výběr s minimální normou	46
3.2.2 Výběr s minimální variací	49
3.2.3 Výběr s respektováním referenční trajektorie	52
3.3 Metody Runge–Kutta pro diferenciální inkluze	56
3.4 Řešené příklady	58
4 M-soubory	78
4.1 M-soubor pro Eulerovu explicitní metodu	78
4.2 M-soubor pro klasickou metodu Runge–Kutta	79
4.3 M-soubor pro Eulerovu explicitní metodu pro řešení soustav diferenciálních rovnic	80
4.4 M-soubor pro řešení diferenciálních inkluzí Eulerovou explicitní metodou pomocí různých strategií výběru selekce 1	81
4.5 M-soubor pro řešení diferenciálních inkluzí Eulerovou explicitní metodou pomocí různých strategií výběru selekce 2	87
4.6 M-soubor pro řešení diferenciálních inkluzí metodami Runge–Kutta s různými strategiemi výběru selekce 1	94
4.7 M-soubor pro řešení diferenciálních inkluzí metodami Runge–Kutta s různými strategiemi výběru selekce 2	101
Závěr	108
Literatura	109

Použité značení

$\mathbb{R}$	množina všech reálných čísel
$\mathbb{C}$	množina všech komplexních čísel
$\mathbb{N}$	množina všech přirozených čísel
$\mathbb{R}^+$	pravá polorovina množiny všech reálných čísel $\mathbb{R}$
$\mathbb{R}^*$	rozšířená množina reálných čísel o $\pm\infty$
$\mathbb{C}^-$	levá polorovina množiny všech komplexních čísel $\mathbb{C}$
$\mathbb{R}^m$	m-rozměrný reálný prostor
$\operatorname{Re} z$	reálná část komplexního čísla z
$\langle a, b \rangle$	uzavřený interval s krajními body a, b
$ x $	absolutní hodnota komplexního čísla x
$\langle f, g \rangle$	skalární součin dvou funkcí f a g
$P = [c, d]$	souřadnice bodu P v rovině $\mathbb{R}^2$
$C(J)$	množina funkcí spojitých na intervalu J
$C^2\langle x_0, b \rangle$	množina funkcí, které mají spojitě derivace až do řádu dvě včetně na intervalu $\langle x_0, b \rangle$
$\mathbf{w} \in \mathbb{R}^s$	sloupcový vektor o složkách $w_1, \dots, w_s$
$\mathbf{w}^T$	transpozice vektoru $\mathbf{w}$
$A = (a_{ij})_{i,j=1}^s$	čtvercová matice o velikosti $s \times s$ s prvky a_{ij}
A^{-1}	inverzní matice k matici A
$I = \operatorname{diag}(1, \dots, 1)$	diagonální matice s jedničkami na diagonále
$\operatorname{haus}(M, N)$	Hausdorffova vzdálenost množin M a N

Úvod

Cílem této diplomové práce je nastudovat numerické metody pro řešení diferenciálních rovnic a následně je aplikovat na řešení diferenciálních inkluzí. Numerických metod zabývajících se touto problematikou je mnoho. My se v této práci zaměříme pouze na obecnou jednokrokovou metodu, konkrétněji na Eulerovu metodu, která je nejjednodušší obecnou jednokrokovou metodou a na metody Runge–Kutta.

V první kapitole se seznámíme se symbolikou O a s mnohoznačným zobrazením, jejichž znalost je potřebná k pochopení dalších pojmů.

Ve druhé kapitole se budeme věnovat některým numerickým metodám, kterými můžeme řešit diferenciální rovnice. Nejprve se budeme zabývat obecnou jednokrokovou metodou, dále Eulerovou metodou, metodami Runge–Kutta a na závěr této kapitoly si ukážeme, jak vyřešit soustavu diferenciálních rovnic pomocí obecné jednokrokové metody.

Ve třetí kapitole se budeme zabývat aplikací numerických metod z kapitoly 2 na řešení diferenciálních inkluzí. Tato řešení budeme znázorňovat pomocí grafů.

Dalším úkolem této diplomové práce je sestavit programy v matematickém softwaru Matlab pro řešení konkrétních úloh. Tyto programy jsou uvedeny v poslední kapitole.

Ráda bych upozornila na situaci, kdy se příkazový řádek, který zadávám do Matlabu, nevejde na šířku stránky. V tomto případě je daný příkaz logicky rozdělen do více řádků.

Všechny M-soubory uvedené v této práci jsou odladěny v matematickém softwaru Matlab 7.1 na notebooku Fujitsu SIEMENS model AMILO L7300 s procesorem Intel Celeron a operační pamětí 512 MB.

V této práci je pro označení konce příkladů použit symbol ★ .

1 Přípravná kapitola

V této kapitole se budeme zabývat tématy, která nejsou přímo obsahem této práce, ale úzce s ním souvisí.

1.1 Symbolika O

Symbolika O se často používá pro vyjádření chyb matematických výrazů.

Definice 1. Nechť f je reálná nebo komplexní funkce definovaná v okolí bodu $a \in \mathbb{R}^*$. Nechť g je funkce kladná v okolí bodu a . Symbol

$$f(x) = O(g(x)) \quad \text{pro } x \rightarrow a$$

značí, že

$$\limsup_{x \rightarrow a} \frac{|f(x)|}{g(x)} < \infty,$$

tedy že funkce $\frac{|f(x)|}{g(x)}$ je v okolí bodu a ohraničená.

V tomto textu budeme tento symbol používat v souvislosti s chybou aproximace numerické metody, kde funkce $g(h) = h^k$, $k \in \mathbb{R}$, $h \in \mathbb{R}$ a $h \rightarrow 0$, tj. budeme užívat symbol $O(h^k)$.

1.2 Mnohoznačné zobrazení

Definice 2. Nechť X a Y jsou dvě množiny. **Mnohoznačným zobrazením** F z X do Y rozumíme zobrazení, které každému prvku $x \in X$ přiřadí množinu $F(x)$, kde $F(x) \in Y$.

Mnohoznačné zobrazení budeme značit následovně: $F : X \multimap Y$.

Mnohoznačné zobrazení můžeme charakterizovat jeho grafem

$$\text{Graf}(F) = \{(x, y) \in X \times Y | y \in F(x)\}.$$

Příklad mnohoznačného zobrazení

$$F : \mathbb{R}^+ \multimap \mathbb{R}$$

definované vztahem

$$F(x) = \langle e^{-x}, 1 \rangle, \quad \text{pro každé } x \in \mathbb{R}^+.$$

Toto mnohoznačné zobrazení můžeme také znázornit následujícím grafem:



Obrázek 1

Na Obrázku 1 vidíme, že toto zobrazení je v počátku souřadnicového systému jednoznačné, v bodech intervalu $(0, \infty)$ však nikoli. Bodům x z intervalu $(0, \infty)$ je přiřazen celý interval hodnot $\langle e^{-x}, 1 \rangle$, tedy zobrazení F je opravdu mnohoznačné.

2 Obecná jednokroková metoda pro diferenciální rovnice

V této kapitole se budeme zabývat numerickým řešením počáteční úlohy

$$y'(x) = f(x, y(x)), \quad (1)$$

$$y(x_0) = a, \quad (2)$$

kde $x \in \langle x_0, b \rangle$, $a, x_0, b \in \mathbb{R}$ a podmínku (2) budeme nazývat **počáteční podmínkou**. Úlohu (1), (2) budeme nazývat **Cauchyova úloha**.

Dále budeme v této kapitole předpokládat:

Předpoklad 1. *Uvedená úloha (1), (2) má na intervalu $\langle x_0, b \rangle$ právě jedno řešení $y(x)$. To je zaručeno právě tehdy, když pro pravou stranu $f(x, y(x))$ diferenciální rovnice (1) platí*

(i) *funkce $f(x, y(x))$ je definovaná a spojitá funkce dvou proměnných v pásu*

$P = \{[x, y]; x_0 \leq x \leq b, -\infty < y < \infty\}$, *kde x_0, b jsou konečná čísla,*

(ii) *funkce $f(x, y(x))$ splňuje v tomto pásu Lipschitzovu podmínku vzhledem k y*

s konstantou nezávislou na x , tj. existuje konstanta $L > 0$ taková, že platí

$$|f(x, y) - f(x, z)| \leq L|y - z| \quad \text{pro každé } [x, y], [x, z] \in P.$$

Hledáme-li řešení dané počáteční úlohy analyticky, může se stát, že toto řešení buď nenalezneme vůbec, nebo je vyjádřeno složitými vzorci. Proto se velmi často tato počáteční úloha řeší přibližně, a to pomocí nějaké numerické metody. V této kapitole se nejprve budeme zabývat obecnou jednokrokovou metodou, v další části Eulerovou metodou, která je nejjednodušším typem obecné jednokrokové metody a dále pak metodami Runge–Kutta.

Jedním typem numerických metod jsou tzv. **diskrétní metody**, kdy na nějaké diskrétní množině bodů $\{x_n\}$, $x_n \in \langle x_0, b \rangle$, kde $x_0, b \in \mathbb{R}$, hledáme pouze některé hodnoty přesného řešení uvažované funkce. Dostáváme tak posloupnost hodnot přibližného řešení v bodech této diskrétní množiny, které budeme značit

$\{y_n\}$. Dále x_n se dá zapsat jako $x_n = x_0 + n h_n$, kde $h_n > 0$. Číslo h_n budeme nazývat **integrační krok**. Tento krok můžeme volit jako konstantní nebo jej můžeme dle potřeby měnit. My jej budeme pro jednoduchost volit jako konstantní, a to tak, aby byl poslední bod posloupnosti $\{x_n\}$ roven pravému krajnímu bodu intervalu, na němž řešení hledáme. Dostaneme tak ekvidistantní posloupnost bodů $\{x_n\}$. Tento konstantní integrační krok budeme značit h .

Mezi diskrétní metody s konstantním krokem h patří např. obecné k -krokové metody, speciálně pro $k = 1$ **obecná jednokroková metoda**. Obecnou jednokrokovou metodou získáváme posloupnost přibližných hodnot hledaného řešení počáteční úlohy (1), (2) pomocí rekurentního vztahu

$$y_{n+1} = y_n + h \Phi(x_n, y_n, h), \quad n = 0, 1, \dots \quad (3)$$

přičemž klademe $y_0 = a$, což plyne z počáteční podmínky (2). Funkce Φ tří reálných proměnných ze vztahu (3) závisí na diferenciální rovnici (1).

Tedy v obecné jednokrokové metodě postupujeme tak, že daný interval $\langle x_0, b \rangle$ rozdělíme s krokem h tak, abychom dostali ekvidistantní posloupnost bodů. Dále hodnotu přibližného řešení y_{n+1} úlohy (1), (2) v bodě x_{n+1} počítáme na základě znalosti jedné hodnoty y_n v předcházejícím bodě x_n , viz vztah (3), přičemž $y_0 = a$.

Obecně **k -krokovou metodou** rozumíme algoritmus, který hodnotu přibližného řešení y_k úlohy (1), (2) v bodě x_k počítá na základě znalosti k hodnot $y_0, y_1, \dots, y_{k-1}$ v bodech $x_0, x_1, \dots, x_{k-1}$.

Důležitou vlastnost, kterou u numerické metody pro řešení počáteční úlohy sledujeme, je konvergence, tj. zajímá nás, jestli vůbec a za jakých podmínek konverguje posloupnost hodnot přibližného řešení získaná danou numerickou metodou k přesnému řešení uvažované počáteční úlohy.

Budeme tedy uvažovat obecnou jednokrokovou metodu s konstantním krokem $h > 0$ a studovat chování $\{y_n\}$ při $h \rightarrow 0$ a pevném $x_n = x_0 + n h, x_n \in \langle x_0, b \rangle$, které je jinak libovolné. Chceme-li studovat chybu přibližného řešení v pevném bodě $x = x_n$, musíme při zmenšování kroku h zvětšovat n . V opačném případě bychom nezjistili nic užitečného, protože by platilo $x_n = x_0 + n h \rightarrow x_0$

a $y_n \rightarrow y_0 = y(x_0)$. Tedy limitní přechod v následující definici je nutno chápat tak, že současně $h \rightarrow 0$ a $n \rightarrow \infty$ přičemž platí, že $x_n = x_0 + n h = x$, $x_n \in \langle x_0, b \rangle$.

Definice 3. Obecná jednokroková metoda je **konvergentní**, jestliže pro každé $x \in \langle x_0, b \rangle$ platí

$$\lim_{\substack{h \rightarrow 0 \\ x_n = x}} y_n = y(x), \quad (4)$$

kde y_n je přibližné řešení úlohy (1), (2) s integračním krokem h a y je přesné řešení této úlohy.

Při numerickém řešení dané počáteční úlohy vznikají chyby dvojího druhu, zaokrouhlovací a chyby aproximace, kterým se také říká **diskretizační chyby metody** a které jsou opět dvojí: lokální a globální.

Definice 4. Lokální diskretizační chybou obecné jednokrokové metody na intervalu $\langle x_n, x_{n+1} \rangle$ nazveme výraz $L(y(x_n); h)$, který je dán vztahem

$$L(y(x_n); h) = y(x_{n+1}) - y(x_n) - h \Phi(x_n, y_n, h).$$

Je to tedy nepřesnost, s níž hodnoty teoretického řešení dané úlohy splňují rekurentní vztah (3), ze kterého se počítá hodnota y_{n+1} přibližného řešení.

Chování lokální chyby se obvykle charakterizuje veličinou, která se nazývá řád diferenční metody.

Definice 5. Řádem diferenční metody rozumíme největší přirozené číslo p takové, že pro danou metodu aplikovanou na libovolnou počáteční úlohu (1), (2) s dostatečně hladkým řešením y platí při každém pevném n a $h \rightarrow 0$ odhad

$$L(y(x_n); h) = O(h^{p+1}),$$

tj. pro $h \rightarrow 0$ jsou lokální diskretizační chyby $L(y(x_n); h)$ uvažované metody řádově velikosti h^{p+1} .

Měřítko toho, jak přesně nám posloupnost hodnot $\{y_n\}$ přibližného řešení uvažované počáteční úlohy, kterou získáme pomocí rekurentního vztahu obecné jednokrokové metody, aproximuje přesné řešení dané úlohy, je **globální diskretizační chyba** $e_n = y(x_n) - y_n$. Tato chyba je samozřejmě závislá na lokálních diskretizačních chybách.

Další důležitou vlastností, kterou u numerických metod sledujeme, je stabilita. Stabilita numerické metody zajistí, že malá změna počátečních dat jen málo ovlivní výsledek.

2.1 Analýza stability obecné jednokrokové metody při pevném kroku h

Při zjišťování stability obecné jednokrokové metody je potřebné znát tzv. funkci stability $R(z)$, kterou dostaneme následujícím postupem.

Nechť $\varphi(x)$ je dostatečně hladké řešení diferenciální rovnice (1), tj. platí $\varphi'(x) = f(x, \varphi(x))$. Pak můžeme na jeho okolí funkci f linearizovat a to následovně

$$y'(x) = f(x, \varphi(x)) + \frac{\partial f}{\partial y}(x, \varphi(x))(y(x) - \varphi(x)) + \dots$$

Užijeme-li rovnosti $f(x, \varphi(x)) = \varphi'(x)$, zavedeme-li substituci

$$\bar{y}(x) = y(x) - \varphi(x)$$

a užijeme-li pravidlo pro počítání s derivacemi, že $\bar{y}'(x) = y'(x) - \varphi'(x)$, dostaneme

$$\bar{y}'(x) = \frac{\partial f}{\partial y}(x, \varphi(x))\bar{y}(x) + \dots = J(x)\bar{y}(x) + \dots$$

V prvním přiblížení můžeme předpokládat, že Jakobián $J(x)$ je konstanta, ozn. ji $\lambda \in \mathbb{C}$. Po linearizaci (uvažujeme pouze první člen tohoto rozvoje, ostatní zanedbáme) a přeznačíme-li $\bar{y}(x)$ na $y(x)$, obdržíme diferenciální rovnici ve tvaru

$$y' = \lambda y.$$

Aplikujeme-li na tuto diferenciální rovnici obecnou jednokrokovou metodu, dostaneme rekurentní vztah

$$y_{n+1} = R(\lambda h)y_n \quad n = 0, 1, \dots \quad (5)$$

Označíme-li $z = \lambda h, z \in \mathbb{C}$, dostáváme ze vztahu (5) zmiňovanou funkci stability $R(z)$. Tato funkce pak závisí na konkrétní obecné jednokrokové metodě, přičemž platí $R(\lambda h) = 1 + h \frac{\Phi(x_n, y_n, h)}{y_n}$.

Definice 6. Obecná jednokroková metoda (3) je pro dané h **absolutně stabilní**, jestliže

$$|R(z)| \leq 1,$$

kde $R(z)$ je funkce stability dané metody ze vztahu (5), $z \in \mathbb{C}, z = \lambda h$.

Obecná jednokroková metoda (3) je pro dané h **relativně stabilní**, jestliže

$$|R(z)| \leq e^z,$$

kde $R(z)$ je funkce stability dané metody ze vztahu (5), $z \in \mathbb{C}, z = \lambda h$.

Definice 7. Množina

$$S = \{z \in \mathbb{C}; |R(z)| \leq 1\}$$

se nazývá **oblast absolutní stability**, kde $R(z)$ je funkce stability dané metody ze vztahu (5).

Definice 8. Obecná jednokroková metoda (3) je **A - stabilní**, jestliže

$$S \supset \mathbb{C}^- = \{z \in \mathbb{C}; \operatorname{Re} z \leq 0\}.$$

Definice 9. Obecná jednokroková metoda (3) je **L - stabilní**, jestliže je A - stabilní a jestliže platí

$$\lim_{z \rightarrow \infty} R(z) = 0,$$

kde $R(z)$ je funkce stability dané metody ze vztahu (5), $z \in \mathbb{C}, z = \lambda h$.

Poznámka 1. Teorie A - stability, při níž analýzu stability provádíme pomocí funkce stability $R(z)$, je založena na autonomních lineárních úlohách. Tato analýza se tedy používá pro úlohy tvaru $y'(x) = \lambda y(x)$, popřípadě také pro úlohy $y'(x) = \lambda(y(x) + g(x))$, kde $g(x)$ je libovolná spojitá funkce proměnné x , které se linearizováním dají převést na autonomní lineární úlohu, viz např. Příklad 1. Pro obecné nelineární úlohy se užívá teorie B - stability, kterou si ukážeme později.

Příklad 1. Vyšetřete stabilitu obecné jednokrokové metody dané vztahem

$$y_{n+1} = y_n + hf(x_n, y_n)$$

pro úlohu

$$y'(x) = \lambda(y(x) - \cos(x))$$

$$y(0) = 0$$

na intervalu $\langle 0, 1.5 \rangle$ s krokem $h = 1.5/80$ pro $\lambda = -20$ a $\lambda = -200$. Nalezněte oblast absolutní stability.

Řešení: Nejprve musíme určit funkci stability $R(z)$: Postupujeme-li podle návodu z odstavce 2.1, dostaneme novou diferenciální rovnici ve tvaru

$$y'(x) = \lambda y(x),$$

protože $\frac{\partial f}{\partial y} = \lambda$. Aplikujeme-li na tuto rovnici zadanou obecnou jednokrokovou metodu, dostaneme její tvar

$$y_{n+1} = y_n + \lambda h y_n,$$

tedy

$$y_{n+1} = (1 + \lambda h)y_n,$$

a tedy

$$R(z) = 1 + z,$$

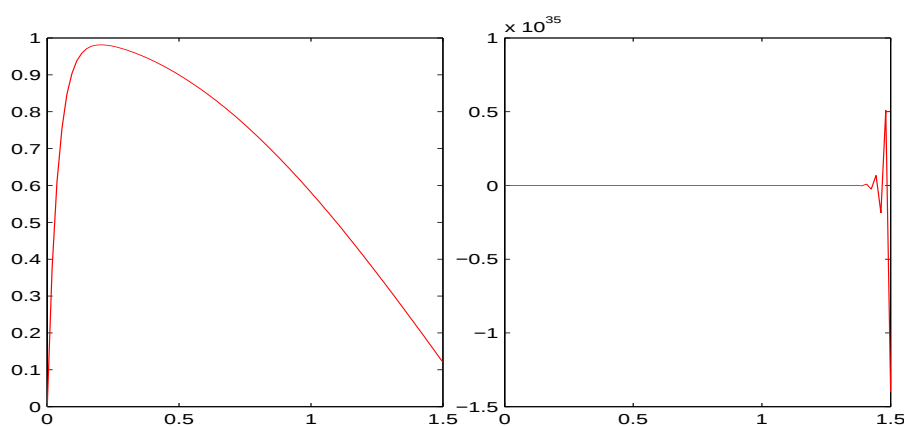
kde $z = \lambda h$.

a) Dle Definice 6 je numerická metoda absolutně stabilní, jestliže $|R(z)| \leq 1$.
Pro $\lambda = -20$ je $R(z) = 0.625$ a tedy platí, že $|R(z)| \leq 1$, proto je metoda pro tyto hodnoty absolutně stabilní.

Pro $\lambda = -200$ je $R(z) = -2.75$ a tedy neplatí, že $|R(z)| \leq 1$, proto není metoda pro tyto hodnoty absolutně stabilní.

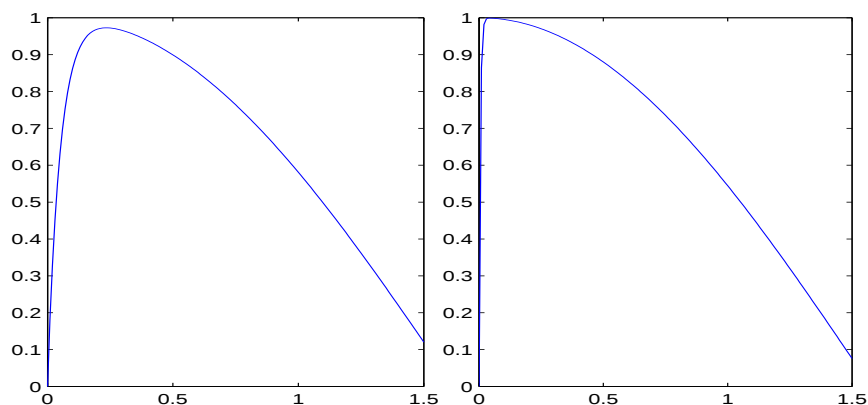
b) Dle Definice 6 je numerická metoda relativně stabilní, jestliže $|R(z)| \leq e^z$.
Pro $\lambda = -20$ je $R(z) = 0.625$ a e^z přibližně rovno 0.6873, a tedy je splněna podmínka relativní stability.

Pro $\lambda = -200$ je $R(z) = -2.75$ a e^z přibližně rovno 0.0235, a tedy podmínka relativní stability splněna není.



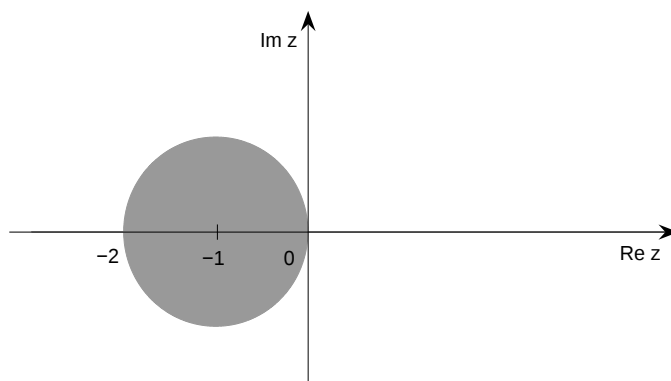
Obrázek 2: vlevo přibližné řešení pro $\lambda = -20$, vpravo pro $\lambda = -200$

Na Obrázku 2 vlevo vidíme nalezené přibližné řešení pomocí dané metody pro $\lambda = -20$. Na Obrázku 2 vpravo je znázorněno řešení nalezené pomocí zadané numerické metody, které je však odlišné od přesného řešení dané úlohy (viz Obrázek 3 vpravo) a to kvůli tomu, že daná numerická metoda není pro $\lambda = -200$ absolutně ani relativně stabilní. Pro srovnání uvádím řešení zadané úlohy vypočítané analyticky pro jednotlivé hodnoty λ .



Obrázek 3: vlevo přesné řešení pro $\lambda = -20$, vpravo pro $\lambda = -200$

- c) Dle Definice 7 určíme oblast absolutní stability vyřešením nerovnice tvaru $|1 + z| \leq 1$, tj. pro danou metodu je $S = \{z \in \mathbb{C}; |1 + z| \leq 1\}$, která je znázorněna na Obrázku 4 šedou výplní.



Obrázek 4

Správnost výsledků z a) můžeme zkontrolovat tak, že ověříme, zda hodnota z pro daná λ leží v S : pro $\lambda = -20$ je $z = -0.375$, tedy $z \in S$, pro $\lambda = -200$ je $z = -3.75$ a tedy $z \notin S$.

- d) Dle Definice 8 je metoda A - stabilní, jestliže S obsahuje celou levou polovinu množiny komplexních čísel $\mathbb{C}$, což v tomto případě neplatí, takže zadaná metoda není A - stabilní.
- e) Jelikož zadaná metoda není A - stabilní, tak z Definice 9 plyne, že není ani L - stabilní.



Poznámka 2. Analýza stability nám také pomáhá určit, jak velký, resp. malý integrační krok máme volit, abychom dostali přibližné řešení blízké přesnému řešení. Například z Příkladu 1 víme, že oblast absolutní stability je pro danou metodu rovna $S = \{z \in \mathbb{C}; |1 + z| \leq 1\}$. Museli bychom tedy integrační krok h volit tak, aby $\lambda h \in S$. To znamená, aby pro $\lambda = -20$ byl $h \leq 0.1$, což je splněno, neboť $\frac{1.5}{80} < 0.1$. Pro $\lambda = -200$ je však požadováno, aby $h \leq 0.01$, což splněno není.

Poznámka 3. Uvažovaná metoda z Příkladu 1 se nazývá Eulerova explicitní metoda.

Jak už jsem dříve poznamenala, nejjednodušší obecnou jednokrokovou metodou je Eulerova metoda, kterou se nyní budeme podrobněji zabývat.

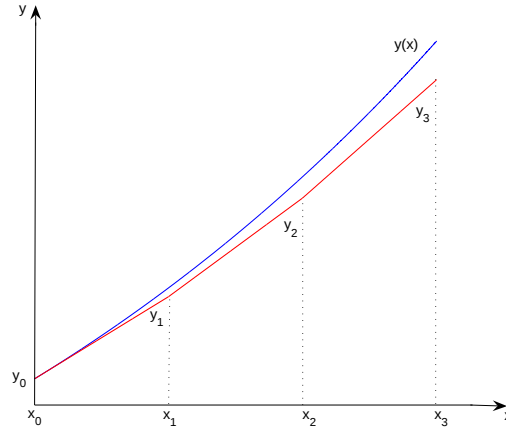
2.2 Eulerova metoda

Pro $N \in \mathbb{N}$ zvolme síť $x_0 < x_1 < \dots < x_N = b$ s krokem $h = \frac{b-x_0}{N}$. Eulerovou metodou získáváme pro $n = 0, \dots, N-1$ posloupnost přibližných hodnot hledaného řešení počáteční úlohy (1), (2) pomocí rekurentního vztahu

$$y_{n+1} = y_n + hf(x_n, y_n), \quad (6)$$

přičemž klademe $y_0 = a$. Eulerova metoda tedy opravdu patří mezi **jednokrokové** metody.

Spojíme-li body o souřadnicích $[x_n, y_n]$ lomenou čarou, dostaneme tzv. **Eulerův polygon**, který aproximuje hledané řešení $y(x)$ viz Obrázek 5.



Obrázek 5: modrá: přesné řešení počáteční úlohy (1), (2); červená: Eulerův polygon

Eulerovu metodu je možné také získat tak, že hodnotu přesného řešení $y(x_{n+1})$, $y(x_{n+1}) = y(x_n + h)$, aproximujeme pomocí Taylorova polynomu prvního stupně pro funkci y v bodě x_n , tj.

$$y(x_{n+1}) \approx y(x_n) + hy'(x_n) = y(x_n) + hf(x_n, y(x_n)),$$

což je rekurentní vztah Eulerovy metody.

Lokální diskretizační chybou Eulerovy metody na intervalu $\langle x_n, x_{n+1} \rangle$ je dle Definice 4 výraz, který je dán vztahem

$$L(y(x); h) = y(x_{n+1}) - y(x_n) - hf(x_n, y(x_n)).$$

Napíšeme-li Taylorův rozvoj řešení $y(x)$ v bodě x_{n+1} druhého řádu, tj.

$$y(x_{n+1}) = y(x_n) + hf(x_n, y(x_n)) + \frac{h^2 y''(\xi)}{2},$$

kde $\xi \in \langle x_n, x_{n+1} \rangle$, pak porovnáním s výrazem $L(y(x); h)$ dostáváme, že chyba, které se dopustíme v jednom kroku Eulerovy metody je také dána tvarem

$$L(y(x); h) = \frac{h^2 y''(\xi)}{2}. \quad (7)$$

Tedy z Definice 5 a ze vztahu (7) ihned vidíme, že Eulerova metoda je metoda prvního řádu.

Jednou ze základních vlastností, kterou u numerické metody sledujeme je konvergence. Zformulujeme tedy konvergenční větu pro Eulerovu metodu:

Věta 1. *Nechť platí Předpoklad 1 a nechť pro přesné řešení $y(x)$ rovnice (1) určené počáteční podmínkou (2) platí $y(x) \in C^2\langle x_0, b \rangle$. Nechť*

$$M(x) = \frac{1}{2} \max_{t \in \langle x_0, x \rangle} |y''(t)|.$$

Nechť dále posloupnost $\{\tilde{y}_n\}$, $n = 0, \dots, N$ je definovaná rekurentně

$$\begin{aligned}\tilde{y}_0 &= a \\ \tilde{y}_{n+1} &= \tilde{y}_n + hf(x_n, \tilde{y}_n) + \theta_n Ch^2, \quad n = 0, \dots, N-1,\end{aligned}$$

kde C je konstanta, h je integrační krok a θ_n jsou libovolná reálná čísla, pro něž platí $|\theta_n| \leq 1$ pro $n = 0, \dots, N-1$. Pak platí

$$|\tilde{y}_n - y(x_n)| \leq h[M(x_n) + C]E_L(x_n - x_0), \quad n = 0, \dots, N,$$

kde N je největší přirozené číslo, pro něž je $x_N \in \langle x_0, b \rangle$ a funkce E_L je definována takto:

$$E_L(x) = \begin{cases} \frac{e^{Lx}-1}{L} & \text{pro } L > 0, \\ x & \text{pro } L = 0. \end{cases}$$

Důkaz: viz [9] str. 26.

Tato věta říká, že k přesnému řešení dané počáteční úlohy konverguje nejen přibližné řešení získané Eulerovou metodou, ale i řešení, které vznikne tak, že se v každém kroku Eulerovy metody dopustíme chyby o velikosti řádu h^2 .

Oblast stability

Funkce stability pro Eulerovu metodu má tvar

$$R(z) = 1 + z,$$

jak jsme odvodili v Příkladu 1. Oblast stability je dle Definice 7 dána vztahem $S = \{z \in \mathbb{C}; |R(z)| \leq 1\}$, tedy $S = \{z \in \mathbb{C}; |1 + z| \leq 1\}$. Jelikož S neobsahuje

celou levou polovinu množiny $\mathbb{C}$, není Eulerova metoda dle Definic 8 a 9 ani A - stabilní, ani L - stabilní.

Dále si ukážeme praktický příklad, který vyřešíme pomocí Eulerovy metody. Obecné zadání tohoto příkladu je převzato ze skript [7]. Abychom mohli tuto úlohu vyřešit Eulerovou metodou, je potřeba zadat konkrétní hodnoty. Hmotnost kapky, kapající z kohoutku, je orientačně zvážena na kuchyňské váze.

Příklad 2. *Pád tělesa proměnné hmotnosti:* Kapka vody s počáteční hmotností M volně padá vzduchem, stejnoměrně se odpařuje, každou vteřinu ztrácí hmotnost m . Síla odporu vzduchu je úměrná rychlosti pádu kapky. Najděte závislost rychlosti pádu kapky na čase, jestliže v počátečním čase pádu ($t = 0$) byla rychlost kapky rovna nule. Řešte nejprve analyticky a poté Eulerovou metodou s konkrétními hodnotami $k = 0.1$, $M = 0.1575\text{g}$, $m = 0.01\text{g}$ a $g = 981\text{cm/s}^2$.

Řešení: V důsledku rovnoměrného odpařování kapky je její hmotnost v čase t rovna $M - mt$ a tíha kapky je $(M - mt)g$, kde g je tíhové zrychlení. Síla odporu vzduchu je úměrná rychlosti pádu kapky, její velikost je tedy kv , kde o koeficientu úměrnosti k předpokládáme, že $k \neq m$. Výslednice sil F působící na kapku vody je

$$F = (M - mt)g - kv.$$

Dále podle 2. Newtonova zákona platí

$$F = (M - mt)v'.$$

Tedy diferenciální rovnice popisující danou úlohu má tvar

$$v' + \frac{k}{M - mt}v = g, \quad (8)$$

což je nehomogenní lineární diferenciální rovnice 1. řádu pro rychlost $v(t)$.

Tuto diferenciální rovnici vyřešíme nejprve analyticky a poté pomocí Eulerovy metody.

1. Analytické řešení

- a) Nejprve vyřešíme homogenní diferenciální rovnici příslušnou rovnici (8), tedy řešíme rovnici

$$v' + \frac{k}{M - mt}v = 0.$$

Metodou separace proměnných dostáváme

$$\frac{dv}{v} = -\frac{k}{M - mt}dt$$

a integrací

$$\ln|v| = \frac{k}{m}\ln|M - mt| + \ln|c_1|, \quad c_1 \in \mathbb{R}, c_1 \neq 0$$

Po úpravě máme

$$v = K(M - mt)^{\frac{k}{m}}, \quad K \in \mathbb{R}.$$

- b) Nyní metodou variace konstanty nalezneme řešení nehomogenní diferenciální rovnice (8), tj.

$$v(t) = K(t)(M - mt)^{\frac{k}{m}}, \quad (9)$$

kde $K(t)$ je neznámá funkce. Dosazením do rovnice (8) zjistíme, že

$$K'(t) = g(M - mt)^{-\frac{k}{m}}$$

a poté integrací, že

$$K(t) = \frac{g}{k - m} \left(M - mt \right)^{1 - \frac{k}{m}} + T, \quad T \in \mathbb{R}.$$

Dosazením do vztahu (9) dostaneme obecné řešení rovnice (8), které je tvaru

$$v(t) = T \left(M - mt \right)^{\frac{k}{m}} + \frac{g}{k - m} \left(M - mt \right), \quad T \in \mathbb{R}. \quad (10)$$

- c) Konečně dosazením počáteční podmínky, že v čase $t = 0$ je rychlost $v = 0$, tj. $v(0) = 0$, do vztahu (10) zjistíme, že

$$T = \frac{g}{m-k} M^{1-\frac{k}{m}}.$$

Dosadíme-li toto T do vztahu (10), dostaneme po menších úpravách hledanou závislost rychlosti pádu kapky na čase

$$v(t) = \frac{g}{m-k} \left(mt - M + M \left(1 - \frac{m}{M} t \right)^{\frac{k}{m}} \right), \quad (11)$$

kde $t \in \langle 0, \frac{M}{m} \rangle$.

- d) Dosazením konkrétních hodnot za neznámé k, M, m a g dostaneme funkci

$$v(t) = -109t + \frac{6867}{4} - \frac{6867}{4} \left(1 - \frac{4}{63} t \right)^{10}, \quad (12)$$

kde $t \in \langle 0, 15.75 \rangle$.

2. Dosazením konkrétních hodnot za neznámé do rovnice (8) dostaneme počáteční úlohu

$$v'(t) = 981 - \frac{v(t)}{1.575 - 0.1t}, \quad (13)$$

$$v(0) = 0.$$

Pomocí Eulerovy metody s integračním krokem $h = 0.1575$ aplikovanou na tuto počáteční úlohu, která je tvaru

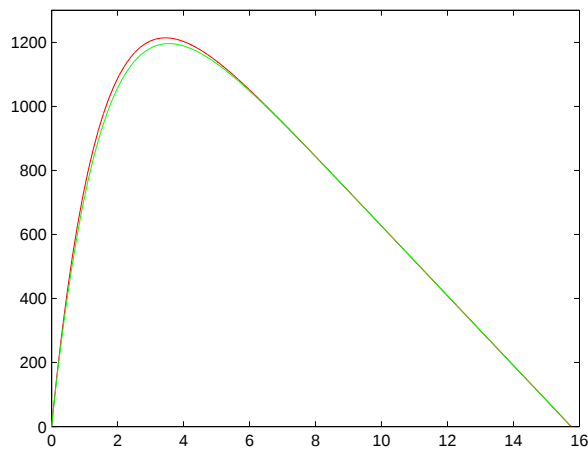
$$y_{n+1} = y_n + h \left(981 - \frac{y_n}{1.575 - 0.1t_n} \right)$$

dostaneme na intervalu $\langle 0, 15.75 \rangle$ posloupnost $\{y_n\}$ přibližných hodnot hledaného řešení, které jsou zaznamenány v následující tabulce. V této tabulce uvádím také pro srovnání hodnoty přesného řešení z d) v diskrétní množině bodů t_n .

n	t_n	$v(t_n)$	y_n	$y_n - v(t_n)$
0	0.0000	0.0000	0.0000	0.0000
1	0.1575	146.9846	154.5075	7.5229
2	0.3150	279.7053	293.4082	13.7029
3	0.4725	399.2746	417.9761	18.7015
4	0.6300	506.7286	529.3933	22.6647
$\vdots$				
96	15.1200	68.6700	68.6700	0.0000
97	15.2775	51.5025	51.5025	0.0000
98	15.4350	34.3350	34.3350	0.0000
99	15.5925	17.1675	17.1675	0.0000
100	15.7500	0.0000	0.0000	0.0000

Tabulka 1

Z tabulky vidíme, že nejprve lokální diskretizační chyba Eulerovy metody roste, pak se ale zmenšuje, což je znázorněno na Obrázku 6, kde je zelenou barvou znázorněn graf přesného řešení Cauchyovy úlohy (13) a červeně Eulerův polygon této úlohy.



Obrázek 6

Tento příklad byl vyřešen pomocí mnou sestaveného M-souboru Eulerovy metody, a to následovně: do matematického softwaru Matlab bylo zadáno

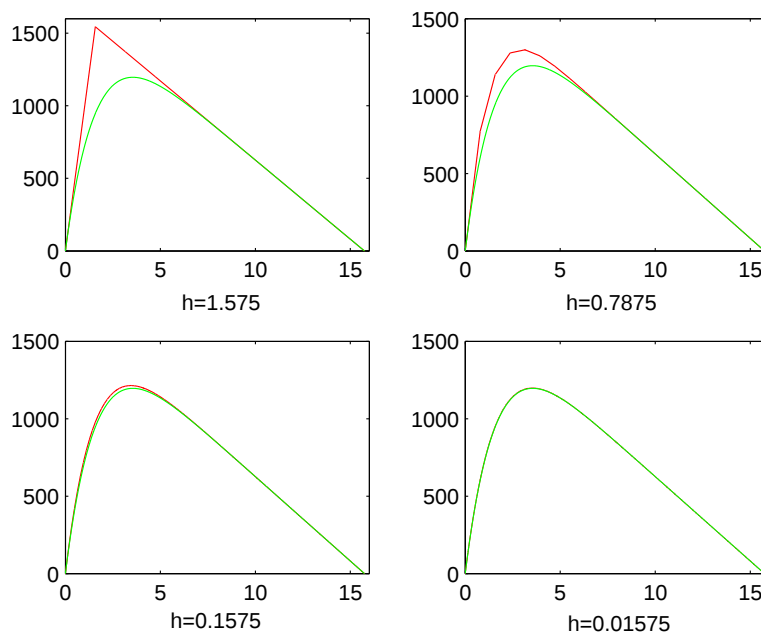
```

k=0.1;M=0.1575;m=0.01;g=981;
a=0;b=M/m;ni=0;h=0.1575;
f=@(x,y)g-k*y/(M-m*x);
[y]=Euler(a,b,ni,f,h);

```

M-soubor Eulerovy metody je uveden v odstavci 4. 1.

Na závěr tohoto příkladu ještě uvedu pro srovnání obrázky, jak se Eulerův polygon (znázorněn červeně) liší od přesného řešení této úlohy (znázorněno zeleně) v závislosti na volbě délky kroku, viz Obrázek 7.



Obrázek 7



2.3 Metody Runge–Kutta

Další metody, které patří mezi obecné jednokrokové metody, jsou tzv. metody Runge–Kutta.

V obecných metodách Runge–Kutta je funkce Φ z rovnice (3) dána předpisem

$$\Phi(x_n, y_n, h) = w_1 k_1 + \dots + w_s k_s,$$

kde

$$\begin{aligned} k_1 &= f(x_n, y_n), \\ k_i &= f\left(x_n + \alpha_i h, y_n + h \sum_{j=1}^{i-1} \beta_{ij} k_j\right), \quad i = 2, \dots, s, \end{aligned} \quad (14)$$

kde $w_1, \dots, w_s, \alpha_i, \beta_{ij}, i = 2, \dots, s, j = 1, \dots, i-1$, jsou konstanty.

Pro získání řešení počáteční úlohy (1), (2) pomocí metod Runge–Kutta postupujeme tak, že pro $N \in \mathbb{N}$ zvolíme síť $x_0 < x_1 < \dots < x_N = b$ s krokem $h = \frac{b-x_0}{N}$. Dále pro $n = 0, \dots, N-1$ užijeme rekurentní vztah

$$y_{n+1} = y_n + h \sum_{i=1}^s w_i k_i, \quad (15)$$

přičemž $y_0 = a$. Konstanty $w_1, \dots, w_s$ nazýváme koeficienty metody Runge–Kutta.

V každém kroku metody nejprve postupně spočítáme $k_1, \dots, k_s$, které nazýváme stádia metody Runge–Kutta, a potom podle vztahu (15) vypočteme hodnotu přibližného řešení y_{n+1} v bodě x_{n+1} .

Počítáme-li stádia v metodě Runge–Kutta ze vztahu (14), jedná se o tzv. **explicitní metodu Runge–Kutta**. Počítáme-li je z následujícího vztahu

$$k_i = f\left(x_n + \alpha_i h, y_n + h \sum_{j=1}^i \beta_{ij} k_j\right), \quad i = 1, \dots, s, \quad (16)$$

jedná se o tzv. **implicitní metodu Runge–Kutta**.

Metody Runge–Kutta se dají také zapsat jiným způsobem, dle Butchera, a to tak, že všechny konstanty, které se objevují ve vztazích (15), (14), respektive (15), (16), se dají zapsat do následující tabulky:

α_1	β_{11}	$\cdots$	β_{1s}
α_2	β_{21}	$\cdots$	β_{2s}
$\vdots$	$\vdots$		$\vdots$
α_s	β_{s1}	$\cdots$	β_{ss}
	w_1	$\cdots$	w_s

Tabulka 2

Metod Runge–Kutta je mnoho, my se budeme zabývat nejužívanější z nich, kterou budeme nazývat **klasická metoda Runge–Kutta**. Tato metoda je řádu 4 a má následující tvar:

$$y_{n+1} = y_n + h \frac{k_1 + 2k_2 + 2k_3 + k_4}{6},$$

kde

$$k_1 = f(x_n, y_n),$$

$$k_2 = f\left(x_n + \frac{h}{2}, y_n + \frac{h}{2}k_1\right),$$

$$k_3 = f\left(x_n + \frac{h}{2}, y_n + \frac{h}{2}k_2\right),$$

$$k_4 = f(x_n + h, y_n + hk_3).$$

Zápis klasické metody Runge–Kutta dle Butchera:

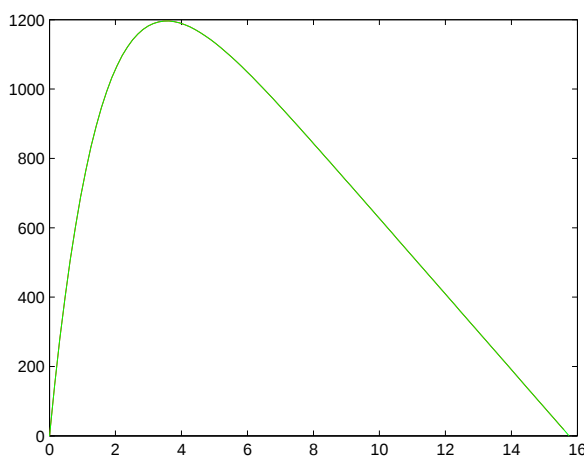
0	0	0	0	0
$\frac{1}{2}$	$\frac{1}{2}$	0	0	0
$\frac{1}{2}$	0	$\frac{1}{2}$	0	0
1	0	0	1	0
	$\frac{1}{6}$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{6}$

Příklad 3. Vyřešte Příklad 2 pomocí klasické metody Runge–Kutta.

Řešení: Tento příklad opět vyřešíme pomocí Matlabu a to pomocí mnou sestaveného M-souboru klasické metody Runge–Kutta, který je uveden v odstavci 4.2. Zadáme do Matlabu

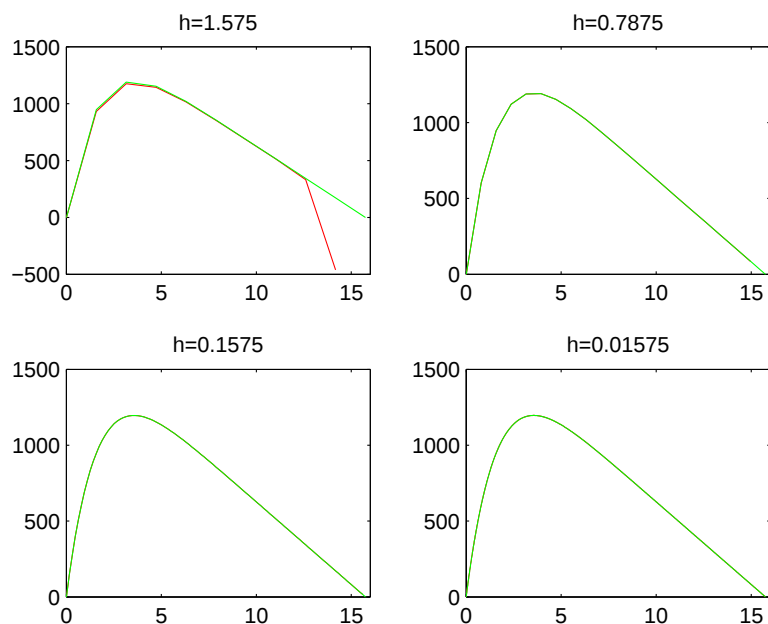
```
k=0.1;M=0.1575;m=0.01;g=981;  
a=0;b=M/m;ni=0;h=0.1575;  
f=@(x,y)g-k*y/(M-m*x);  
[y]=RungeKutta4(a,b,ni,f,h);
```

S tímto krokem $h = 0.1575$ dostáváme přesnější výsledky než při použití Eulerovy metody, což vidíme na následujícím obrázku, kde je znázorněno přesné řešení dané úlohy červeně a nalezené přibližné řešení zeleně (zelená překryje červenou).



Obrázek 8

Na závěr tohoto příkladu na Obrázku 9 opět uvedu, jak se hledané přibližné řešení (znázorněno červeně) liší od přesného (znázorněno zeleně) v závislosti na volbě velikosti délky integračního kroku:



Obrázek 9



Poznámka 4. Při ověřování stability pro nelineární úlohy, tj. úlohy, kdy pravá strana diferenciální rovnice (1) je nelineární funkce, nastává problém při sestavování funkce stability $R(z)$ z kapitoly 2. 1, protože v tomto případě je zde nutné znát řešení zadané úlohy, které ale ovšem hledáme. Hledalo se tedy zobecnění A - stability. Zrodil se tak koncept B - stability pro implicitní metody Runge–Kutta.

Definice 10. Implicitní metoda Runge–Kutta (15), (16) je **B - stabilní**, jestliže podmínka kontraktivity

$$\langle f(x, y) - f(x, z), y - z \rangle \leq 0$$

implikuje pro všechny kroky $h \geq 0$ nerovnost

$$|y_1 - \bar{y}_1| \leq |y_0 - \bar{y}_0|,$$

kde y_1 , respektive $\bar{y}_1$, je druhý člen posloupnosti přibližného řešení rovnice (1) získané pomocí obecné jednokrokové metody s krokem h a s počáteční podmínkou $y(x_0) = y_0$, respektive $y(x_0) = \bar{y}_0$.

Poznámka 5. Ověřování podmínky kontraktivity z předchozí definice bývá v praxi obtížné, hledají se tedy jiné způsoby ověření B - stability dané metody. Následující věta nám pomůže B - stabilitu ověřit.

Věta 2. *Jestliže koeficienty implicitní metody Runge–Kutta splňují následující podmínky*

$$i) \quad w_i \geq 0 \text{ pro } i = 1, \dots, s,$$

$$ii) \quad \text{matice } M = (m_{ij})_{i,j=1}^s, \text{ kde } m_{ij} = w_i \beta_{ij} + w_j \beta_{ji} - w_i w_j, \text{ je pozitivně semi-definitní,}$$

pak je metoda B - stabilní.

Důkaz: Viz [4] str. 182.

Poznámka 6. Mezi těmito dvěma extrémy: teorií A - stability pro autonomní lineární úlohy $y' = \lambda y, \lambda \in \mathbb{C}$, a teorií B - stability pro obecné nelineární úlohy $y' = f(x, y)$ se nachází teorie AN - stability pro studium skalární, lineární, neautonomní úlohy $y'(x) = \lambda(x) y(x), \operatorname{Re} \lambda(x) \leq 0$.

Definice 11. Implicitní metoda Runge–Kutta je **AN - stabilní**, jestliže

$$|K(Z)| \leq 1 \quad \begin{cases} \text{pro všechna } Z = \operatorname{diag}(z_1, \dots, z_s) \text{ splňující } \operatorname{Re} z_j \leq 0 \\ \text{a } z_j = z_k \text{ kdykoliv } \alpha_j = \alpha_k \quad (j, k = 1, \dots, s), \end{cases}$$

kde s je počet stádií dané metody a dále

$$K(Z) = 1 + \mathbf{w} Z (I - AZ)^{-1} \mathbf{v},$$

$$\mathbf{w} = (w_1, \dots, w_s),$$

$$\mathbf{v} = (1, \dots, 1)^T,$$

$$z_j = h\lambda(x_0 + \alpha_j h),$$

$$A = (\beta_{ij})_{i,j=1}^s,$$

$$I = \operatorname{diag}(1, \dots, 1),$$

přičemž $\mathbf{v}$ je typu $s \times 1$ a I má rozměr $s \times s$.

Pro srovnání A - stability, B - stability a AN - stability platí následující věta, která je převzata z literatury [4].

Věta 3. *Pro implicitní metody Runge–Kutta platí*

$$B - \text{stabilita} \implies AN - \text{stabilita} \implies A - \text{stabilita}.$$

Příklad 4. Vyšetřete B-stabilitu a AN-stabilitu implicitní metody Runge–Kutta dané vztahem

$$y_{n+1} = y_n + hf(x_{n+1}, y_{n+1})$$

pro úlohu

$$y'(x) = \lambda(y(x) - \cos(x)), \lambda \in \mathbb{R}$$

$$y(0) = 0$$

na intervalu $\langle 0, 1.5 \rangle$ s krokem $h = 1.5/80$ pro $\lambda = -20$ a $\lambda = -200$.

Řešení:

- a) B - stabilitu vyšetříme pomocí Věty 2. Z tvaru zadané metody vidíme, že $w_1 = 1$ a $\beta_{11} = 1$. Tedy $w_1 > 0$ a podmínka *i*) z Věty 2 je splněna. Sestavíme matici M . Jelikož se daná metoda dá zapsat v následujícím tvaru

$$y_{n+1} = y_n + hk_1,$$

$$k_1 = f(x_n + h, y_n + hk_1),$$

vidíme, že počet stádií $s = 1$, tedy $i, j = 1$. Matice M bude mít tedy rozměr 1×1 . Dle uvažované věty tedy $M = w_1\beta_{11} + w_1\beta_{11} - w_1w_1 = 1$, odkud vidíme, že je podmínka *ii*) také splněna. A tudíž daná metoda je opravdu B - stabilní.

- b) AN - stabilitu vyšetříme pomocí Definice 11. Nejprve musíme sestavit funkci $K(Z)$: Ze vztahu zadané metody Runge–Kutta vidíme, že $\alpha_1 = 1$, $\beta_{11} = 1$ a $w_1 = 1$. Matice Z má takovou hodnotu, kolik má uvažovaná metoda stádií,

v našem případě je to 1. Tedy $Z = z_1$, přičemž $z_1 = h\lambda(x_0 + \alpha_1 h) = h\lambda$.
Dále $A = \beta_{11}$ a $I = 1$. Tedy funkce $K(Z)$ má tvar

$$K(Z) = 1 + 1z_1(1 - 1z_1)^{-1}1 = 1 + \frac{z_1}{1 - z_1} = \frac{1}{1 - z_1},$$

kde $z_1 = h\lambda$.

Pro $\lambda = -20$ je $z_1 = -\frac{3}{8}$, odkud vidíme, že $\operatorname{Re} z_1 < 0$ a $|K(Z)| = \frac{8}{11} < 1$.

Pro $\lambda = -200$ je $z_1 = -\frac{15}{4}$, odkud vidíme, že $\operatorname{Re} z_1 < 0$ a $|K(Z)| = \frac{4}{19} < 1$.

Tedy zadaná úloha je pro danou metodu, zadaný krok h a obě hodnoty λ AN-stabilní.



Poznámka 7. Uvažovaná metoda z Příkladu 4 se nazývá Eulerova implicitní metoda. Eulerova implicitní metoda je speciálním a nejjednodušším případem implicitních metod Runge–Kutta. Je to jednostádiová ($s = 1$) metoda Runge–Kutta s koeficienty $w_1 = 1$, $\alpha_1 = 1$ a $\beta_{11} = 1$.

Poznámka 8. Funkce $K(Z)$ z Příkladu 4 je funkce jedné proměnné z_1 . Obecně je však funkcí s proměnných $z_1, \dots, z_s$. Na následujícím příkladě ukáží, jak tuto funkci sestavit pro metodu Runge–Kutta se dvěma stádii.

Příklad 5. Vyšetřete AN - stabilitu implicitní metody Runge–Kutta dané vztahem

$$\begin{aligned} y_{n+1} &= y_n + \frac{h}{2}(k_1 + k_2), \\ k_1 &= f(x_n + h, y_n + hk_1), \\ k_2 &= f(x_n + 2h, y_n - hk_1 + 3hk_2), \end{aligned}$$

pro úlohu

$$\begin{aligned} y'(x) &= C x y(x), \quad C \in \mathbb{R} \\ y(0) &= 0 \end{aligned}$$

na intervalu $\langle 0, 1.5 \rangle$ s krokem $h = 1.5/80$ pro $C = -20$.

Řešení: AN - stabilitu vyšetříme opět pomocí Definice 11. V tomto případě je $\lambda(x) = Cx$. Nejprve sestavíme funkci $K(Z)$: Ze vztahu zadané metody Runge–Kutta vidíme, že $\alpha_1 = 1$, $\alpha_2 = 2$, $\beta_{11} = 1$, $\beta_{12} = 0$, $\beta_{21} = -1$, $\beta_{22} = 3$ a $w_1 = \frac{1}{2}$, $w_2 = \frac{1}{2}$. Matice Z má v tomto případě hodnotu 2, protože $s = 2$. Tedy

$$Z = \begin{pmatrix} z_1 & 0 \\ 0 & z_2 \end{pmatrix},$$

přičemž $z_1 = h\lambda(x_0 + \alpha_1 h) = h\lambda(x_0 + h) = Ch(x_0 + h)$ a $z_2 = h\lambda(x_0 + \alpha_2 h) = h\lambda(x_0 + 2h) = Ch(x_0 + 2h)$. Dále

$$A = \begin{pmatrix} \beta_{11} & \beta_{12} \\ \beta_{21} & \beta_{22} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ -1 & 3 \end{pmatrix}$$

a $I = \text{diag}(1, 1)$. Tedy funkce $K(Z)$ má tvar

$$K(Z) = 1 + \left(\frac{1}{2}, \frac{1}{2}\right) \begin{pmatrix} z_1 & 0 \\ 0 & z_2 \end{pmatrix} \left[\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} - \begin{pmatrix} 1 & 0 \\ -1 & 3 \end{pmatrix} \begin{pmatrix} z_1 & 0 \\ 0 & z_2 \end{pmatrix} \right]^{-1} \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

Po několika úpravách dostaneme, že

$$K(Z) = \frac{2 - 5z_2 - z_1 + z_1 z_2}{2(1 - z_1)(1 - 3z_2)},$$

kde $z_1 = Ch(x_0 + h)$ a $z_2 = Ch(x_0 + 2h)$.

Pro $C = -20$ je $z_1 = -\frac{9}{1280}$, $z_2 = -\frac{9}{640}$ odkud vidíme, že $\text{Re } z_1 < 0$, $\text{Re } z_2 < 0$ a $|K(Z)| = \frac{1701841}{1719526} < 1$.

Tedy pro daný krok h je zadaná metoda AN - stabilní.

2.4 Obecná jednokroková metoda pro soustavu diferenciálních rovnic

V tomto odstavci budeme hledat numerické řešení soustavy diferenciálních rovnic

$$y_1'(x) = f_1(x, y_1(x), \dots, y_m(x)), \quad (17)$$

$$y_2'(x) = f_2(x, y_1(x), \dots, y_m(x)), \quad (18)$$

$\vdots$

$$y_m'(x) = f_m(x, y_1(x), \dots, y_m(x))$$

s počátečními podmínkami

$$y_1(x_0) = a_1, \quad (19)$$

$$y_2(x_0) = a_2, \quad (20)$$

$$\vdots$$

$$y_m(x_0) = a_m,$$

což se dá vektorově zapsat

$$\mathbf{y}'(x) = \mathbf{f}(x, \mathbf{y}(x)), \quad (21)$$

$$\mathbf{y}(x_0) = \mathbf{a}, \quad (22)$$

kde $\mathbf{y} = (y_1, \dots, y_m)^T$, $\mathbf{f} = (f_1, \dots, f_m)^T$ a $\mathbf{a} = (a_1, \dots, a_m)^T$.

Hledáme-li řešení této soustavy s počátečními podmínkami, tedy řešení úlohy (21), (22), pomocí obecné jednokrokové metody, postupujeme tak, že tuto metodu s daným krokem h aplikujeme na jednotlivé diferenciální rovnice uvažované soustavy s příslušnými počátečními podmínkami. Konkrétněji, opět zvolíme ekvidistantní síť $\{x_n\}$, kde $x_n = x_0 + n h$, určenou krokem h , na které řešení hledáme a dále postupně v každém bodě této sítě vyřešíme obecnou jednokrokovou metodu s krokem h všechny diferenciální rovnice s příslušnými počátečními podmínkami, tedy úlohu (17), (19), dále úlohu (18), (20), atd. Tento postup si ukážeme na Příkladě 6.

Poznámka 9. M-soubor obecné jednokrokové metody pro řešení soustavy diferenciálních rovnic s počátečními podmínkami se trochu změní. Jak se změní je uvedeno na příkladě Eulerovy explicitní metody v odstavci 4.3.

Příklad 6. Najděte spojitou funkci $y : I \rightarrow \mathbb{R}^2$ takovou, že platí

$$y_1'(x) = -y_2(x) + 0.1 y_1(x)(9 - y_1(x)^2 - y_2(x)^2) \quad (23)$$

$$y_2'(x) = y_1(x) + 0.1 y_2(x)(9 - y_1(x)^2 - y_2(x)^2) \quad (24)$$

pro všechna $x \in I = \langle 0, 6 \rangle$ a

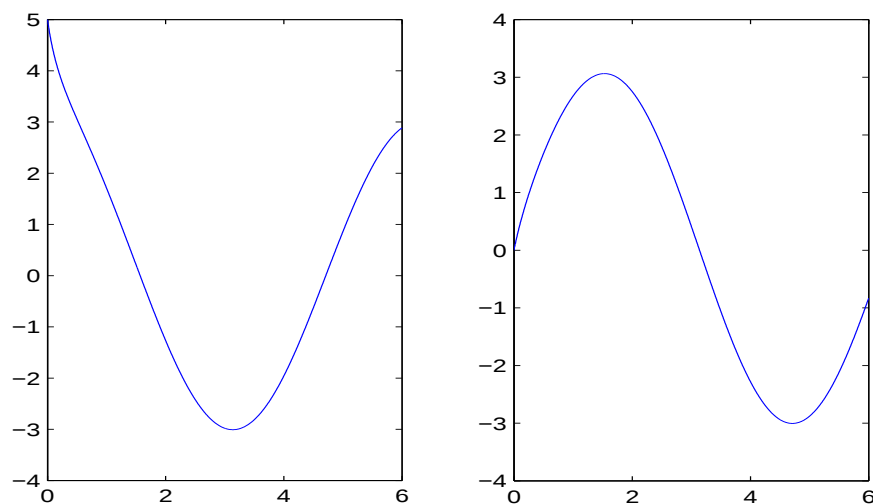
$$y_1(0) = 5, \quad y_2(0) = 0.$$

Řešte pomocí Eulerovy explicitní metody s krokem $h = 0.005$.

Řešení: Řešení najdeme pomocí Matlabu a pomocí M-souboru Eulerovy explicitní metody pro řešení soustavy diferenciálních rovnic s počátečními podmínkami, který je uveden v odstavci 4.3. Zadáme do Matlabu

```
f=@(x,y) [-y(2)+0.1*y(1)*(9-y(1)^2-y(2)^2);  
           y(1)+0.1*y(2)*(9-y(1)^2-y(2)^2)];  
[y]=EulerSoustavy(0,6,[5 0],f,0.005);
```

Nalezené řešení, tedy jednotlivé jeho složky, je znázorněno na Obrázku 10.



Obrázek 10: vlevo 1. složka řešení, vpravo 2. složka řešení



3 Diferenciální inkluze

Diferenciálními rovnicemi se dají popsat některé přírodní jevy, na všechny však nestačí, proto se zavádí pojem diferenciální inkluze. Příkladem je kyvadlo s kombinovaným suchým a viskózním tlumením.

Důležitost pojmu diferenciální inkluze ukážeme na následujícím motivačním příkladě:

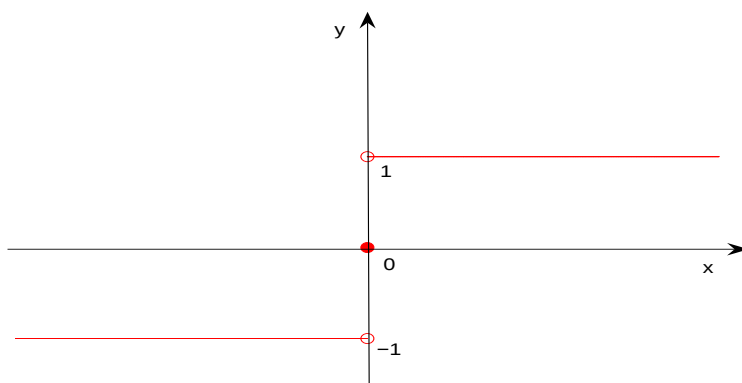
Příklad 7. Řešte diferenciální rovnici

$$y'(x) = -x + \operatorname{sgn}(x)$$

s počáteční podmínkou $y(-1) = 1$ na intervalu $\langle -1, 1 \rangle$.

Řešení: Pravá strana uvažované diferenciální rovnice je tvořena funkcí signum, což je nespojitá funkce v počátku souřadnicového systému, jak můžeme vidět na Obrázku 11. Je tedy porušen Předpoklad 1 z kapitoly 2 a nemůžeme tedy obecně očekávat existenci či jednoznačnost řešení.

$$\operatorname{sgn}(x) = \begin{cases} 1 & \text{pro } x > 0 \\ 0 & \text{pro } x = 0 \\ -1 & \text{pro } x < 0 \end{cases} \quad (25)$$

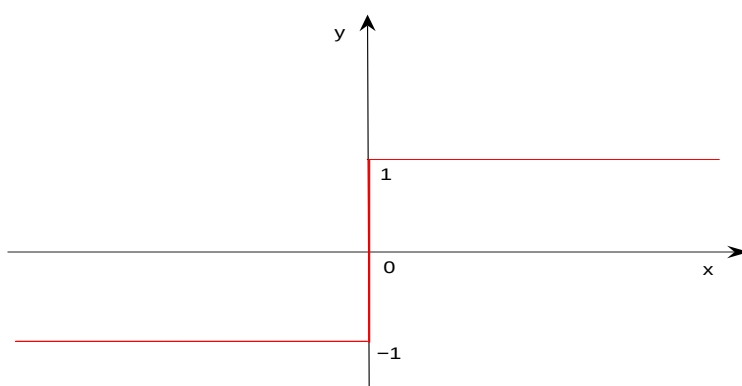


Obrázek 11

Tento problém nespojitosti můžeme vyřešit tím, že k funkci $sgn(x)$ zavedeme mnohoznačnou funkci $Sgn(x)$, a to následovně

$$Sgn(x) = \begin{cases} 1 & \text{pro } x > 0 \\ \langle -1, 1 \rangle & \text{pro } x = 0 \\ -1 & \text{pro } x < 0 \end{cases} \quad (26)$$

Zbavíme se tedy problému nespojitosti, vyvstane však jiný, a to mnohoznačnost v počátku souřadnicového systému, což lze vidět na Obrázku 12.

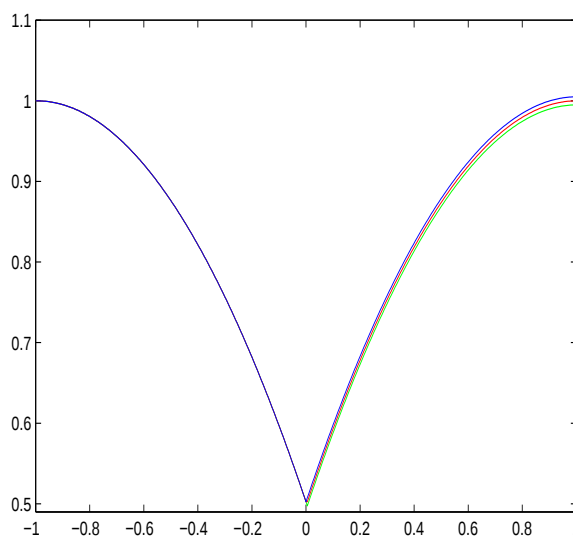


Obrázek 12

Zadanou počáteční úlohu s nespojitou pravou stranou tak převedeme na nový počáteční problém vyřešit diferenciální inkluzi

$$y'(x) \in -x + Sgn(x)$$

na intervalu $\langle -1, 1 \rangle$ s počáteční podmínkou $y(-1) = 1$. Díky spojitosti pravé strany můžeme očekávat existenci řešení této úlohy, jednoznačnost z důvodu mnohoznačnosti pravé strany však nikoli. Ve skutečnosti existuje celý svazek řešení v závislosti na tom, jakou hodnotu v počátku souřadnicového systému vybereme z intervalu $\langle -1, 1 \rangle$, jelikož těchto hodnot v tomto bodě mnohoznačná funkce $Sgn(x)$ nabývá. Na následujícím obrázku ukáži, jak se řešení mění v závislosti na výběru hodnoty funkce $Sgn(x)$ v počátku souřadnicového systému.



Obrázek 13

Na Obrázku 13 vidíme 3 různá řešení pro různé hodnoty mnohoznačné funkce $Sgn(x)$ v počátku souřadnicového systému: pro vybranou hodnotu $Sgn(0) = -1$ dostáváme řešení znázorněné zelenou barvou, pro $Sgn(0) = 0$ červené řešení a pro $Sgn(0) = 1$ modré. Vidíme, že na intervalu $\langle -1, 0 \rangle$ se nám řešení neliší, protože funkce $Sgn(x)$ je na tomto intervalu dána jednoznačně, ve zbylé části intervalu však ano, protože v bodě $x = 0$ nastává mnohoznačnost funkce $Sgn(x)$.



Tedy mějme počáteční úlohu (1), (2) s nespojitou pravou stranou. Při vyšetřování existence a jednoznačnosti řešení diferenciální rovnice je jedním ze základních požadavků na pravou stranu této rovnice spojitost. V opačném případě nelze obecně očekávat vůbec existenci hledaného řešení. Tyto případy jsou řešeny zavedením mnohoznačné funkce a následně řešením nikoli diferenciální rovnice, ale již diferenciální inkluze. Tento postup je uveden v Příkladě 7.

Nyní tento počáteční problém zformulujeme obecně.

V této kapitole se tedy budeme zabývat hledáním numerického řešení počá-

teční úlohy

$$y'(x) \in F(x, y(x)) \quad (27)$$

$$y(x_0) = y_0 \quad (28)$$

na konečném intervalu $I = \langle x_0, b \rangle$, kde $y_0 \in \mathbb{R}^m$ a $F : I \times \mathbb{R}^m \multimap \mathbb{R}^m$.

Poznámka 10. Symbol $\multimap$ je znak pro mnohoznačné zobrazení, viz odstavec 1.2.

Numerické řešení diferenciální inkluze lze nalézt pomocí numerických metod popsaných v kapitole 2, s malými obměnami. Jelikož se nám v inkluzi objevuje mnohoznačná funkce, nemůžeme očekávat jednoznačné řešení, naopak dostáváme celý svazek řešení. Je tedy důležitý výběr strategie, neboli volba další metody, která nám najde pouze jedno řešení. To si však ukážeme později.

3.1 Eulerova metoda pro diferenciální inkluze

Uvažujme počáteční úlohu (27), (28). Pro získání řešení této úlohy Eulerovou metodou postupujeme tak, že pro $N \in \mathbb{N}$ zvolíme síť $x_0 < x_1 < \dots < x_N = b$ s krokem $h = \frac{b-x_0}{N}$. Dále nechť $\eta_0 = y_0$ a pro $n = 0, \dots, N-1$ počítáme η_{n+1} ze vztahu

$$\eta_{n+1} \in \eta_n + hF(x_n, \eta_n). \quad (29)$$

Jak jsem již dříve poznamenala, řešíme-li inkluzi, dostaneme celou množinu řešení pro daný krok h , které splňují (29). Naším úkolem je pomocí vhodné strategie vybrat jedno z těchto řešení. Které z nich to bude, záleží právě na naší zvolené strategii. Různé strategie výběru si ukážeme později, na následujícím příkladě ukáži jednu z nich, a to strategii náhodného výběru, kdy v bodech, ve kterých funkce pravé strany diferenciální inkluze (29) nabývá více hodnot, vybereme jednu z těchto možných hodnot, a to náhodně. Zadání Příkladu 8 je převzato z článku [3].

Příklad 8. Najděte spojitou funkci $y : I \rightarrow \mathbb{R}^2$ takovou, že platí

$$y'_1(x) \in -y_2(x) + 0.1 y_1(x)(9 - y_1(x)^2 - y_2(x)^2) + \langle -1, 1 \rangle, \quad (30)$$

$$y'_2(x) \in y_1(x) + 0.1 y_2(x)(9 - y_1(x)^2 - y_2(x)^2) + \langle -1, 1 \rangle, \quad (31)$$

pro všechna $x \in I = \langle 0, 12 \rangle$ a

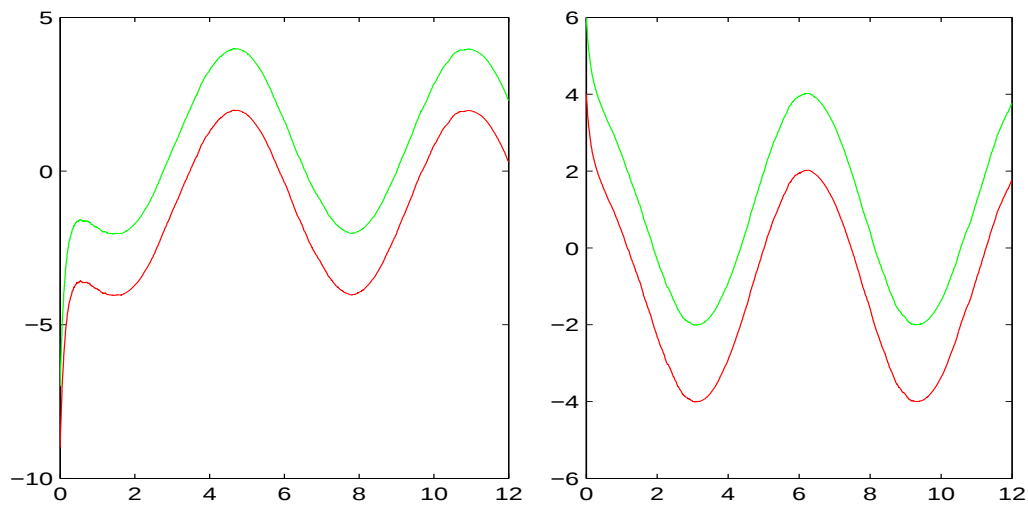
$$y_1(0) = 5, \quad y_2(0) = 0.$$

Řešení: Užijeme Eulerovu metodu s krokem $h = 0.005$. Řešení je mnoho, v tomto příkladě zvolíme strategii náhodného výběru. Tento příklad vyřeším pomocí mnou sestaveného M-souboru, který je uveden v odstavci 4. 4. Je zde povinné zadat hodnotu *parametr*, který nabývá hodnot 1-7 a udává, jakou strategií výběru jediného řešení chci diferenciální inkluzi vyřešit. V tomto případě zadám hodnotu 4, která odpovídá strategii náhodného výběru. Více viz odstavec 4.4.

Do Matlabu tedy zadám:

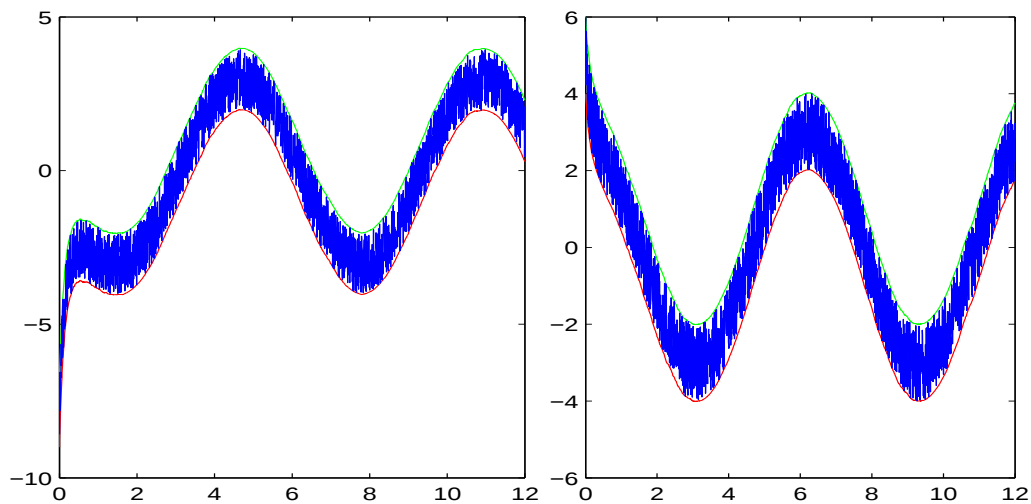
```
F.min=@(x,y)[-y(2)+0.1*y(1)*(9-y(1)^2-y(2)^2)-1;
              y(1)+0.1*y(2)*(9-y(1)^2-y(2)^2)-1];
F.max=@(x,y)[-y(2)+0.1*y(1)*(9-y(1)^2-y(2)^2)+1;
              y(1)+0.1*y(2)*(9-y(1)^2-y(2)^2)+1];
[y]=EulerInkluze(F,0,12,[5 0],0.005,4);
```

Na Obrázku 14 vlevo je znázorněna pravá strana diferenciální inkluze (30) v bodech řešení získaného pomocí Eulerovy metody, a to zeleně její maximální hodnoty a červeně minimální hodnoty. Obdobně na obrázku vpravo pro diferenciální inkluzi (31). Tyto křivky znázorňujeme proto, že pravé strany diferenciálních inkluzí jsou mnohoznačná zobrazení, jelikož obsahují interval $\langle -1, 1 \rangle$. Hodnoty, kterých pravé strany uvažované úlohy mohou nabývat, tak leží v pásu, jehož horní hranice má zelenou barvu a dolní hranice červenou barvu. Horní hranice pásu odpovídá maximálním hodnotám pravých stran, které získáme tak, že z intervalu $\langle -1, 1 \rangle$ vybereme maximální hodnotu, tedy 1. Podobně dolní hranice pásu odpovídá minimálním hodnotám pravých stran a získáme je tak, že z intervalu $\langle -1, 1 \rangle$ vybereme minimální hodnotu, tedy -1.



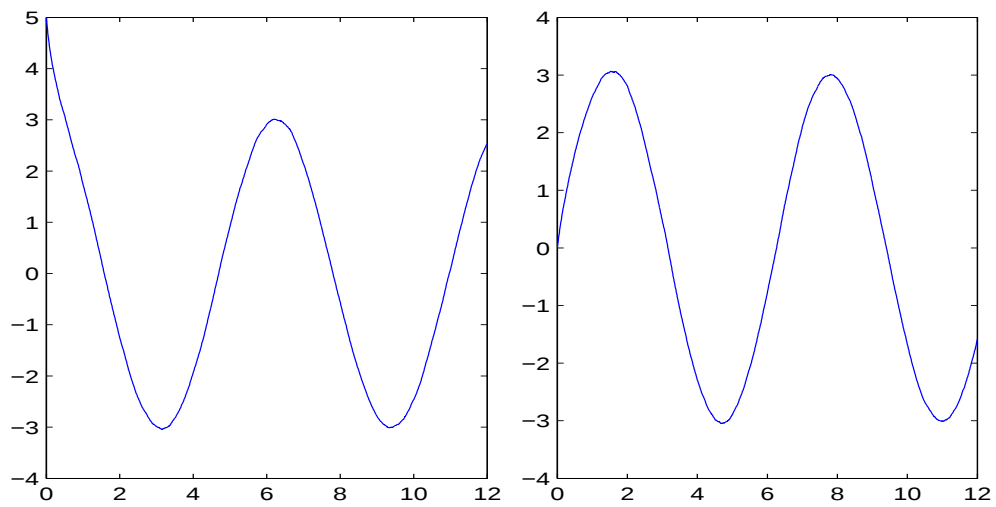
Obrázek 14

Na Obrázku 15 je znázorněn, a to modře, výběr náhodného řešení pro jednotlivé pravé strany inkluzí. Tedy z intervalu $\langle -1, 1 \rangle$ vybíráme náhodné číslo, počítáme hodnoty pravých stran příslušné tomuto číslu a řešíme příslušné diferenciální rovnice.



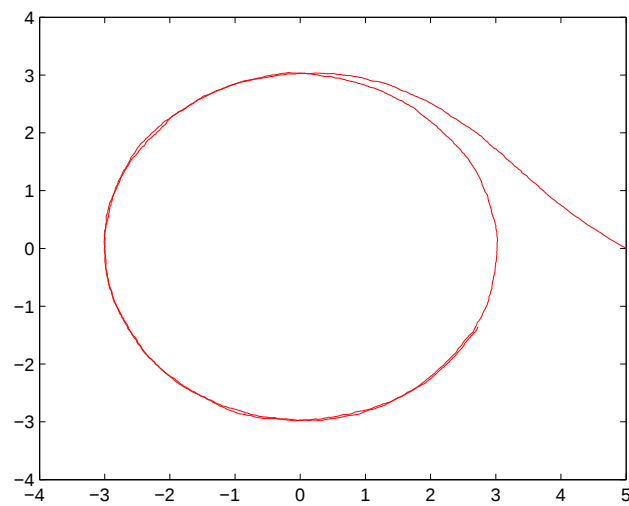
Obrázek 15

Na Obrázku 16 je znázorněna vlevo první složka řešení pomocí strategie náhodného výběru a vpravo druhá složka řešení.



Obrázek 16: vlevo 1. složka řešení, vpravo 2. složka řešení

Na závěr je na posledním Obrázku 17 znázorněn fázový portrét složek nalezených řešení, přičemž na vodorovnou osu je nanesena první složka řešení a na svislou druhá složka řešení.



Obrázek 17: fázový portrét řešení



Konvergence

Označme X jako množinu všech řešení úlohy (27), (28), X_h jako svazek řešení diferenciální inkluze (29) pro dané h a $\eta^N(\cdot) : I \times \mathbb{R}^m$ jako řešení diferenciální inkluze (29) pro daný krok h , což je spojitá, po částech lineární funkce taková, že

$$\eta^N(x) = \eta_n + \frac{1}{h}(x - x_n)(\eta_{n+1} - \eta_n)$$

pro každé $x \in \langle x_n, x_{n+1} \rangle$, $n = 0, \dots, N-1$, přičemž $\eta_0, \dots, \eta_N$ vyhovují inkluzi tvaru (29).

Vyvstává otázka, zda-li alespoň jedna podposloupnost posloupnosti $(\eta^N(\cdot))_{N \in \mathbb{N}}$ konverguje k řešení $y(\cdot) \in X$ úlohy (27), (28). Na to nám odpoví následující věta.

Věta 4. *Předpokládejme, že mnohoznačné zobrazení $F : I \times \mathbb{R}^m \rightrightarrows \mathbb{R}^m$ splňuje následující podmínky:*

- (i) *F je neprázdné, kompaktní, konvenční, shora polospojité v $I \times \mathbb{R}^m$.*
- (ii) *Existují konstanty k a c takové, že*

$$\|z\| \leq k\|w\| + c$$

pro každé $z \in F(x, w)$, $w \in \mathbb{R}^m$, $x \in I$.

Pak každá posloupnost $(\eta^N(\cdot))_{N \in \mathbb{N}}$, kde $\eta^N(\cdot) \in X_h$ pro $N \in \mathbb{N}$, obsahuje podposloupnost, která konverguje pro $N \rightarrow \infty$ stejnoměrně v I k nějakému řešení úlohy (27), (28).

Důkaz: Viz [3] str. 267.

Přidáme-li nějaké další podmínky, jsme schopni dokázat, že každé řešení $\eta^N(\cdot)$ úlohy (29) s dostatečně malým krokem obsahuje v jeho ϵ -novém okolí jediné řešení z $C(I)^m$ úlohy (27), (28), viz Věta 5.

Poznámka 11. Zápis $y(x) = (y_1(x), \dots, y_m(x)) \in C(I)^m$ znamená, že $y_1(x) \in C(I)$, $y_2(x) \in \mathbb{C}(I)$, $\dots$, $y_m(x) \in C(I)$.

Věta 5. Předpokládejme, že jsou splněny předpoklady z věty 4 a nechť navíc zobrazení F je spojitý v $I \times \mathbb{R}^m$ a Lipschitzovské ve w na omezené množině v $\mathbb{R}^m$ s Lipschitzovskou konstantou integrovatelnou v I . Pak pro každé $\epsilon > 0$ existuje $N_1 \in \mathbb{N}$ takové, že pro všechna $N \in \mathbb{N}$, $N > N_1$ a pro každé řešení $\eta^N(\cdot)$ úlohy (29) existuje řešení $y(\cdot)$ úlohy (27), (28) takové, že

$$\max_{x \in I} \|\eta^N(x) - y(x)\| \leq \epsilon.$$

Důkaz: Viz [3] str. 269.

Další otázkou, kterou se budeme zabývat, je odhad chyby, neboli odhad vzdálenosti mezi množinami řešení počátečního problému (27), (28) a diferenciální inkluze (29):

Věta 6. Nechť $F : \mathbb{R}^m \multimap \mathbb{R}^m$ je kompaktní, konvexní zobrazení a y je řešení

$$y'(x) \in F(y(x)) \quad \text{pro skoro všechna } x \in I, \quad y(x_0) = y_0.$$

Předpokládejme, že existuje $\epsilon > 0$ takové, že F je Lipschitzovské na množině $\{w \in \mathbb{R}^m : \|w - y(x)\| \leq \epsilon \text{ pro některá } x \in I\}$. Pak existují $c \in \mathbb{R}$ a $N_1 \in \mathbb{N}$ takové, že pro všechna $N \in \mathbb{N}$, $N > N_1$ existuje řešení $\eta^N(\cdot)$ inkluze

$$\eta_{n+1} \in \eta_n + h F(\eta_n), \quad n = 0, 1, \dots, N-1, \quad \eta_0 = y_0$$

takové, že

$$\max_{0 \leq n \leq N} \|\eta^N(x_n) - y(x_n)\| \leq c h.$$

Důkaz: Viz [3] str. 283.

Věta 6 tedy říká, že za uvedených předpokladů se v každém kroku při hledání řešení daného počátečního problému (27), (28) Eulerovou metodou dopustíme chyby o velikosti maximálně $O(h)$.

Věta 7. *Nechť jsou splněny následující předpoklady*

(i) $F : I \times \mathbb{R}^m \multimap \mathbb{R}^m$ je neprázdné, kompaktní a konvexní zobrazení.

(ii) Existují konstanty k a c takové, že

$$\|z\| \leq k\|w\| + c$$

pro každé $z \in F(x, w)$, $w \in \mathbb{R}^m$, $x \in I$.

(iii) F je Lipschitzovské ve w na omezené množině stejnoměrně v x .

(iv) F je omezená variace v x stejnoměrně ve w na omezené množině, tj. pro každou omezenou množinu $U \subset \mathbb{R}^m$ platí

$$\sup \left\{ \sum_{i=1}^{\nu} \sup_{w \in U} \{ \text{haus}(F(x_{i+1}, w), F(x_i, w)) \} : x_0 \leq \dots \leq x_{\nu+1} \leq x_N, N, \nu \in \mathbb{N} \right\} < \infty.$$

Pak

$$\text{haus}(X, X_h) = O(h) \quad \text{na } C(I)^m.$$

Důkaz: Viz [3] str. 283.

Věta 7 tedy říká, že za uvedených předpokladů je odhad Hausdorffovy vzdálenosti mezi množinami řešení počátečního problému (27), (28) a diferenciální inkluze (29) řádu h .

Poznámka 12. Nechť X je úplný metrický prostor s metrikou $d(., .)$, $H(X)$ je množina neprázdných uzavřených podmnožin prostoru X . Pak $\text{haus}(M, N)$ značí Hausdorffovu vzdálenost množin $M, N \in H(X)$, která je definována následovně

$$\text{haus}(M, N) = \max\{d(M, N), d(N, M)\},$$

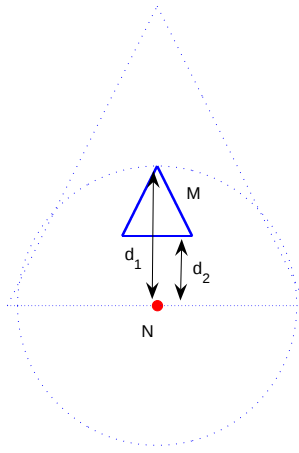
kde

$$d(M, N) = \sup_{y \in M} d(y, N).$$

Poznámka 13. Uvedeme si příklad na Hausdorffovu vzdálenost dvou množin, konkrétně bodu a trojúhelníku: Nechť množina M je trojúhelník a množina N je bod. Hausdorffova vzdálenost těchto množin je znázorněna na Obrázku 18, přičemž

$$haus(M, N) = haus(\triangle, \circ) = \max\{d_1, d_2\},$$

kde $d_1 = d(\triangle, \circ)$ a $d_2 = d(\circ, \triangle)$. V našem případě je $haus(\triangle, \circ) = d_1$.



Obrázek 18

Doposud jsme se zabývali množinami X a X_h všech řešení daných úloh. Někdy nás však mohou více zajímat hodnoty všech řešení v určitých bodech. Dostáváme se tak k problému aproximace dosažitelných množin.

Definice 12. Nechť $x \in I$. **Dosažitelná množina** v čase x úlohy (27), (28) je definována následovně:

$$R(x) = \{w \in \mathbb{R}^m : w = y(x) \text{ pro některé } y(\cdot) \in X\}.$$

Dosažitelná množina je vlastně množina všech bodů w , které jsou konci trajektorií z (27), (28) na $\langle x_0, x \rangle$.

Podobně můžeme definovat dosažitelnou množinu diferenciální inkluze (29). Nechť n je pevné číslo z množiny $\{0, 1, \dots, N\}$. Pak dosažitelná množina diferenciální inkluze (29) v čase x_n je

$$R_n^N = \{w \in \mathbb{R}^m : w = \eta^N(x_n) \text{ pro některá } \eta^N(\cdot) \in X_h\}.$$

Zabývejme se nyní tím, jestli dosažitelná množina diferenciální inkluze (29) konverguje k dosažitelné množině úlohy (27), (28). Na to nám odpoví následující věta.

Věta 8. *Nechť $F : I \times \mathbb{R}^m \multimap \mathbb{R}^m$ je kompaktní, konvexní zobrazení, které je spojitě v obou proměnných x a w v $I \times \mathbb{R}^m$ a nechť dosažitelná množina $R(x)$ počátečního problému (27), (28) je obsažena v omezené množině $K \subset \mathbb{R}^m$ pro všechna $x \in I$. Nechť F je Lipschitzovské v okolí množiny K vzhledem k w stejnoměrně v $x \in I$. Pak*

$$\max_{0 \leq n \leq N} \text{haus}(R_n^N, R(x_n)) \rightarrow 0$$

pro $N \rightarrow \infty$.

Důkaz: Viz [3] str. 285.

Problematika konvergence je velmi obsáhlá. V této práci je z důvodu jejího rozsahu uvedeno jen několik základních konvergenčních vět z článku [3]. Další konvergenční výsledky je možno nalézt například v publikacích

- Dontchev, A. L., Farkhi, E. M.: *Error Estimates for Discretized Differential Inclusions*. Computing, 41 (1989), pp. 349-358.
- Filippov, A. F.: *Differential Equations with Discontinuous Righthand Side*. Mathematics and Its Applications, Kluwer Academic Publishers, Dordrecht, Boston, London, 1988.

V následující části této kapitoly se budeme zabývat některými strategiemi výběru jediného řešení uvažované diferenciální inkluze.

3.2 Strategie výběru selekce

Řešíme-li diferenciální inkluzi, dostaneme celý svazek řešení. Proto je důležitý výběr strategie, tedy metody, která vybere pouze jedno řešení z tohoto svazku. Toto jedno řešení budeme nazývat selekce. Strategií na výběr selekce je mnoho. Na Příkladě 8 jsme si ukázali strategii náhodného výběru. Nyní si ukážeme nějaké další možné strategie.

3.2.1 Výběr s minimální normou

Podstatou strategie výběru selekce pomocí minimální normy je vybrat v každém kroku jedinou hodnotu mnohoznačné pravé strany uvažované diferenciální inkluze z množiny hodnot, kterých může nabývat, a to tak, aby byla co nejméně vzdálena od nějakého zvoleného prvku. Postupujeme tedy následovně:

Vybereme pevný prvek $z \in \mathbb{R}^m$ a pro $n = 0, \dots, N - 1$ minimalizujeme

$$||z - \zeta_n||,$$

přičemž

$$\eta_{n+1} = \eta_n + h\zeta_n, \quad (32)$$

$$\zeta_n \in F(x_n, \eta_n). \quad (33)$$

Zvolíme-li za $z = 0$, dostáváme v každém kroku výběr s minimální normou.

Ukažme si tento výběr na Příkladě 8.

Příklad 9. Řešte Příklad 8 pomocí Eulerovy metody s krokem $h = 0.005$ a výběru selekce s minimální normou.

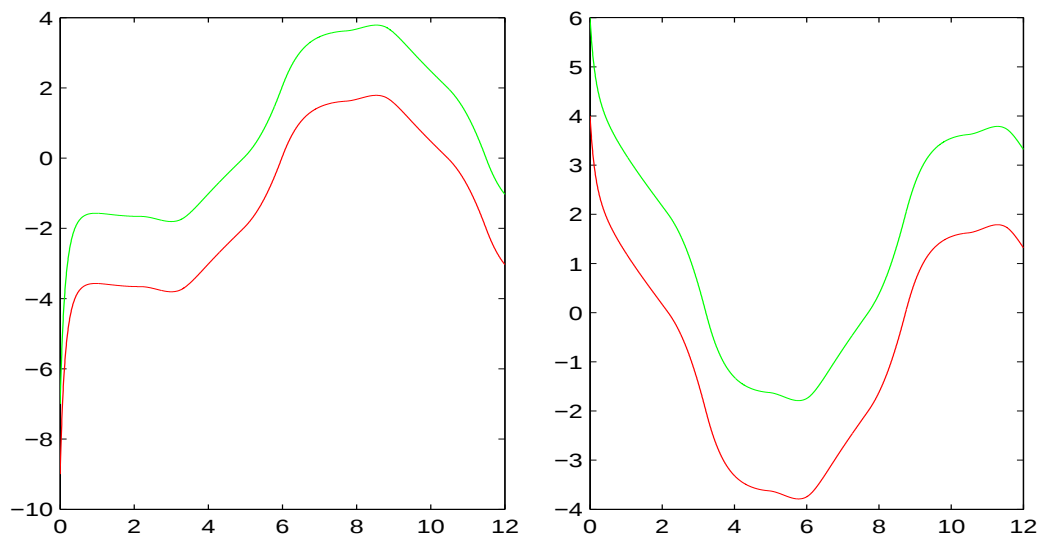
Řešení: Tuto úlohu vyřešíme opět pomocí mnou sestaveného M-souboru uvedeného v odstavci 4.4. Tentokrát však zvolíme hodnotu *parametr* jako číslo 5, které odpovídá výběru s minimální normou. Do Matlabu tedy zadáme:

```

F.min=@(x,y)[-y(2)+0.1*y(1)*(9-y(1)^2-y(2)^2)-1;
           y(1)+0.1*y(2)*(9-y(1)^2-y(2)^2)-1];
F.max=@(x,y)[-y(2)+0.1*y(1)*(9-y(1)^2-y(2)^2)+1;
           y(1)+0.1*y(2)*(9-y(1)^2-y(2)^2)+1];
[y]=EulerInkluze(F,0,12,[5 0],0.005,5);

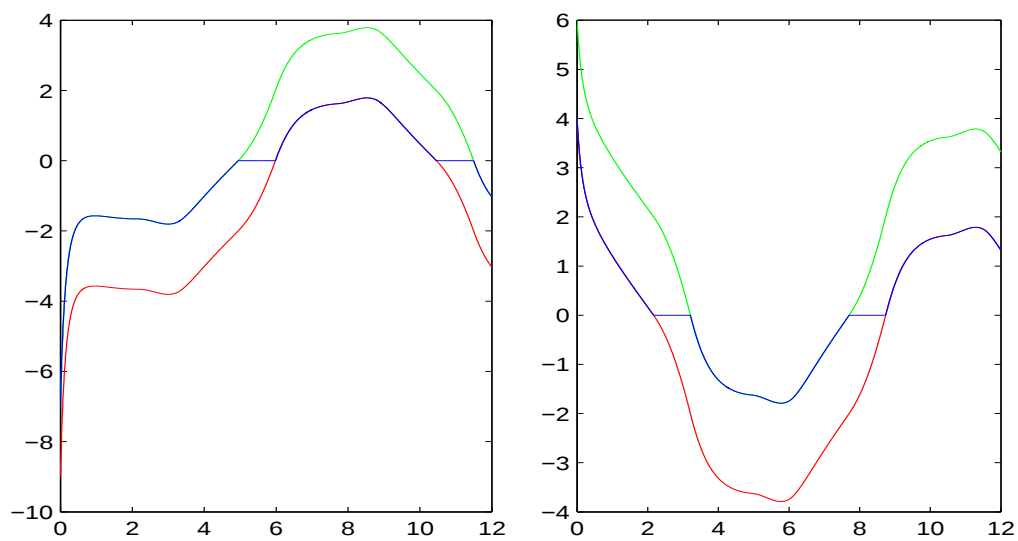
```

Řešení této úlohy si ukážeme na následujících obrázcích: Na Obrázku 19 vlevo je znázorněna pravá strana diferenciální inkluze (30) v bodech řešení získaného pomocí Eulerovy metody, a to zeleně pro maximální hodnoty pravé strany, které odpovídají maximální hodnotě intervalu $\langle -1, 1 \rangle$, tedy 1, a červeně pro minimální hodnoty pravé strany, které odpovídají minimální hodnotě intervalu $\langle -1, 1 \rangle$, tedy -1 . Obdobně na obrázku vpravo pro diferenciální inkluzi (31). Máme tedy znázorněn pás, ze kterého nyní podle výše uvedeného postupu budeme v každém kroku vybírat jedinou hodnotu.



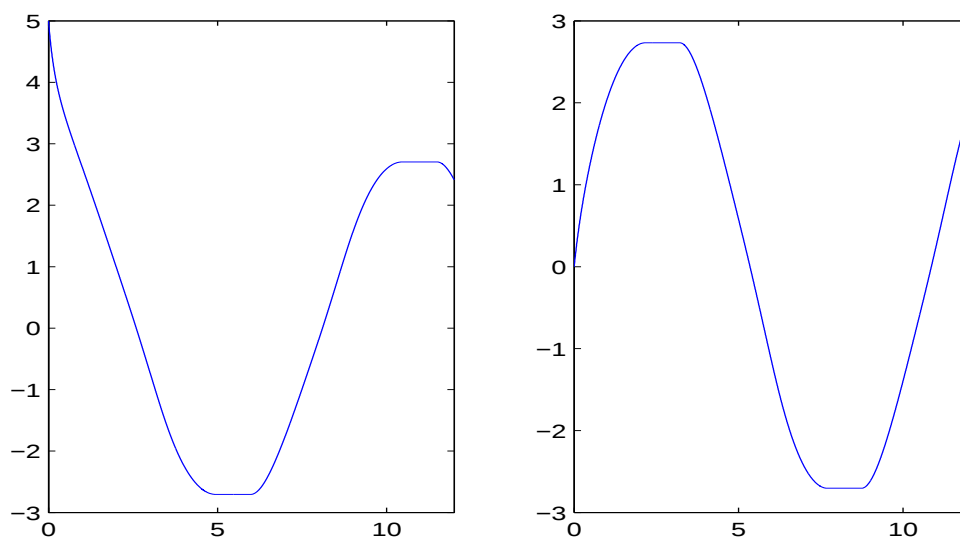
Obrázek 19

Na Obrázku 20 je znázorněn, a to modře, výběr řešení s minimální normou pro jednotlivé pravé strany inkluzí.



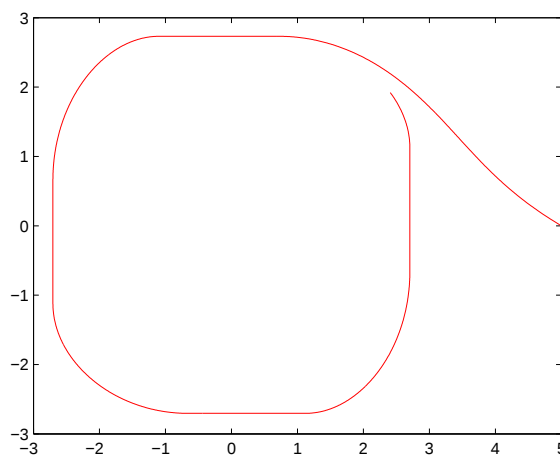
Obrázek 20

Na Obrázku 21 je znázorněna vlevo první složka řešení pomocí strategie výběru s minimální normou a vpravo druhá složka řešení.



Obrázek 21: vlevo 1. složka řešení, vpravo 2. složka řešení

Na závěr je na posledním Obrázku 22 znázorněn fázový portrét složek nalezených řešení, přičemž na vodorovnou osu je nanesena první složka řešení a na svislou druhá složka řešení.



Obrázek 22: fázový portrét řešení

★

3.2.2 Výběr s minimální variací

Podstatou strategie výběru selekce pomocí minimální variace je v aktuálním kroku vybrat hodnotu pravé strany diferenciální inkluze z množiny všech možných hodnot takovou, jejíž vzdálenost od již vybrané hodnoty pravé strany v předcházejícím kroku je nejmenší. Postup je tedy následovný:

Pro $n = 0, \dots, N - 1$ minimalizujeme

$$\|v_n - v_{n+1}\|,$$

přičemž

$$\eta_{n+1} = \eta_n + h v_{n+1}, \tag{34}$$

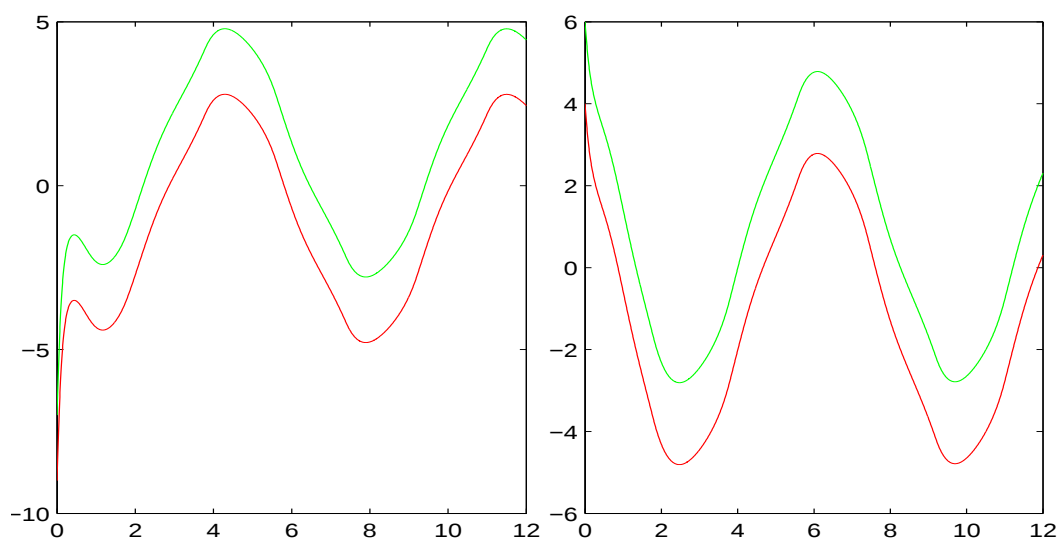
$$v_{n+1} \in F(x_n, \eta_n). \tag{35}$$

Příklad 10. Řešte Příklad 8 pomocí Eulerovy metody s krokem $h = 0.005$ a výběru selekce s minimální variací.

Řešení: Tento příklad opět vyřešíme pomocí M-souboru uvedeného v odstavci 4.4 zadáním za hodnotu *parametr* čísla 6, které odpovídá výběru s minimální variací. Do Matlabu zadáme

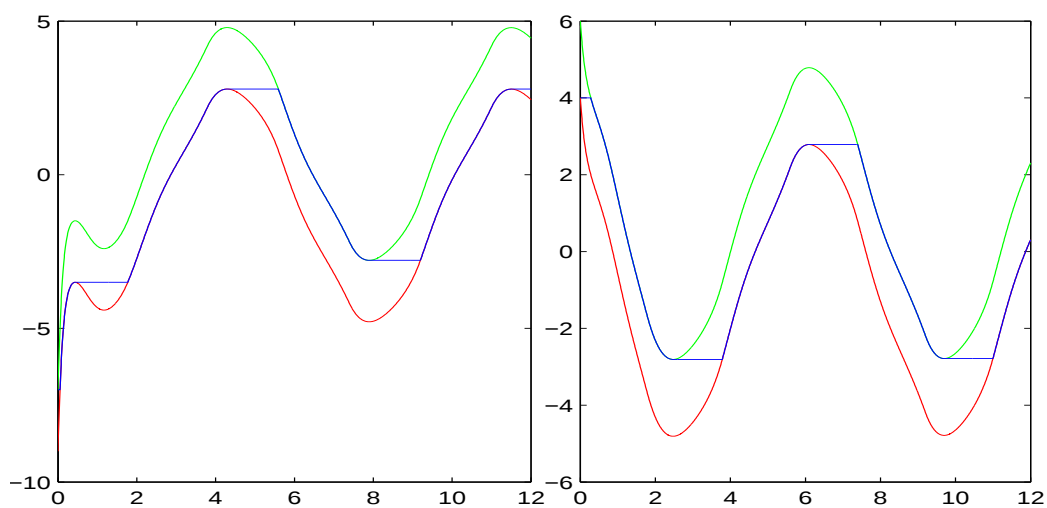
```
F.min=@(x,y)[-y(2)+0.1*y(1)*(9-y(1)^2-y(2)^2)-1;  
            y(1)+0.1*y(2)*(9-y(1)^2-y(2)^2)-1];  
F.max=@(x,y)[-y(2)+0.1*y(1)*(9-y(1)^2-y(2)^2)+1;  
            y(1)+0.1*y(2)*(9-y(1)^2-y(2)^2)+1];  
[y]=EulerInkluze(F,0,12,[5 0],0.005,6);
```

Řešení této úlohy si opět ukážeme na následujících obrázcích: Na Obrázku 23 vlevo je znázorněna pravá strana diferenciální inkluze (30) v bodech řešení získaného pomocí Eulerovy metody, a to zeleně pro maximální hodnoty pravé strany, které odpovídají maximální hodnotě intervalu $\langle -1, 1 \rangle$, tedy 1, a červeně pro minimální hodnoty pravé strany, které odpovídají minimální hodnotě intervalu $\langle -1, 1 \rangle$, tedy -1 . Obdobně na obrázku vpravo pro diferenciální inkluzi (31). Opět tak dostáváme pás, ve kterém leží všechny hodnoty, kterých pravá strana diferenciální inkluze může nabývat.



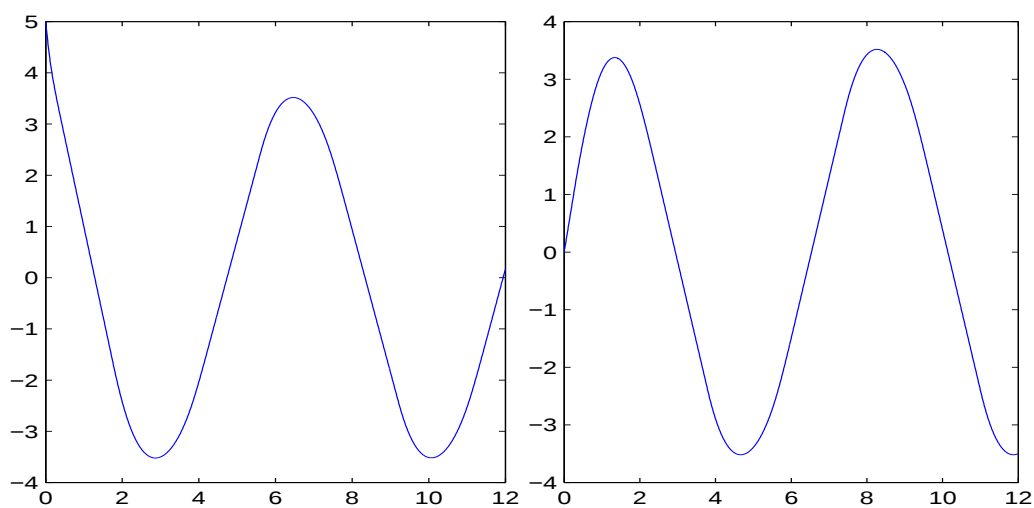
Obrázek 23

Na Obrázku 24 je znázorněn, a to modře, výběr řešení s minimální variací pro jednotlivé pravé strany inkluzí.



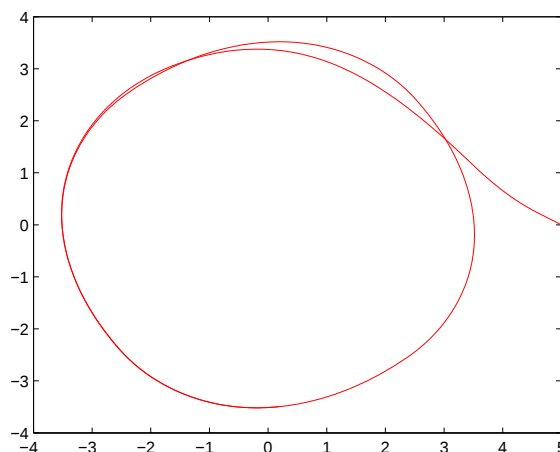
Obrázek 24

Na Obrázku 25 je znázorněna vlevo první složka řešení pomocí strategie výběru s minimální variací a vpravo druhá složka řešení.



Obrázek 25: vlevo 1. složka řešení, vpravo 2. složka řešení

Na závěr je na posledním Obrázku 26 znázorněn fázový portrét složek nalezených řešení, přičemž na vodorovnou osu je nanesena první složka řešení a na svislou druhá složka řešení.



Obrázek 26: fázový portrét řešení

★

3.2.3 Výběr s respektováním referenční trajektorie

V tomto případě vybíráme tentokrát řešení, které nejlépe respektuje zadanou referenční trajektorii. Postupujeme pomocí níže uvedeného návodu:

Nejprve vyberme lipschitzovsky spojitou referenční trajektorii $\bar{y} : I \rightarrow \mathbb{R}^m$. Pro $n = 0, \dots, N - 1$ minimalizujeme

$$\|\bar{y}(x_{n+1}) - \eta_{n+1}\|,$$

přičemž

$$\eta_{n+1} = \eta_n + h \zeta_n, \tag{36}$$

$$\zeta_n \in F(x_n, \eta_n). \tag{37}$$

Ukažme si tuto strategii výběru selekce opět na Příkladě 8 s referenční trajektorií $\bar{y}(x) \equiv 0$.

Příklad 11. Řešte Příklad 8 pomocí Eulerovy metody s krokem $h = 0.005$ a výběru selekce s respektováním referenční trajektorie $\bar{y}(x) \equiv 0$.

Řešení: Příklad vyřešíme opět pomocí M-souboru z odstavce 4.4, přičemž za *parametr* volíme číslo 7. Do Matlabu zadáme

```
F.min=@(x,y)[-y(2)+0.1*y(1)*(9-y(1)^2-y(2)^2)-1;  
            y(1)+0.1*y(2)*(9-y(1)^2-y(2)^2)-1];  
F.max=@(x,y)[-y(2)+0.1*y(1)*(9-y(1)^2-y(2)^2)+1;  
            y(1)+0.1*y(2)*(9-y(1)^2-y(2)^2)+1];  
[y]=EulerInkluze(F,0,12,[5 0],0.005,7);
```

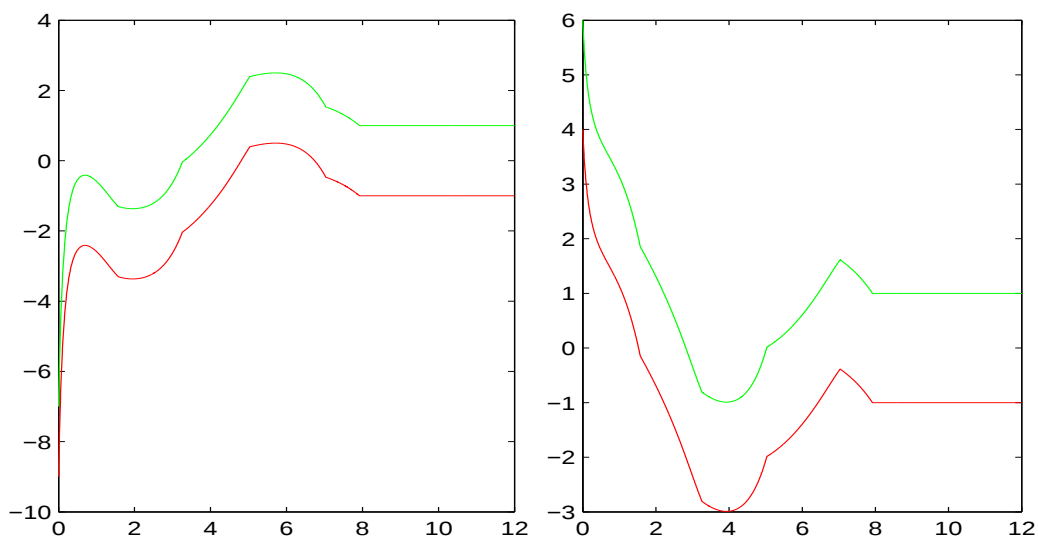
Jelikož jsme zvolili strategii výběru s respektováním referenční trajektorie, budeme vyzváni, abychom zadali referenční trajektorii. V Matlabu se objeví hlášení

zadej referenční trajektorii:

Referenční trajektorii zadáváme jako funkci proměnné x . My v našem případě zadáme

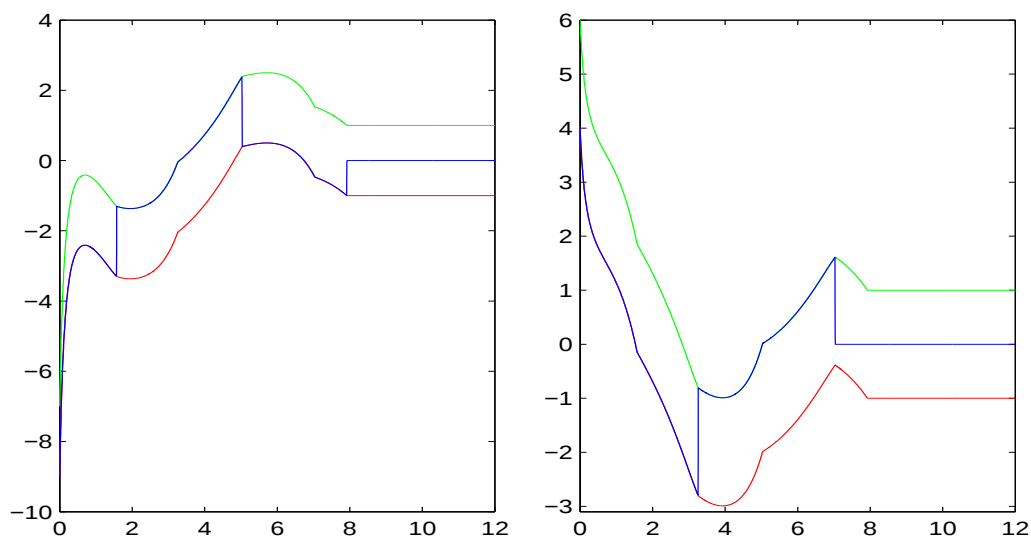
```
@(x)0
```

Po stisknutí tlačítka Enter dostaneme řešení zadané úlohy, které si ukážeme na následujících obrázcích: Na Obrázku 27 vlevo je znázorněna pravá strana diferenciální inkluze (30) v bodech řešení získaného pomocí Eulerovy metody, a to zeleně pro maximální hodnoty pravé strany, které odpovídají maximální hodnotě intervalu $\langle -1, 1 \rangle$, tedy 1, a červeně pro minimální hodnoty pravé strany, které odpovídají minimální hodnotě intervalu $\langle -1, 1 \rangle$, tedy -1 . Obdobně na obrázku vpravo pro diferenciální inkluzi (31).



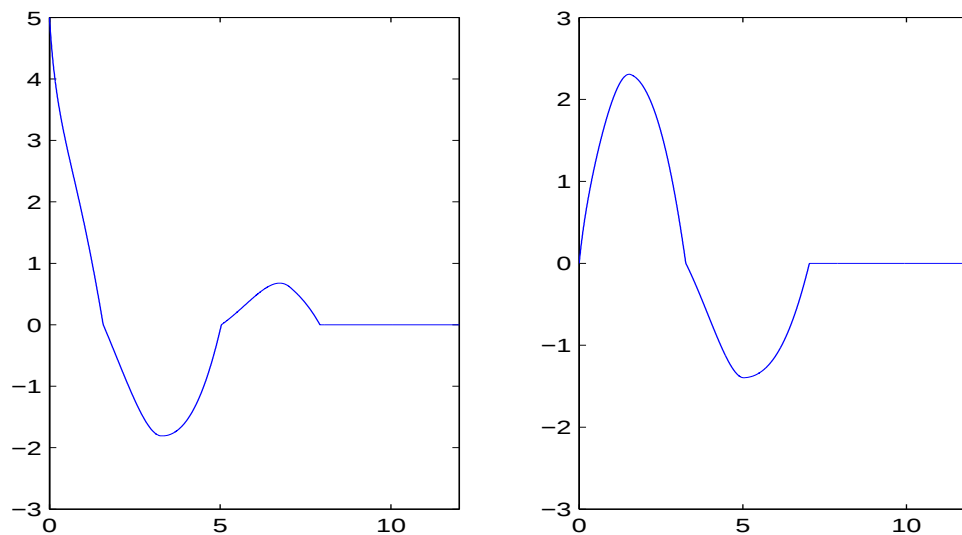
Obrázek 27

Na Obrázku 28 je znázorněn, a to modře, výběr řešení s respektováním referenční trajektorie pro jednotlivé pravé strany inkluzí.



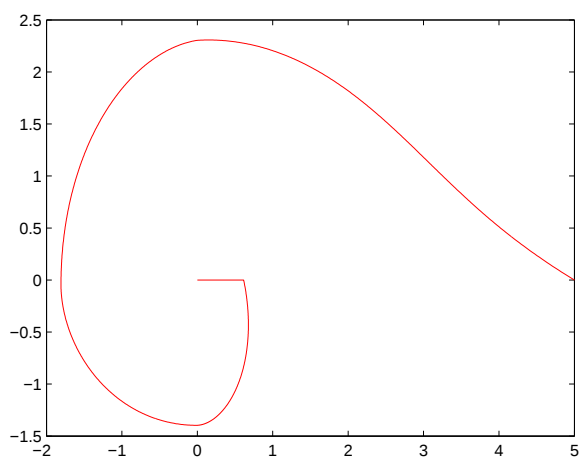
Obrázek 28

Na Obrázku 29 je znázorněna vlevo první složka řešení pomocí strategie výběru s respektováním referenční trajektorie a vpravo druhá složka řešení.



Obrázek 29: vlevo 1. složka řešení, vpravo 2. složka řešení

Na závěr je na posledním Obrázku 30 znázorněn fázový portrét složek nalezených řešení, přičemž na vodorovnou osu je nanesena první složka řešení a na svislou druhá složka řešení.



Obrázek 30: fázový portrét řešení



V následující části této kapitoly si ukážeme, jak diferenciální inkluze vyřešit pomocí metod Runge–Kutta.

3.3 Metody Runge–Kutta pro diferenciální inkluze

Uvažujme počáteční úlohu (27), (28). Pro získání řešení této úlohy metodami Runge–Kutta postupujeme tak, že pro $N \in \mathbb{N}$ zvolíme síť $x_0 < x_1 < \dots < x_N = b$ s krokem $h = \frac{b-x_0}{N}$. Dále položíme počáteční aproximaci rovnu $\eta_0 = y_0$ a pro $n = 0, \dots, N-1$ řešíme systém inkluzí

$$\bar{\eta}_i = \eta_n + h \sum_{j=1}^s \beta_{ij} \zeta_{nj}, \quad (38)$$

$$\zeta_{nj} \in F(x_n + \alpha_j h, \bar{\eta}_j), \quad i, j = 1, \dots, s. \quad (39)$$

a další aproximaci počítáme ze vztahu

$$\eta_{n+1} = \eta_n + h \sum_{j=1}^s w_j \zeta_{nj}, \quad (40)$$

kde $w_j, \alpha_j, \beta_{ij}$, pro $i, j = 1, \dots, s$, jsou konstanty. Číslo s značí počet stádií metody Runge–Kutta.

Poznámka 14. Předchozí zápis metod Runge–Kutta je zápisem jeho implicitní formy, která však v sobě vhodnou volbou konstant β_{ij} zahrnuje i tu explicitní. Vhodnou volbou rozumíme $\beta_{ij} = 0$ pro $i \leq j$.

Nyní vyřešíme Příklad 8 explicitními metodami Runge–Kutta pro stádia $s = 2$, $s = 3$ a $s = 4$, jejichž vzorce jsou zapsány pomocí následujících tabulek:

0	0	0
1	1	0
	$\frac{1}{2}$	$\frac{1}{2}$

$s = 2$

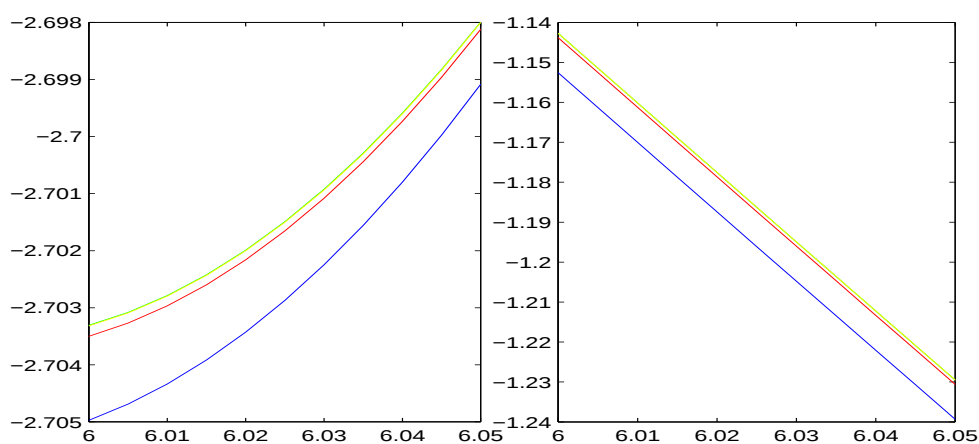
0	0	0	0
$\frac{1}{3}$	$\frac{1}{3}$	0	0
1	$\frac{1}{3}$	$\frac{2}{3}$	0
	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{3}$

$$s = 3$$

0	0	0	0	0
$\frac{1}{2}$	$\frac{1}{2}$	0	0	0
$\frac{1}{2}$	0	$\frac{1}{2}$	0	0
1	0	0	1	0
	$\frac{1}{6}$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{6}$

$s = 4$: klasická metoda Runge–Kutta

Jako strategii výběru selekce zvolíme např. výběr s minimální normou. Řešení získaná pomocí jednotlivých metod Runge–Kutta a strategií výběru selekce s minimální normou se příliš neliší od řešení získané Eulerovou metodou se stejným výběrem, jak lze vidět z Obrázku 31. Pro názornost je znázorněn výřez řešení, přičemž modře řešení získané Eulerovou metodou, zeleně metodou Runge–Kutta s $s = 2$, červeně metodou Runge–Kutta s $s = 3$ a žlutě klasickou metodou Runge–Kutta.



Obrázek 31: vlevo 1. složka řešení, vpravo 2. složka řešení

3.4 Řešené příklady

V tomto odstavci vyřešíme tři úlohy pomocí poznatků získaných v kapitole 3.

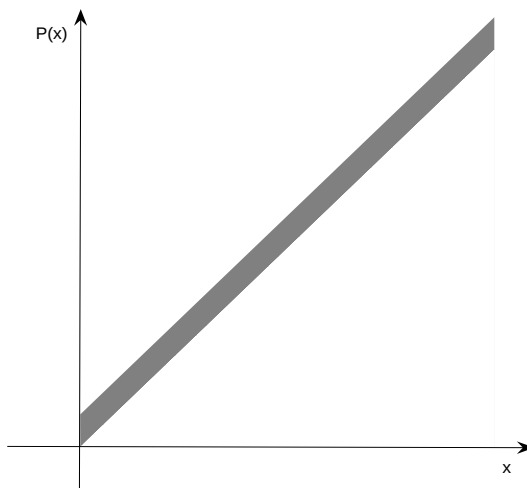
Příklad 12. Vyřešte diferenciální inkluzi

$$y''(x) + ay'(x) + by(x) \in P(x),$$

kde $a, b \in \mathbb{R}$, $y(0) = 2$, $y'(0) = 3$, $x \in \langle 0, 12 \rangle$. Za $P(x)$ volte různé pásy hodnot.

Řešení:

a) Za $P(x)$ zvolíme $P(x) = \langle x, x + 1 \rangle$. Tento pás je znázorněn na Obrázku 32.



Obrázek 32

Dále zvolíme např. $a = 1$, $b = 2$. Dostaneme tak diferenciální inkluzi

$$y''(x) + y'(x) + 2y(x) \in \langle x, x + 1 \rangle,$$

kteřou můžeme přepsat na soustavu

$$y'_1 = y_2 \tag{41}$$

$$y'_2 \in -y_2 - 2y_1 + \langle x, x + 1 \rangle \tag{42}$$

s počátečními podmínkami $y_1(0) = 2$, $y_2(0) = 3$. Tuto soustavu vyřešíme Eulerovou metodou pro řešení diferenciálních inkluzí s krokem $h = 0.005$. Zvolíme např. strategii výběru selekce s respektováním referenční trajektorie $\bar{y}(x) = \sin(x)$. Tento příklad vyřešíme pomocí M-souboru z odstavce 4.4 (za hodnotu *parametr* zvolíme číslo 7, které odpovídá strategii výběru selekce pomocí referenční trajektorie): Do Matlabu zadáme

```
a=1;
b=2;
F.min=@(x,y) [y(2);-a*y(2)-b*y(1)+x];
F.max=@(x,y) [y(2);-a*y(2)-b*y(1)+x+1];
[y]=EulerInkluze(F,0,12,[2 3],0.005,7);
```

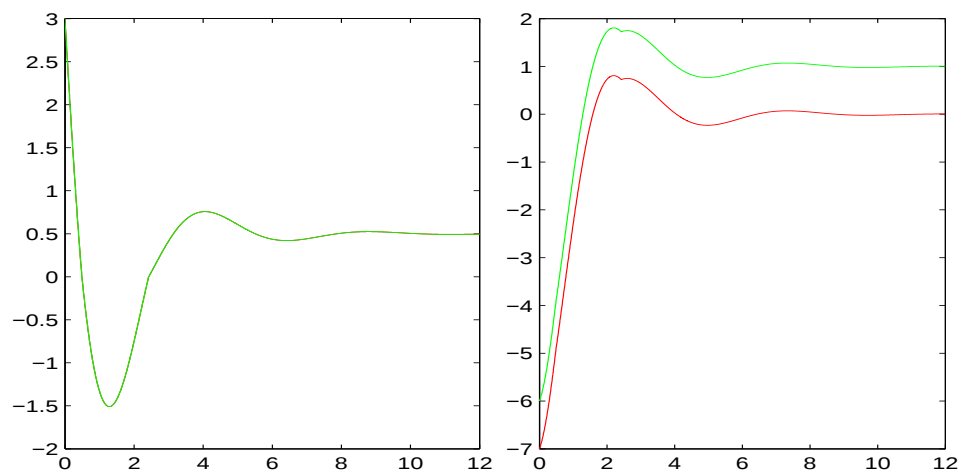
Dále budeme Matlabem vyzváni

zadej referenční trajektorii:

Referenční trajektorii zadáváme jako funkci proměnné x , tedy

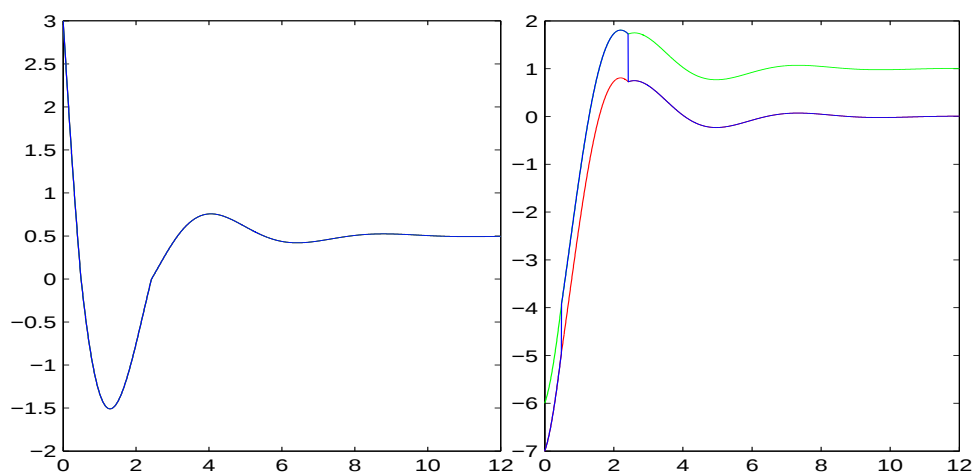
```
@(x)sin(x)
```

Řešení této úlohy znázorníme na následujících obrázcích: Na Obrázku 33 vpravo je znázorněna pravá strana diferenciální inkluze (42) v bodech řešení získané pomocí Eulerovy metody, a to zeleně pro maximální hodnotu pravé strany a červeně pro minimální hodnotu pravé strany. Na Obrázku 33 vlevo je znázorněna pravá strana diferenciální rovnice (41), která je jednoznačná.



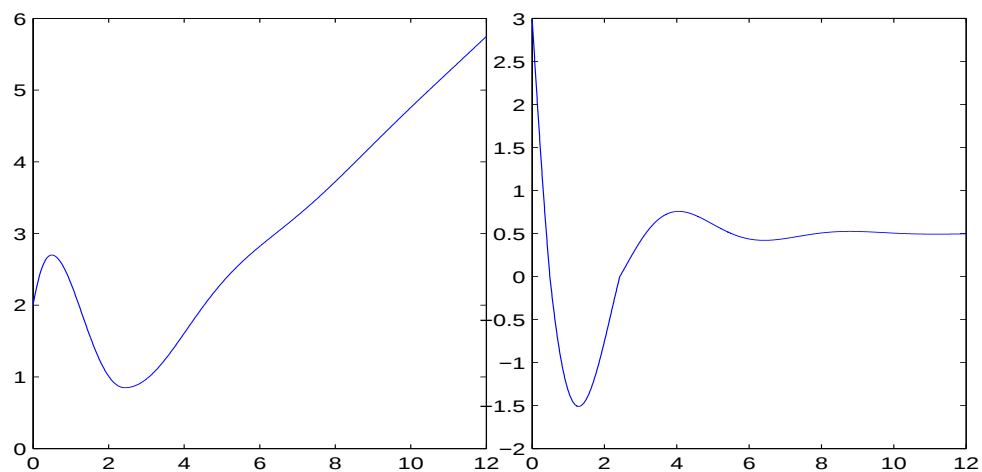
Obrázek 33

Na Obrázku 34 je znázorněn, a to modře, výběr řešení s respektováním referenční trajektorie pro jednotlivé pravé strany rovnice a inkluze. Jeli-kož pravá strana diferenciální rovnice (41) je jednoznačná, výběr je tu dán jednoznačně, na rozdíl od pravé strany diferenciální inkluze (42), kde mu-síme zvolit strategii výběru selekce, v tomto případě výběr s respektováním zvolené referenční trajektorie.



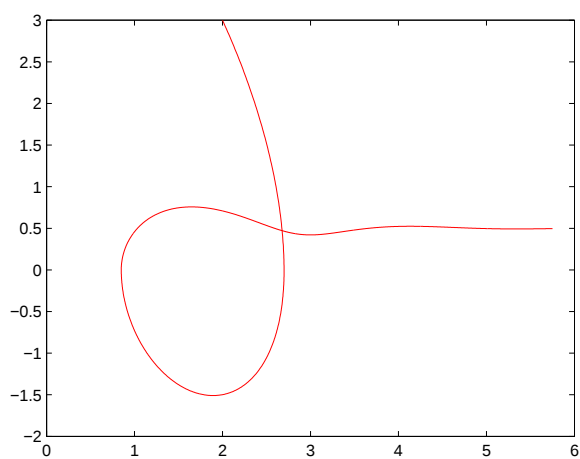
Obrázek 34

Na Obrázku 35 je znázorněna vlevo první složka řešení, vpravo druhá složka řešení pomocí strategie výběru s respektováním referenční trajektorie.



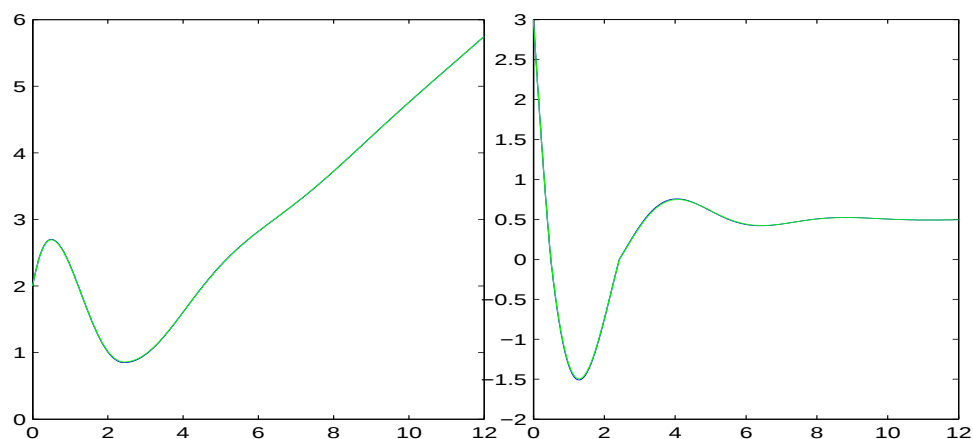
Obrázek 35: vlevo 1. složka řešení, vpravo 2. složka řešení

Na závěr je na posledním Obrázku 36 znázorněn fázový portrét složek nalezených řešení, přičemž na vodorovnou osu je nanesena první složka řešení a na svislou druhá složka řešení.



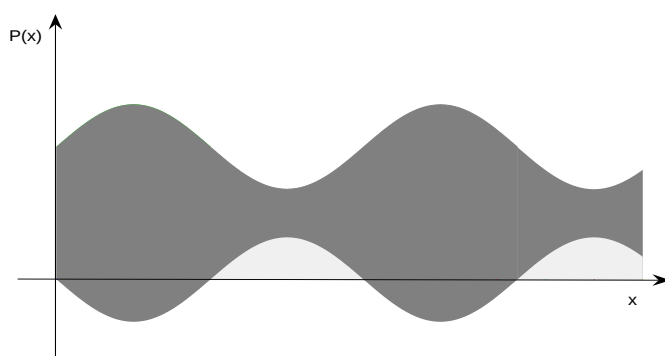
Obrázek 36: fázový portrét řešení

Kdybychom tento příklad řešili pomocí klasické metody Runge–Kutta se stejným výběrem selekce a stejnou délkou kroku, dostali bychom téměř identické řešení, které je společně s řešením získaným Eulerovou metodou znázorněno na následujícím obrázku:



Obrázek 37: modře řešení získané Eulerovou metodou, zeleně řešení získané klasickou metodou Runge–Kutta, vlevo 1. složka řešení, vpravo 2. složka řešení

- b) Za $P(x)$ zvolíme $P(x) = \langle -\sin(x), \sin(x) + \pi \rangle$. Tento pás je znázorněn na Obrázku 38.



Obrázek 38

Zvolíme opět např. $a = 1$, $b = 2$. Dostaneme tak diferenciální inkluzi

$$y''(x) + y'(x) + 2y(x) \in \langle -\sin(x), \sin(x) + \pi \rangle,$$

kterou můžeme přepsat na soustavu

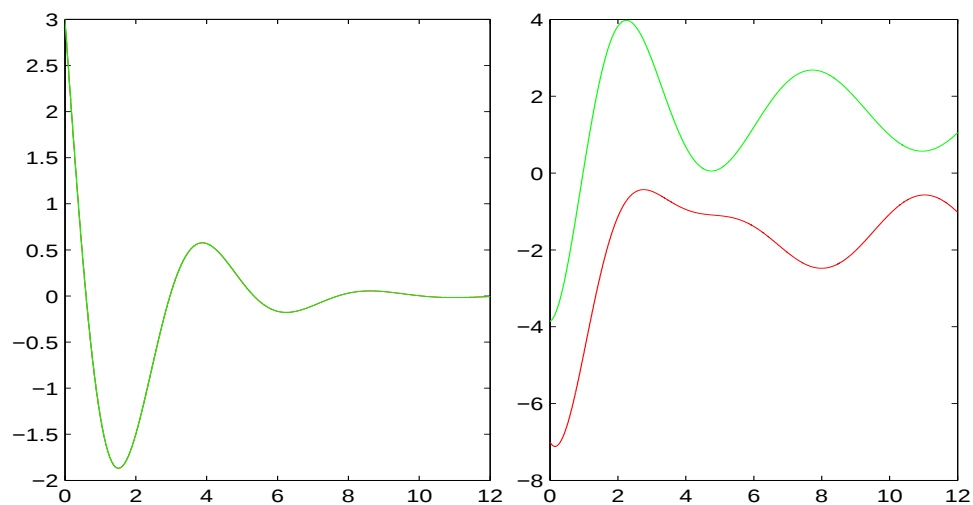
$$y'_1 = y_2 \tag{43}$$

$$y'_2 \in -y_2 - 2y_1 + \langle -\sin(x), \sin(x) + \pi \rangle \tag{44}$$

s počátečními podmínkami $y_1(0) = 2$, $y_2(0) = 3$. Tuto soustavu vyřešíme Eulerovou metodou pro řešení diferenciálních inkluzí s krokem $h = 0.005$. Zvolíme např. strategii výběru selekce pomocí střední hodnoty, tj. v bodech, ve kterých funkce pravé strany dané diferenciální inkluze nabývá více hodnot vybereme jednu z nich a to střední hodnotu. Tento příklad vyřešíme pomocí M-souboru z odstavce 4.4 (strategii výběru selekce pomocí střední hodnoty odpovídá hodnota *parametr* číslu 3): Do Matlabu zadáme

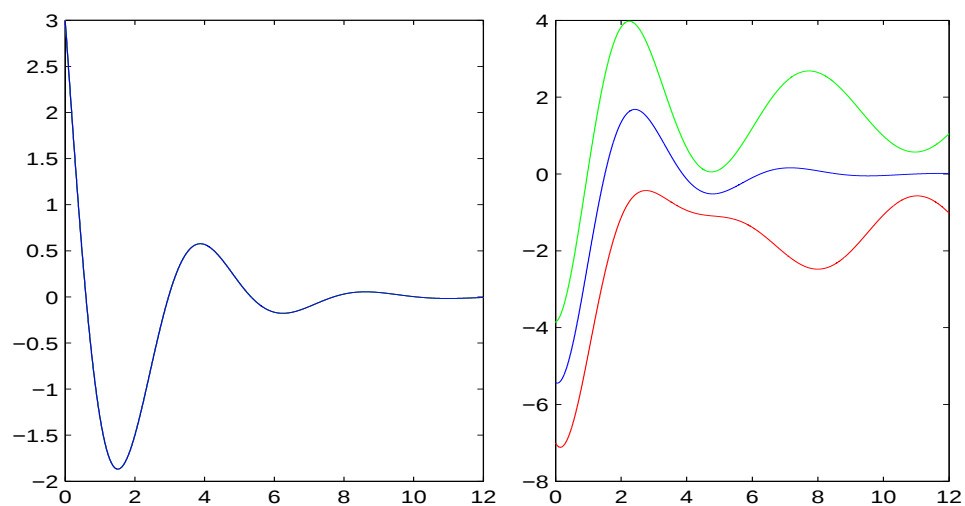
```
a=1;
b=2;
F.min=@(x,y)[y(2);-a*y(2)-b*y(1)-sin(x)];
F.max=@(x,y)[y(2);-a*y(2)-b*y(1)+sin(x)+pi];
[y]=EulerInkluze(F,0,12,[2 3],0.005,3);
```

Řešení této úlohy znázorníme na následujících obrázcích: Na Obrázku 39 vpravo je znázorněna pravá strana diferenciální inkluze (44) v bodech řešení získané pomocí Eulerovy metody, a to zeleně pro maximální hodnotu pravé strany a červeně pro minimální hodnotu pravé strany. Na Obrázku 39 vlevo je znázorněna pravá strana diferenciální rovnice (43), která je opět jednoznačná.



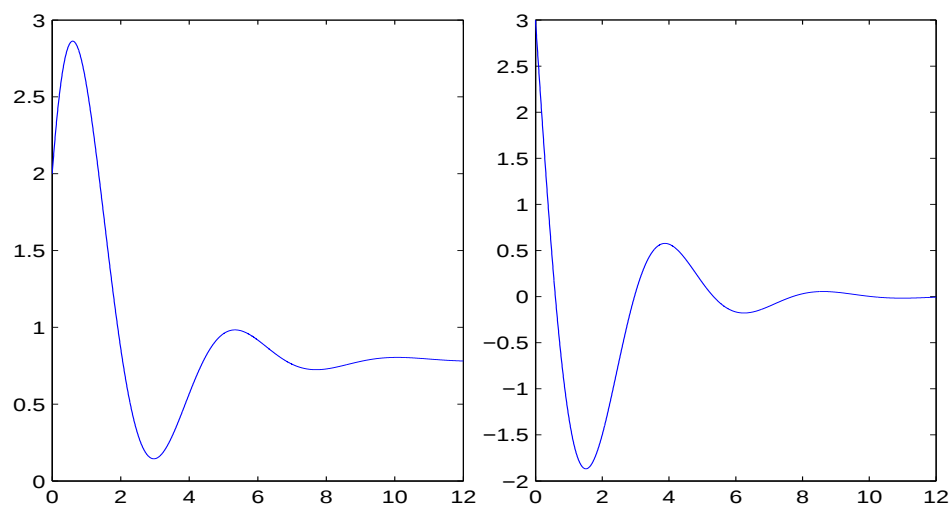
Obrázek 39

Na Obrázku 40 je znázorněn, a to modře, výběr řešení pomocí střední hodnoty pro jednotlivé pravé strany rovnice a inkluze. Jelikož pravá strana diferenciální rovnice (43) je jednoznačná, výběr je tu dán jednoznačně, na rozdíl od pravé strany diferenciální inkluze (44), kde musíme zvolit strategii výběru selekce, v tomto případě výběr pomocí střední hodnoty.



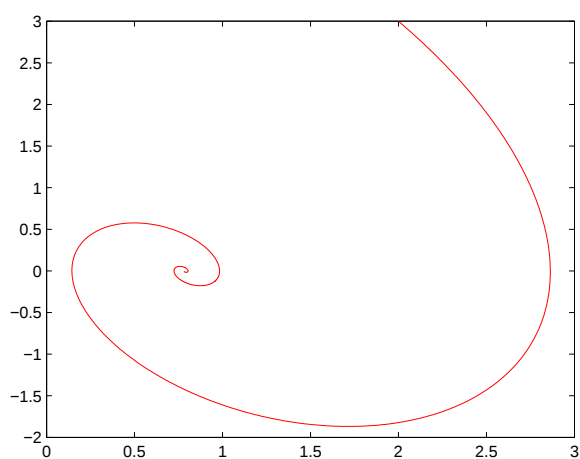
Obrázek 40

Na Obrázku 41 je znázorněna vlevo první složka řešení, vpravo druhá složka řešení pomocí strategie výběru pomocí střední hodnoty.



Obrázek 41: vlevo 1. složka řešení, vpravo 2. složka řešení

Na závěr je na posledním Obrázku 42 znázorněn fázový portrét složek nalezených řešení, přičemž na vodorovnou osu je nanesena první složka řešení a na svislou druhá složka řešení.

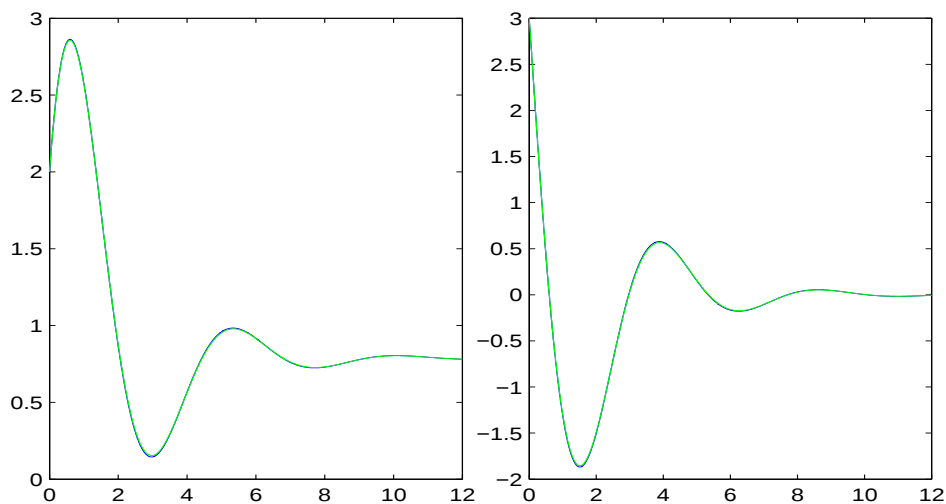


Obrázek 42: fázový portrét řešení

Tento příklad také vyřeším pomocí klasické metody Runge–Kutta pro řešení diferenciálních inkluzí se strategií výběru pomocí střední hodnoty. Použiji M-soubor, který jsem sestavila v Matlabu a který je uveden v odstavci 4. 6. Do Matlabu zadám

```
a=1;
b=2;
F.min=@(x,y)[y(2);-a*y(2)-b*y(1)-sin(x)];
F.max=@(x,y)[y(2);-a*y(2)-b*y(1)+sin(x)+pi];
A=[0 0.5 0.5 1];
B=[0 0 0 0;0.5 0 0 0;0 0.5 0 0;0 0 1 0];
W=[1/6 1/3 1/3 1/6];
[y]=RungeKuttaInkluze(F,0,12,[2 3],0.005,3,A,B,W);
```

Na Obrázku 43 je pro srovnání znázorněno zeleně řešení získané klasickou metodou Runge–Kutta a modře řešení získané Eulerovou metodou. Vidíme, že se tato řešení příliš neliší.



Obrázek 43: vlevo 1. složka řešení, vpravo 2. složka řešení



Příklad 13. Vyřešte diferenciální rovnici

$$y''(x) + a \operatorname{sgn}(y'(x)) + b y(x) = p(x),$$

kde $a, b \in \mathbb{R}$, $y(0) = 0$, $y'(0) = \pi$, $x \in \langle 0, 8\pi \rangle$. Zvolte např. $p(x) = c \cos(x)$, $c \in \mathbb{R}$.

Řešení: Nejprve zadanou diferenciální rovnici přepíšeme na soustavu:

$$y'_1 = y_2 \quad (45)$$

$$y'_2 = c \cos x - a \operatorname{sgn}(y_2) - b y_1. \quad (46)$$

Jelikož je pravá strana diferenciální rovnice (46) kvůli funkci signum nespojitá, nahradíme ji mnohoznačnou funkcí Sgn stejně jako jsme to provedli v Příkladě 7. Dostaneme tak ale diferenciální inkluzi. Řešíme tedy soustavu

$$y'_1 = y_2 \quad (47)$$

$$y'_2 \in c \cos x - a \operatorname{Sgn}(y_2) - b y_1 \quad (48)$$

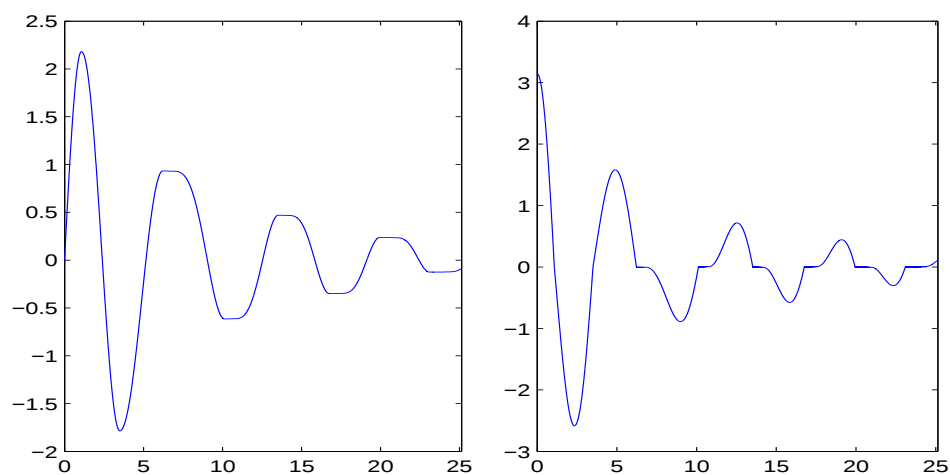
s počátečními podmínkami $y_1(0) = 0$, $y_2(0) = \pi$. Nyní za konstanty a, b, c zvolíme konkrétní hodnoty.

- a) Nejprve volíme např. $a = 1$, $b = 2$, $c = 1$. Tuto soustavu vyřešíme Eulerovou metodou pro řešení diferenciálních inkluzí s krokem $h = 0.005$. Zvolíme např. strategii výběru selekce pomocí minimální normy. Tento příklad vyřešíme pomocí mnou sestaveného M-souboru z odstavce 4.5: Do Matlabu zadáme

```
a=1;
b=2;
c=1;
F.min=@(x,y) [y(2); c*cos(x)-b*y(1)+a];
F.max=@(x,y) [y(2); c*cos(x)-b*y(1)-a];
[y]=EulerInkluzeSignum(F,0,8*pi,[0 pi],0.005,5);
```

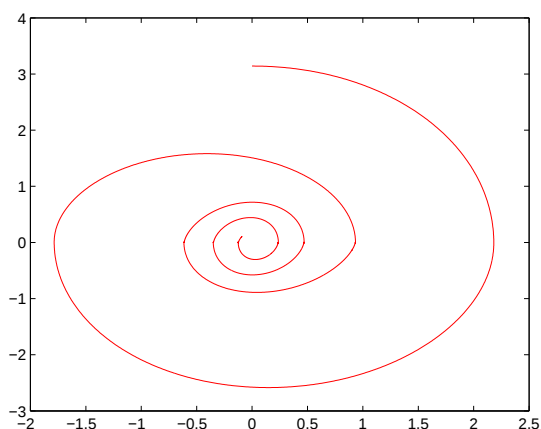
Řešení této úlohy je znázorněno na následujících obrázcích:

Na Obrázku 44 je znázorněna vlevo první složka řešení, vpravo druhá složka řešení pomocí strategie výběru s minimální normou.



Obrázek 44: vlevo 1. složka řešení, vpravo 2. složka řešení

Na Obrázku 45 je znázorněn fázový portrét složek nalezených řešení, přičemž na vodorovnou osu je nanesena první složka řešení a na svislou druhá složka řešení.

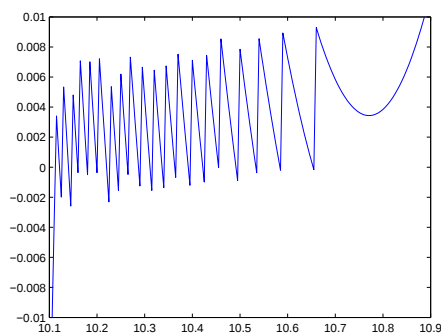


Obrázek 45: fázový portrét řešení

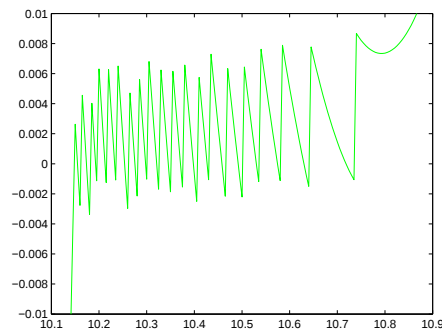
Kdybychom tento příklad řešili pomocí klasické metody Runge–Kutta pro inkluze, na první pohled bychom dostali podobné řešení. Při přiblížení se však na některých úsecích liší a to hlavně druhá složka řešení. Řešení bychom získali pomocí mnou sestaveného M-souboru z odstavce 4. 7 zadáním do Matlabu

```
a=1;
b=2;
c=1;
F.min=@(x,y) [y(2);c*cos(x)-b*y(1)+a];
F.max=@(x,y) [y(2);c*cos(x)-b*y(1)-a];
A=[0 0.5 0.5 1];
B=[0 0 0 0;0.5 0 0 0;0 0.5 0 0;0 0 1 0];
W=[1/6 1/3 1/3 1/6];
[y]=RungeKuttaInkluzeSignum(F,0,8*pi,[0 pi],0.005,5,A,B,W);
```

Na Obrázcích 46 a 47 jsou pro srovnání znázorněny výřezy druhé složky řešení nalezené nejprve pomocí Eulerovy metody a poté pomocí klasické metody Runge–Kutta.



Obrázek 46: Aproximace $y'(\cdot)$ získaná Eulerovou metodou s $h = 0.005$

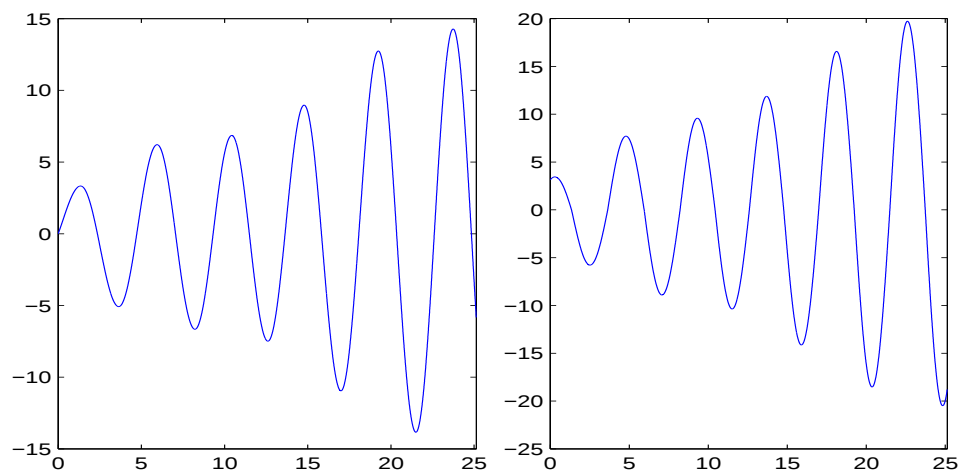


Obrázek 47: Aproximace $y'(\cdot)$ získaná klasickou metodou Runge–Kutta s $h = 0.005$

- b) Zvolme např. $a = -1$, $b = 2$, $c = 1$. Tuto úlohu vyřešíme pomocí Eulerovy metody pro řešení diferenciálních inkluzí s krokem $h = 0.005$, strategie výběru minimální selekce a pomocí M-souboru z odstavce 4.5. Při strategii výběru minimální selekce postupujeme tak, že v bodech, ve kterých funkce pravé strany dané diferenciální inkluze nabývá více hodnot vybereme minimální hodnotu. Výběru minimální selekce odpovídá hodnota *parametr* číslu 1. Do Matlabu zadáme

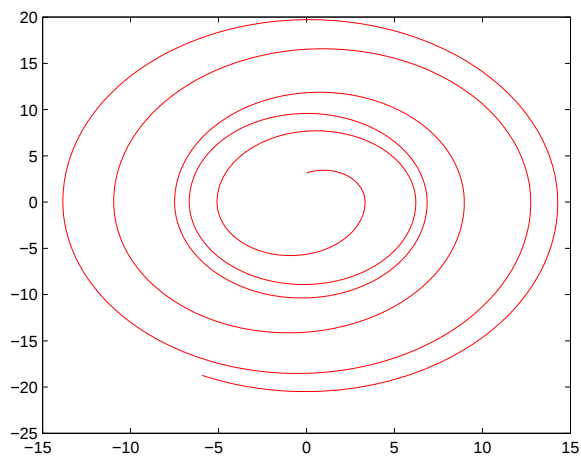
```
a=-1;
b=2;
c=1;
F.min=@(x,y) [y(2); c*cos(x)-b*y(1)+a];
F.max=@(x,y) [y(2); c*cos(x)-b*y(1)-a];
[y]=EulerInkluzeSignum(F,0,8*pi,[0 pi],0.005,1);
```

Na Obrázku 48 je znázorněna vlevo první složka řešení, vpravo druhá složka řešení pomocí strategie výběru minimální selekce.



Obrázek 48: vlevo 1. složka řešení, vpravo 2. složka řešení

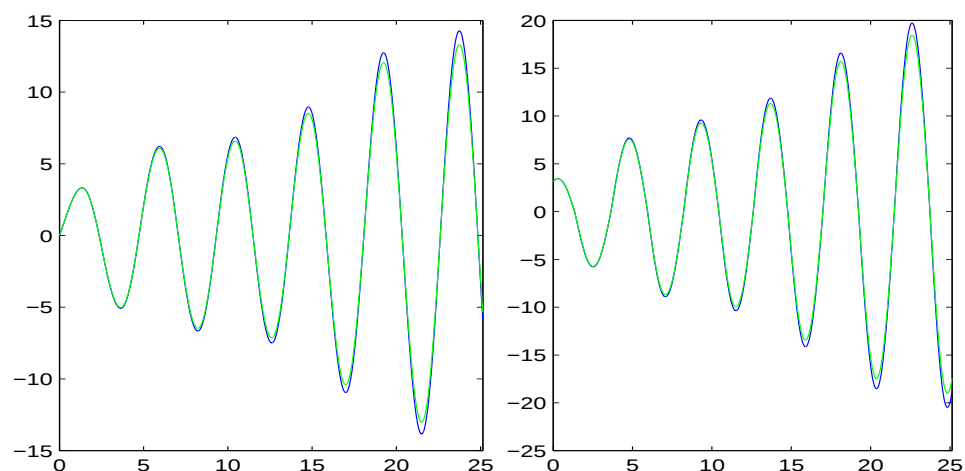
Na Obrázku 49 je znázorněn fázový portrét složek nalezených řešení, přičemž na vodorovnou osu je nanesena první složka řešení a na svislou druhá složka řešení.



Obrázek 49: fázový portrét řešení

Kdybychom tento příklad řešili klasickou metodou Runge–Kutta se stejnou strategií výběru a s krokem $h = 0.005$, dostali bychom řešení znázorněné na

Obrázku 50 zelenou barvou, kde je společně s ním znázorněno modře také řešení získané Eulerovou metodou se stejným krokem.



Obrázek 50: vlevo 1. složka řešení, vpravo 2. složka řešení



Poznámka 15. Z obrázků 44 a 48 vidíme, že se řešení tlumí pro $a > 0$ a amplituda řešení narůstá pro $a < 0$.

Příklad 14. Vyřešte diferenciální rovnici

$$y''(x) + a \operatorname{sgn}(y'(x)) + b \sin y(x) = p(x),$$

kde $a, b \in \mathbb{R}$, $y(0) = 0$, $y'(0) = \pi$, $x \in \langle 0, 8\pi \rangle$. Zvolte např. $p(x) = c \cos(x)$, $c \in \mathbb{R}$.

Řešení: Nejdříve opět nahradíme funkci signum kvůli její nespojitosti funkcí Sgn dle Příkladu 7. Dostaneme tak diferenciální inkluzi, kterou přepíšeme na soustavu:

$$y_1' = y_2 \tag{49}$$

$$y_2' \in c \cos x - a \operatorname{Sgn}(y_2) - b \sin y_1. \tag{50}$$

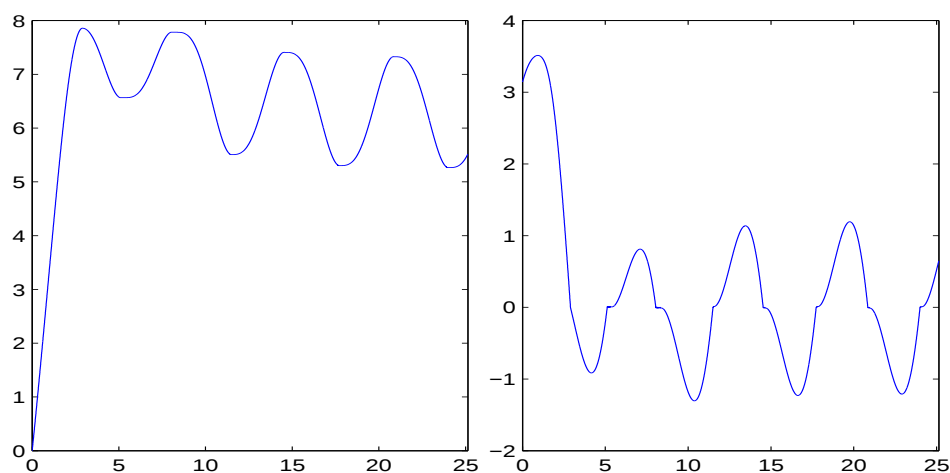
Opět budeme volit za a, b, c různé hodnoty.

a) Nejprve zvolíme $a = 1.2$, $b = 0.2$, $c = 2$. Tuto soustavu vyřešíme Eulerovou metodou pro řešení diferenciálních inkluzí s krokem $h = 0.005$. Zvolíme např. strategii výběru selekce pomocí minimální variace. Tento příklad vyřešíme pomocí M-souboru z odstavce 4.5. Strategii výběru selekce pomocí minimální variace odpovídá hodnota *parametr* číslu 6. Do Matlabu zadáme

```
a=1.2;b=0.2;c=2;
F.min=@(x,y)[y(2);c*cos(x)-b*sin(y(1))+a];
F.max=@(x,y)[y(2);c*cos(x)-b*sin(y(1))-a];
[y]=EulerInkluzeSignum(F,0,8*pi,[0 pi],0.005,6);
```

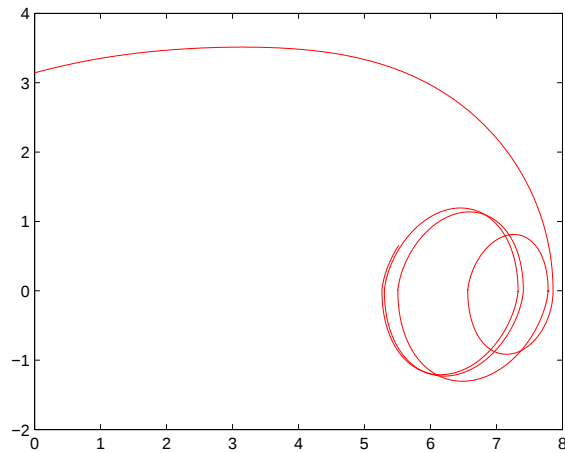
Řešení této úlohy je znázorněno na následujících obrázcích:

Na Obrázku 51 je znázorněna vlevo první složka řešení, vpravo druhá složka řešení pomocí strategie výběru s minimální variací.



Obrázek 51: vlevo 1. složka řešení, vpravo 2. složka řešení

Na Obrázku 52 je znázorněn fázový portrét složek nalezených řešení, přičemž na vodorovnou osu je nanesena první složka řešení a na svislou druhá složka řešení.

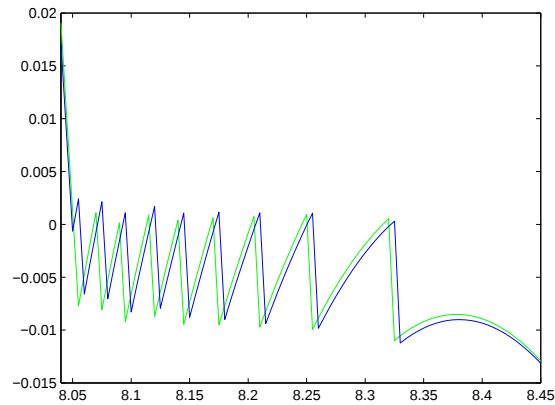


Obrázek 52: fázový portrét řešení

Kdybychom tento příklad řešili pomocí klasické metody Runge–Kutta pro inkluze se stejným výběrem jako u Eulerovy metody, na první pohled bychom dostali podobné řešení. Při přiblížení se však na některých úsecích liší a to hlavně druhá složka řešení. Řešení bychom získali pomocí mnou sestaveného M-souboru z odstavce 4. 7 zadáním do Matlabu

```
a=1.2;b=0.2;c=2;
F.min=@(x,y)[y(2);c*cos(x)-b*sin(y(1))+a];
F.max=@(x,y)[y(2);c*cos(x)-b*sin(y(1))-a];
A=[0 0.5 0.5 1];
B=[0 0 0 0;0.5 0 0 0;0 0.5 0 0;0 0 1 0];
W=[1/6 1/3 1/3 1/6];
[y]=RungeKuttaInkluzeSignum(F,0,8*pi,[0 pi],0.005,6,A,B,W);
```

Na Obrázku 53 je pro srovnání znázorněn výřez druhé složky řešení nalezené nejprve pomocí Eulerovy metody (modře) a poté pomocí klasické metody Runge–Kutta (zeleně).



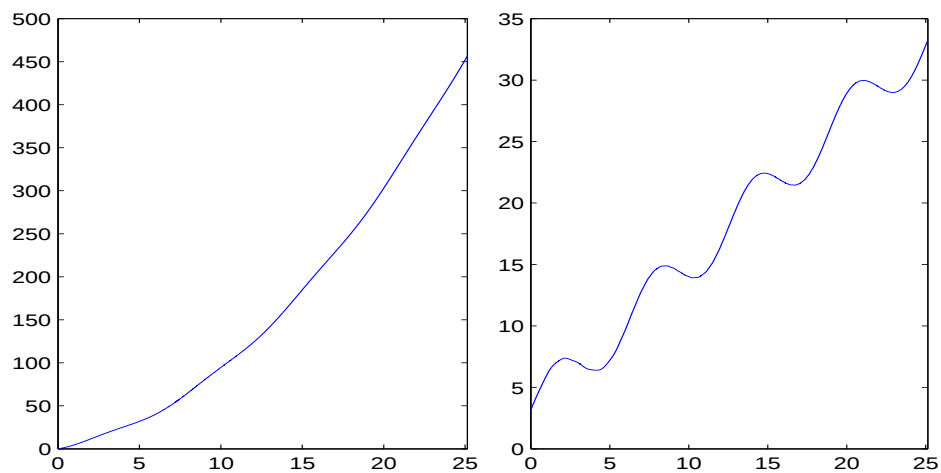
Obrázek 53

- b) Nyní zvolme $a = -1.2$, $b = 0.2$, $c = 2$. Tuto soustavu opět vyřešíme Eulero-vou metodou pro řešení diferenciálních inkluzí s krokem $h = 0.005$. Zvolíme např. strategii výběru maximální selekce. Při strategii výběru maximální selekce postupujeme tak, že v bodech, ve kterých funkce pravé strany dané diferenciální inkluze nabývá více hodnot vybereme maximální hodnotu. Výběru maximální selekce odpovídá hodnota *parametr* číslu 2. Tento příklad vyřešíme pomocí M-souboru z odstavce 4.5: Do Matlabu zadáme

```
a=-1.2;b=0.2;c=2;
F.min=@(x,y) [y(2);c*cos(x)-b*sin(y(1))+a];
F.max=@(x,y) [y(2);c*cos(x)-b*sin(y(1))-a];
[y]=EulerInkluzeSignum(F,0,8*pi,[0 pi],0.005,2);
```

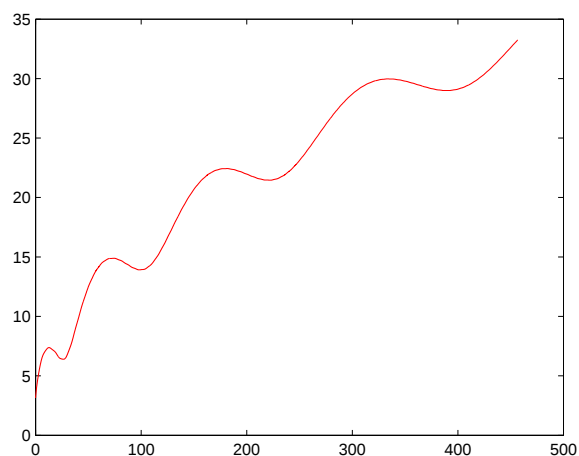
Řešení této úlohy je znázorněno na následujících obrázcích:

Na Obrázku 54 je znázorněna vlevo první složka řešení, vpravo druhá složka řešení pomocí strategie výběru maximální selekce.



Obrázek 54: vlevo 1. složka řešení, vpravo 2. složka řešení

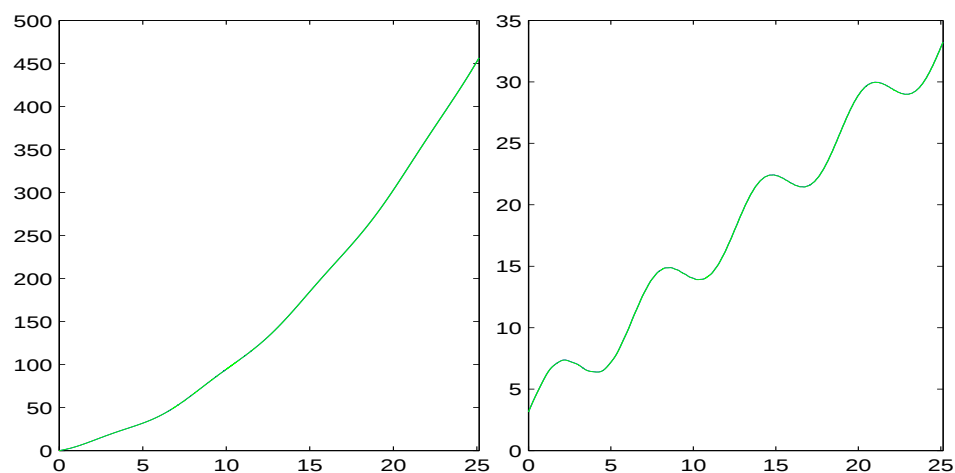
Na Obrázku 55 je znázorněn fázový portrét složek nalezených řešení, přičemž na vodorovnou osu je nanesena první složka řešení a na svislou druhá složka řešení.



Obrázek 55: fázový portrét řešení

Kdybychom tento příklad řešili pomocí klasické metody Runge–Kutta se stejným výběrem selekce a stejnou délkou kroku, dostali bychom téměř

identické řešení, které je společně s řešením získaným Eulerovou metodou znázorněno na následujícím obrázku:



Obrázek 56: modře řešení získané Eulerovou metodou, zeleně řešení získané klasickou metodou Runge–Kutta, vlevo 1. složka řešení, vpravo 2. složka řešení



Poznámka 16. U řešení Příkladů 12, 13 a 14 je vidět, že volba Eulerovy metody je pro získání řešení dostačující. Klasickou metodou Runge–Kutta, která je výpočetně náročnější, jsme dostali téměř identická řešení.

4 M-soubory

4.1 M-soubor pro Eulerovu explicitní metodu

```
function [y]=Euler(a,b,ni,f,h)
%vstup:a...levý krajní bod intervalu
%      b...pravý krajní bod intervalu
%      ni...hodnota počáteční podmínky
%      f...pravá strana rovnice  $y'=f(x,y)$  zadaná jako
%          funkce dvou proměnných ve tvaru @(x,y) a tvar
%          funkce
%      h...integrační krok
%výstup:y..vektor, jehož složky jsou tvořeny prvky
%         posloupnosti přibližných hodnot hledaného
%         řešení

if (nargin()~=5)
    error('nezadány všechny parametry')
end
if (h<=0)
    error('zadej kladný krok')
end
x=a:h:b;
if (x(end)~=b)
    error('zvol jiné h - body nejsou ekvidistantní')
end
y=zeros(length(x),1);y(1)=ni;
for i=2:length(x)
    y(i)=y(i-1)+h*(f(x(i-1),y(i-1)));
end
graf=plot(x,y);
```

4.2 M-soubor pro klasickou metodu Runge–Kutta

```
function [y]=RungeKutta4(a,b,ni,f,h)

%vstup:a...levý krajní bod intervalu
%      b...pravý krajní bod intervalu
%      ni...hodnota počáteční podmínky
%      f...pravá strana rovnice  $y'=f(x,y)$  zadaná jako
%          funkce dvou proměnných ve tvaru @(x,y) a tvar
%          funkce
%      h...integrační krok
%výstup:y..vektor, jehož složky jsou tvořeny prvky
%         posloupnosti přibližných hodnot hledaného
%         řešení
if (nargin()~=5)
    error('nezadány všechny parametry')
end
if (h<=0)
    error('zadej kladný krok')
end
x=a:h:b;y=zeros(length(x),1);y(1)=ni;
if (x(end)~=b)
    error('zvol jiné h - body nejsou ekvidistantní')
end
for i=2:length(x)
    k1=f(x(i-1),y(i-1));
    k2=f(x(i-1)+h/2,y(i-1)+h*k1/2);
    k3=f(x(i-1)+h/2,y(i-1)+h*k2/2);
    k4=f(x(i-1)+h,y(i-1)+h*k3);
    y(i)=y(i-1)+h*k1/6+h*k2/3+h*k3/3+h*k4/6;
end
```

```
graf=plot(x,y);
```

4.3 M-soubor pro Eulerovu explicitní metodu pro řešení soustav diferenciálních rovnic

```
function [y]=EulerSoustavy(a,b,ni,f,h)

%vstup:a...levý krajní bod intervalu
%      b...pravý krajní bod intervalu
%      ni..vektor počátečních podmínek
%      f...pravé strany soustavy  $y'=f(x,y)$  zadané jako
%          funkce dvou proměnných ve tvaru @(x,y)[ a tvar
%          funkce ve formě vektoru, jednotlivé pravé strany
%          odděleny středníkem]
%      h...integrační krok
%výstup:y..matice, jejíž sloupce jsou tvořeny jednotlivými
%         složkami hledaného řešení, tj. každý sloupec
%         obsahuje posloupnost přibližných hodnot hledané
%         složky řešení
if (nargin()~=5)
    error('nezadány všechny parametry')
end
if (h<=0)
    error('zadej kladný krok')
end
x=a:h:b;
if (x(end)~=b)
    error('zvol jiné h - body nejsou ekvidistantní')
end
y(1,:)=ni;
for i=2:length(x)
```

```

        y(i,:)=y(i-1,:)+h*f(x(i-1),y(i-1,:))';
    end
    [o,p]=size(y);
    for j=1:p
        subplot(j,p,j);
        plot(x,y(:,j));
        xlabel(sprintf('%d. složka řešení',j ));
    end

```

4.4 M-soubor pro řešení diferenciálních inkluzí Eulerovou explicitní metodou pomocí různých strategií výběru selekce 1

Tento M-soubor najde pomocí explicitní Eulerovy metody a pomocí různých strategií výběru selekce jediné řešení diferenciální inkluze, jejíž pravá strana je mnohoznačná ve všech bodech intervalu, na němž řešení hledáme.

```

function [y]=EulerInkluze(F,a,b,ni,h,parametr)

%vstup:F...pravé strany soustavy y'=f(x,y) zadané jako
%      funkce více proměnných ve tvaru @(x,y)[ a tvar
%      funkce ve formě vektoru, jednotlivé pravé strany
%      odděleny středníkem]
%      a...levý krajní bod intervalu
%      b...pravý krajní bod intervalu
%      ni..vektor počátečních podmínek
%      h...integrační krok
%      parametr..jeho hodnotu volíme dle toho, jakou strategii
%      výběru selekce zvolíme
%      parametr=1...minimální selekce
%      parametr=2...maximální selekce
%      parametr=3...střední hodnota

```

```

%      parametr=4...náhodný výběr
%      parametr=5...min.norma
%      parametr=6...min.variace
%      parametr=7...výběr selekce s respektováním referenční
%      trajektorie
%      ...ref.trajektorii zadej po vyzvání jako funkci
%      proměnné x, tedy ve tvaru @(x) a tvar funkce
%výstup:y..matice, jejíž sloupce jsou tvořeny jednotlivými
%      složkami hledaného řešení, tj. každý sloupec
%      obsahuje posloupnost přibližných hodnot hledané
%      složky řešení
y(1,:)=ni;x=a:h:b;n=length(x);
if (nargin()~=6)
    error('nezadány všechny parametry')
end
if (x(end)~=b)
    error('zvol jiné h - body nejsou ekvidistantní')
end
if (h<=0)
    error('zadej kladný krok')
end
if parametr~= [1 2 3 4 5 6 7]
    error('zadej parametr jako číslo 1-7')
end
if parametr==7
    reftraj=input('zadej referenční trajektorii: ')
end
for i=1:n
    gmin(i,:)=F.min(x(i),y(i,:));
    gmax(i,:)=F.max(x(i),y(i,:));

```

```

if parametr==1
% minimální selekce
    pom(i,:)=gmin(i,:);
end
if parametr==2
% maximální selekce
    pom(i,:)=gmax(i,:);
end
if parametr==3
% střední hodnota
    pom(i,:)=(gmax(i,:)+gmin(i,:))/2;
end
if parametr==4
% náhodný výběr
pom(i,:)=gmin(i,:)+(gmax(i,:)-gmin(i,:)).*rand(1,length(ni));
end
if parametr==5
% min.norma
    if gmin(i,1)>0
        pom(i,1)=gmin(i,1);
    end
    if gmax(i,1)<0
        pom(i,1)=gmax(i,1);
    end
    if gmax(i,1)>=0&gmin(i,1)<=0
        pom(i,1)=0;
    end
    if gmin(i,2)>0
        pom(i,2)=gmin(i,2);
    end
end

```

```

    if gmax(i,2)<0
        pom(i,2)=gmax(i,2);
    end
    if gmax(i,2)>=0&gmin(i,2)<=0
        pom(i,2)=0;
    end
end
if parametr==6
%min.variace
    if i==1
        if gmin(i,1)>0
            pom(i,1)=gmin(i,1);
        end
        if gmax(i,1)<0
            pom(i,1)=gmax(i,1);
        end
        if gmax(i,1)>=0&gmin(i,1)<=0
            pom(i,1)=0;
        end
        if gmin(i,2)>0
            pom(i,2)=gmin(i,2);
        end
        if gmax(i,2)<0
            pom(i,2)=gmax(i,2);
        end
        if gmax(i,2)>=0&gmin(i,2)<=0
            pom(i,2)=0;
        end
    end
end
if i>1

```

```

    if gmin(i,1)==gmax(i,1)
        pom(i,1)=gmin(i,1);
    else
        if abs(gmax(i,1)-pom(i-1,1))>abs(gmin(i,1)-pom(i-1,1))
            pom(i,1)=gmin(i,1);
        end
        if abs(gmax(i,1)-pom(i-1,1))<abs(gmin(i,1)-pom(i-1,1))
            pom(i,1)=gmax(i,1);
        end
        if abs(gmax(i,1)>=pom(i-1,1))&abs(gmin(i,1)<=pom(i-1,1))
            pom(i,1)=pom(i-1,1);
        end
    end
    if abs(gmax(i,2)-pom(i-1,2))>abs(gmin(i,2)-pom(i-1,2))
        pom(i,2)=gmin(i,2);
    end
    if abs(gmax(i,2)-pom(i-1,2))<abs(gmin(i,2)-pom(i-1,2))
        pom(i,2)=gmax(i,2);
    end
    if abs(gmax(i,2)>=pom(i-1,2))&abs(gmin(i,2)<=pom(i-1,2))
        pom(i,2)=pom(i-1,2);
    end
end
end
if parametr==7
%referencni trajektorie
    pom1=y(i,:)+h*gmin(i,:);
    pom2=y(i,:)+h*gmax(i,:);
    if pom1(1)>0
        pom(i,1)=gmin(i,1);
    end
end

```

```

end
if pom2(1)<0
    pom(i,1)=gmax(i,1);
end
if pom2(1)>=0&pom1(1)<=0
    pom(i,1)=-y(i,1)/h;
end
if pom1(2)>0
    pom(i,2)=gmin(i,2);
end
if pom2(2)<0
    pom(i,2)=gmax(i,2);
end
if pom2(2)>=0&pom1(2)<=0
    pom(i,2)=-y(i,2)/h;
end
end
y(i+1,:)=y(i,:)+h*pom(i,:);
end

figure(1)
subplot(1,2,1)
plot(x(1:n),gmin(:,1),'r'),
hold on,
plot(x(1:n),gmax(:,1),'g'),
subplot(1,2,2)
plot(x(1:n),gmin(:,2),'r'),
hold on,
plot(x(1:n),gmax(:,2),'g'),
figure(2)

```

```

subplot(1,2,1)
plot(x(1:n),gmin(:,1),'r')
hold on,
plot(x(1:n),gmax(:,1),'g')
hold on
plot(x(1:n),pom(:,1),'b')
subplot(1,2,2)
plot(x(1:n),gmin(:,2),'r')
hold on,
plot(x(1:n),gmax(:,2),'g')
hold on
plot(x(1:n),pom(:,2),'b')
figure(3)
subplot(1,2,1)
plot(x(1:n),y(1:n,1))
title('prvni slozka řešení')
subplot(1,2,2)
plot(x(1:n),y(1:n,2))
title('druha slozka řešení')
figure(4)
plot(y(:,1),y(:,2),'r')
title('fazovy portret řešení')

```

4.5 M-soubor pro řešení diferenciálních inkluzí Eulerovou explicitní metodou pomocí různých strategií výběru selekce 2

Tento M-soubor najde pomocí explicitní Eulerovy metody a pomocí různých strategií výběru selekce jediné řešení diferenciální inkluze, jejíž pravá strana obsahuje funkci signum.

```
function [y]=EulerInkluzeSignum(F,a,b,ni,h,parametr)
```

```

%vstup:F...pravé strany soustavy y'=f(x,y) zadané jako
%      funkce více proměnných ve tvaru @(x,y) [ a tvar
%      funkce ve formě vektoru, jednotlivé pravé strany
%      odděleny středníkem]
%      a...levý krajní bod intervalu
%      b...pravý krajní bod intervalu
%      ni..vektor počátečních podmínek
%      h...integrační krok
%      parametr..jeho hodnotu volíme dle toho, jakou strategii
%      výběru selekce zvolíme
%      parametr=1...minimální selekce
%      parametr=2...maximální selekce
%      parametr=3...střední hodnota
%      parametr=4...náhodný výběr
%      parametr=5...min.norma
%      parametr=6...min.variace
%      parametr=7...výběr selekce s respektováním referenční
%      trajektorie
%      ...ref.trajektorii zadej po vyzvání jako funkci
%      proměnné x, tedy ve tvaru @(x) a tvar funkce
%výstup:y..matice, jejíž sloupce jsou tvořeny jednotlivými
%      složkami hledaného řešení, tj. každý sloupec
%      obsahuje posloupnost přibližných hodnot hledané
%      složky řešení
if (nargin()~=6)
    error('nezadány všechny parametry')
end
if (h<=0)
    error('zadej kladný krok')

```

```

end
x=a:h:b;y(1,:)=ni';n=length(x);epsilon=h/10;
if (x(end)~=b)
    error('zvol jiné h - body nejsou ekvidistantní')
end
if parametr~=[1 2 3 4 5 6 7]
    error('zadej parametr jako číslo 1-7')
end
if parametr==7
    reftraj=input('zadej referenční trajektorii: ')
end
for i=1:n
    gmin(i,:)=F.min(x(i),y(i,:));
    gmax(i,:)=F.max(x(i),y(i,:));
    if abs(y(i,2))<=epsilon
        if parametr==1
            % minimální selekce
            pom(i,:)=gmin(i,:);
        end
        if parametr==2
            % maximální selekce
            pom(i,:)=gmax(i,:);
        end
        if parametr==3
            % střední hodnota
            pom(i,:)=(gmax(i,:)+gmin(i,:))/2;
        end
        if parametr==4
            % náhodný výběr
            pom(i,:)=gmin(i,:)+(gmax(i,:)-gmin(i,:)).*rand(1,length(ni));
        end
    end
end

```

```

end
if parametr==5
% min.norma
    if gmin(i,1)>0
        pom(i,1)=gmin(i,1);
    end
    if gmax(i,1)<0
        pom(i,1)=gmax(i,1);
    end
    if gmax(i,1)>=0&gmin(i,1)<=0
        pom(i,1)=0;
    end
    if gmin(i,2)>0
        pom(i,2)=gmin(i,2);
    end
    if gmax(i,2)<0
        pom(i,2)=gmax(i,2);
    end
    if gmax(i,2)>=0&gmin(i,2)<=0
        pom(i,2)=0;
    end
end
if parametr==6
%min.variace
    if i==1
        if gmin(i,1)>0
            pom(i,1)=gmin(i,1);
        end
        if gmax(i,1)<0
            pom(i,1)=gmax(i,1);
        end
    end
end

```

```

        end
        if gmax(i,1)>=0&gmin(i,1)<=0
            pom(i,1)=0;
        end
        if gmin(i,2)>0
            pom(i,2)=gmin(i,2);
        end
        if gmax(i,2)<0
            pom(i,2)=gmax(i,2);
        end
        if gmax(i,2)>=0&gmin(i,2)<=0
            pom(i,2)=0;
        end
    end
    if i>1
        if abs(gmax(i,1)-pom(i-1,1))>abs(gmin(i,1)-pom(i-1,1))
            pom(i,1)=gmin(i,1);
        end
        if abs(gmax(i,1)-pom(i-1,1))<abs(gmin(i,1)-pom(i-1,1))
            pom(i,1)=gmax(i,1);
        end
        if abs(gmax(i,1)>=pom(i-1,1))&abs(gmin(i,1)<=pom(i-1,1))
            pom(i,1)=pom(i-1,1);
        end
        if abs(gmax(i,2)-pom(i-1,2))>abs(gmin(i,2)-pom(i-1,2))
            pom(i,2)=gmin(i,2);
        end
        if abs(gmax(i,2)-pom(i-1,2))<abs(gmin(i,2)-pom(i-1,2))
            pom(i,2)=gmax(i,2);
        end
    end
end

```

```

        if abs(gmax(i,2))>=pom(i-1,2)&&abs(gmin(i,2))<=pom(i-1,2)
            pom(i,2)=pom(i-1,2);
        end
    end
end
end
if parametr==7
    %referencni trajektorie
    pom1=y(i,:)+h*gmin(i,:);
    pom2=y(i,:)+h*gmax(i,:);
    if pom1(1)>0
        pom(i,1)=gmin(i,1);
    end
    if pom2(1)<0
        pom(i,1)=gmax(i,1);
    end
    if pom2(1)>=0&pom1(1)<=0
        pom(i,1)=-y(i,1)/h;
    end
    if pom1(2)>0
        pom(i,2)=gmin(i,2);
    end
    if pom2(2)<0
        pom(i,2)=gmax(i,2);
    end
    if pom2(2)>=0&pom1(2)<=0
        pom(i,2)=-y(i,2)/h;
    end
end
end
if y(i,2)<-epsilon

```

```

        pom(i,:)=gmin(i,:);
    end
    if y(i,2)>epsilon
        pom(i,:)=gmax(i,:);
    end
    y(i+1,:)=y(i,:)+h*pom(i,:);
end
figure(1)
subplot(1,2,1)
plot(x(1:n),gmin(:,1),'r'),
hold on,
plot(x(1:n),gmax(:,1),'g'),
subplot(1,2,2)
plot(x(1:n),gmin(:,2),'r'),
hold on,
plot(x(1:n),gmax(:,2),'g'),
figure(2)
subplot(1,2,1)
plot(x(1:n),gmin(:,1),'r')
hold on,
plot(x(1:n),gmax(:,1),'g')
hold on
plot(x(1:n),pom(:,1),'b')
subplot(1,2,2)
plot(x(1:n),gmin(:,2),'r')
hold on,
plot(x(1:n),gmax(:,2),'g')
hold on
plot(x(1:n),pom(:,2),'b')
figure(3)

```

```

subplot(1,2,1)
plot(x(1:n),y(1:n,1))
title('prvni slozka řešení')
subplot(1,2,2)
plot(x(1:n),y(1:n,2))
title('druha slozka řešení')
figure(4)
plot(y(:,1),y(:,2),'r')
title('fazovy portret řešení')

```

4.6 M-soubor pro řešení diferenciálních inkluzí metodami Runge–Kutta s různými strategiemi výběru selekce 1

Tento M-soubor řeší diferenciální inkluze s mnohoznačnou pravou stranou pomocí explicitních s-stádiových metod Runge–Kutta a pomocí různých strategií výběru selekce. Metodu Runge–Kutta zadáme pomocí jejích koeficientů do vektorů A a W a matice B.

```

function [y]=RungeKuttaInkluze(F,a,b,ni,h,parametr,A,B,W)

%vstup:F...pravé strany soustavy y'=f(x,y) zadané jako
%      funkce více proměnných ve tvaru @(x,y)[ a tvar
%      funkce ve formě vektoru, jednotlivé pravé strany
%      odděleny středníkem]
%      a...levý krajní bod intervalu
%      b...pravý krajní bod intervalu
%      ni..vektor počátečních podmínek
%      h...integrační krok
%      parametr..jeho hodnotu volíme dle toho, jakou strategii
%      výběru selekce zvolíme
%      parametr=1...minimální selekce
%      parametr=2...maximální selekce

```

```

%      parametr=3...střední hodnota
%      parametr=4...náhodný výběr
%      parametr=5...min.norma
%      parametr=6...min.variace
%      parametr=7...výběr selekce s respektováním referenční
%      trajektorie
%      ...ref.trajektorii zadej po vyzvání jako funkci
%      proměnné x, tedy ve tvaru @(x) a tvar funkce
%      A...vektor, jehož složky alfa_1,...,alfa_s jsou konstanty
%      použité při výpočtu stádií metody Runge-Kutta
%      B...matice, jejíž složky beta_11,...,beta_1s,...beta_ss
%      jsou konstanty použité při výpočtu stádií metody
%      Runge-Kutta
%      W...vektor koeficientů metody Runge-Kutta, který je tvořen
%      hodnotami w_1,...,w_s
%výstup:y..matice, jejíž sloupce jsou tvořeny jednotlivými
%      složkami hledaného řešení, tj. každý sloupec
%      obsahuje posloupnost přibližných hodnot hledané
%      složky řešení
if (nargin()~=9)
    error('nezadány všechny parametry')
end
if (h<=0)
    error('zadej kladný krok')
end
x=a:h:b;
if (x(end)~=b)
    error('zvol jiné h - body nejsou ekvidistantní')
end
if parametr~=[1 2 3 4 5 6 7]

```

```

        error('zadej parametr jako číslo 1-7')
    end
    if parametr==7
        reftraj=input('zadej referenční trajektorii: ')
    end
    y(1,:)=ni;n=length(x);s=length(W);k1=zeros(n,2*s);
    for ki=1:s
        for kj=1:s
            if ki<=kj
                if B(ki,kj)~=0
                    error('Tato metoda není explicitní, zadej explicitní
                        metodu Runge-Kutta')
                end
            end
        end
    end
    end
    for i=1:n
        for k=1:s
            ypom(i,:)=zeros(1,2);
            for kl=1:s
                ypom(i,:)=ypom(i,:)+h*B(k,kl)*k1(i,2*kl-1:2*kl);
            end
            y2(i,2*k-1:2*k)=y(i,:)+ypom(i,:);
            gmin(i,:)=F.min(x(i)+A(k)*h,y2(i,2*k-1:2*k));
            gmax(i,:)=F.max(x(i)+A(k)*h,y2(i,2*k-1:2*k));
            if parametr==1
                % minimální selekce
                pom(i,:)=gmin(i,:);
            end
            if parametr==2

```

```

% maximální selekce
    pom(i,:)=gmax(i,:);
end
if parametr==3
% střední hodnota
    pom(i,:)=(gmax(i,:)+gmin(i,:))/2;
end
if parametr==4
% náhodný výběr
pom(i,:)=gmin(i,:)+(gmax(i,:)-gmin(i,:)).*rand(1,length(ni));
end
if parametr==5
% min.norma
    if gmin(i,1)>0
        pom(i,1)=gmin(i,1);
    end
    if gmax(i,1)<0
        pom(i,1)=gmax(i,1);
    end
    if gmax(i,1)>=0&gmin(i,1)<=0
        pom(i,1)=0;
    end
    if gmin(i,2)>0
        pom(i,2)=gmin(i,2);
    end
    if gmax(i,2)<0
        pom(i,2)=gmax(i,2);
    end
    if gmax(i,2)>=0&gmin(i,2)<=0
        pom(i,2)=0;
    end
end

```

```

        end
    end
    if parametr==6
    %min.variace
        if i==1
            if gmin(i,1)>0
                pom(i,1)=gmin(i,1);
            end
            if gmax(i,1)<0
                pom(i,1)=gmax(i,1);
            end
            if gmax(i,1)>=0&gmin(i,1)<=0
                pom(i,1)=0;
            end
            if gmin(i,2)>0
                pom(i,2)=gmin(i,2);
            end
            if gmax(i,2)<0
                pom(i,2)=gmax(i,2);
            end
            if gmax(i,2)>=0&gmin(i,2)<=0
                pom(i,2)=0;
            end
        end
        if i>1
            if abs(gmax(i,1)-pom(i-1,1))>abs(gmin(i,1)-pom(i-1,1))
                pom(i,1)=gmin(i,1);
            end
            if abs(gmax(i,1)-pom(i-1,1))<abs(gmin(i,1)-pom(i-1,1))
                pom(i,1)=gmax(i,1);
            end
        end
    end
end

```

```

end
if abs(gmax(i,1))>=pom(i-1,1)&abs(gmin(i,1)<=pom(i-1,1))
    pom(i,1)=pom(i-1,1);
end
if abs(gmax(i,2)-pom(i-1,2))>abs(gmin(i,2)-pom(i-1,2))
    pom(i,2)=gmin(i,2);
end
if abs(gmax(i,2)-pom(i-1,2))<abs(gmin(i,2)-pom(i-1,2))
    pom(i,2)=gmax(i,2);
end
if abs(gmax(i,2))>=pom(i-1,2)&abs(gmin(i,2)<=pom(i-1,2))
    pom(i,2)=pom(i-1,2);
end
end
end
if parametr==7
%referencni trajektorie
    pom1=y(i,:)+h*gmin(i,:);
    pom2=y(i,:)+h*gmax(i,:);
    if pom1(1)>0
        pom(i,1)=gmin(i,1);
    end
    if pom2(1)<0
        pom(i,1)=gmax(i,1);
    end
    if pom2(1)>=0&pom1(1)<=0
        pom(i,1)=-y(i,1)/h;
    end
    if pom1(2)>0
        pom(i,2)=gmin(i,2);

```

```

        end
        if pom2(2)<0
            pom(i,2)=gmax(i,2);
        end
        if pom2(2)>=0&pom1(2)<=0
            pom(i,2)=-y(i,2)/h;
        end
    end
    k1(i,2*k-1:2*k)=pom(i,:);
end
pravstran(i,:)=zeros(1,2);
for ks=1:s
    pravstran(i,:)=pravstran(i,:)+W(ks)*h*k1(i,2*ks-1:2*ks);
end
y(i+1,:)=y(i,:)+pravstran(i,:);
end
figure(1)
subplot(1,2,1)
plot(x(1:n),y(1:n,1),'g')
title('prvni slozka řešení')
subplot(1,2,2)
plot(x(1:n),y(1:n,2),'g')
title('druha slozka řešení')
figure(2)
plot(y(:,1),y(:,2),'r')
title('fazovy portret řešení')

```

4.7 M-soubor pro řešení diferenciálních inkluzí metodami Runge–Kutta s různými strategiemi výběru selekce 2

Tento M-soubor najde pomocí explicitní metody Runge–Kutta a pomocí různých strategií výběru selekce jediné řešení diferenciální inkluze, jejíž pravá strana obsahuje funkci signum.

```
function [y]=RungeKuttaInkluzeSignum(F,a,b,ni,h,parametr,A,B,W)

%vstup:F...pravé strany soustavy y'=f(x,y) zadané jako
%      funkce více proměnných ve tvaru @(x,y)[ a tvar
%      funkce ve formě vektoru, jednotlivé pravé strany
%      odděleny středníkem]
%      a...levý krajní bod intervalu
%      b...pravý krajní bod intervalu
%      ni..vektor počátečních podmínek
%      h...integrační krok
%      parametr..jeho hodnotu volíme dle toho, jakou strategii
%              výběru selekce zvolíme
%      parametr=1...minimální selekce
%      parametr=2...maximální selekce
%      parametr=3...střední hodnota
%      parametr=4...náhodný výběr
%      parametr=5...min.norma
%      parametr=6...min.variace
%      parametr=7...výběr selekce s respektováním referenční
%              trajektorie
%      ...ref.trajektorii zadej po vyzvání jako funkci
%      proměnné x, tedy ve tvaru @(x) a tvar funkce
%      A...vektor, jehož složky alfa_1,...,alfa_s jsou konstanty
%      použité při výpočtu stádií metody Runge–Kutta
%      B...matice, jejíž složky beta_11,...,beta_1s,...beta_ss
```

```

%          jsou konstanty použité při výpočtu stádií metody
%          Runge-Kutta
%          W...vektor koeficientů metody Runge-Kutta, který je tvořen
%          hodnotami w_1,...,w_s
%výstup:y..matice, jejíž sloupce jsou tvořeny jednotlivými
%          složkami hledaného řešení, tj. každý sloupec
%          obsahuje posloupnost přibližných hodnot hledané
%          složky řešení
if (nargin()~=9)
    error('nezadány všechny parametry')
end
if (h<=0)
    error('zadej kladný krok')
end
x=a:h:b;
if (x(end)~=b)
    error('zvol jiné h - body nejsou ekvidistantní')
end
if parametr~= [1 2 3 4 5 6 7]
    error('zadej parametr jako číslo 1-7')
end
if parametr==7
    reftraj=input('zadej referenční trajektorii: ')
end
y(1,:)=ni;n=length(x);s=length(W);k1=zeros(n,2*s);epsilon=h/10;
for ki=1:s
    for kj=1:s
        if ki<=kj
            if B(ki,kj)~=0
                error('Tato metoda není explicitní, zadej explicitní

```



```

end
if parametr==5
% min.norma
    if gmin(i,1)>0
        pom(i,1)=gmin(i,1);
    end
    if gmax(i,1)<0
        pom(i,1)=gmax(i,1);
    end
    if gmax(i,1)>=0&gmin(i,1)<=0
        pom(i,1)=0;
    end
    if gmin(i,2)>0
        pom(i,2)=gmin(i,2);
    end
    if gmax(i,2)<0
        pom(i,2)=gmax(i,2);
    end
    if gmax(i,2)>=0&gmin(i,2)<=0
        pom(i,2)=0;
    end
end
end
if parametr==6
% min.variace
    if i==1
        if gmin(i,1)>0
            pom(i,1)=gmin(i,1);
        end
        if gmax(i,1)<0
            pom(i,1)=gmax(i,1);
        end
    end
end

```

```

end
if gmax(i,1)>=0&gmin(i,1)<=0
    pom(i,1)=0;
end
if gmin(i,2)>0
    pom(i,2)=gmin(i,2);
end
if gmax(i,2)<0
    pom(i,2)=gmax(i,2);
end
if gmax(i,2)>=0&gmin(i,2)<=0
    pom(i,2)=0;
end
end
if i>1
    if abs(gmax(i,1)-pom(i-1,1))>abs(gmin(i,1)-pom(i-1,1))
        pom(i,1)=gmin(i,1);
    end
    if abs(gmax(i,1)-pom(i-1,1))<abs(gmin(i,1)-pom(i-1,1))
        pom(i,1)=gmax(i,1);
    end
    if abs(gmax(i,1)>=pom(i-1,1))&abs(gmin(i,1)<=pom(i-1,1))
        pom(i,1)=pom(i-1,1);
    end
    if abs(gmax(i,2)-pom(i-1,2))>abs(gmin(i,2)-pom(i-1,2))
        pom(i,2)=gmin(i,2);
    end
    if abs(gmax(i,2)-pom(i-1,2))<abs(gmin(i,2)-pom(i-1,2))
        pom(i,2)=gmax(i,2);
    end
end

```

```

        if abs(gmax(i,2))>=pom(i-1,2)&&abs(gmin(i,2)<=pom(i-1,2))
            pom(i,2)=pom(i-1,2);
        end
    end
end
if parametr==7
% referencni trajektorie
    pom1=y(i,:)+h*gmin(i,:);
    pom2=y(i,:)+h*gmax(i,:);
    if pom1(1)>0
        pom(i,1)=gmin(i,1);
    end
    if pom2(1)<0
        pom(i,1)=gmax(i,1);
    end
    if pom2(1)>=0&pom1(1)<=0
        pom(i,1)=-y(i,1)/h;
    end
    if pom1(2)>0
        pom(i,2)=gmin(i,2);
    end
    if pom2(2)<0
        pom(i,2)=gmax(i,2);
    end
    if pom2(2)>=0&pom1(2)<=0
        pom(i,2)=-y(i,2)/h;
    end
end
k1(i,2*k-1:2*k)=pom(i,:);
end

```

```

        if y(i,2)<-epsilon
            pom(i,:)=gmin(i,:);
            k1(i,2*k-1:2*k)=pom(i,:);
        end
        if y(i,2)>epsilon
            pom(i,:)=gmax(i,:);
            k1(i,2*k-1:2*k)=pom(i,:);
        end
    end
    pravstran(i,:)=zeros(1,2);
    for ks=1:s
        pravstran(i,:)=pravstran(i,:)+W(ks)*h*k1(i,2*ks-1:2*ks);
    end
    y(i+1,:)=y(i,:)+pravstran(i,:);
end
figure(1)
subplot(1,2,1)
plot(x(1:n),y(1:n,1),'g')
title('prvni slozka řešení')
subplot(1,2,2)
plot(x(1:n),y(1:n,2),'g')
title('druha slozka řešení')
figure(2)
plot(y(:,1),y(:,2),'r')
title('fazovy portret řešení')

```

Závěr

Úkolem této diplomové práce bylo vysvětlit princip některých numerických metod pro řešení diferenciálních rovnic, aplikovat je na diferenciální inkluze a předvést je na konkrétních příkladech tak, aby mohla sloužit např. i jako učební pomůcka studentům, kteří budou mít o tuto tematiku zájem.

Teoretická část práce je doplněna příklady, jejichž řešení jsou provedena některá ručně, většina však pomocí programů, které jsem sestavila v matematickém softwaru Matlab, jež jsou v práci uvedeny v poslední kapitole. Snahou bylo tyto M-soubory odladit, což se doufám podařilo.

V této práci jsou z důvodu jejího rozsahu uvedeny jen základní konvergenční výsledky. Tato problematika je však mnohem obsáhlejší a zasloužila by si důkladnější prozkoumání. Proto se nejen otázkami konvergence, ale i teorií diferenciálních inkluzí budu zabývat v doktorském studiu.

Domnívám se, že cíle práce byly splněny. Při studiu daného tématu jsem se seznámila, a následně i čtenáře, s tím, jak postupovat v této problematice, sestavila programy pro vypracování konkrétních úloh a vše doplnila svými poznatky a potížemi při jejich řešení. Přitom jsem si prohloubila své znalosti a dovednosti nejen v programu Matlab, ale i v typografickém programu $\text{\TeX}$, kterým je tato práce vysázena.

K této práci je přiloženo CD se všemi M-soubory uvedenými v kapitole 4.

Literatura

- [1] Andres, J., Górniewicz, L.: *Topological Fixed Point Principles for Boundary Value Problems*. Kluwer, Dordrecht, 2003.
- [2] Aubin, J. P., Cellina, A.: *Differential Inclusions*. Springer-Verlag, Berlin, 1984.
- [3] Dontchev, A., Lempio, F.: *Difference Methods for Differential Inclusions: A Survey*, SIAM Review, Vol. 34, No. 2, 1992, pp. 263-294.
- [4] Hairer, E., Wanner, G.: *Solving Ordinary Differential Equations II*. Springer-Verlag, New York, 2002.
- [5] Horová, I., Zelinka, J.: *Numerické metody*. Masarykova univerzita, Brno, 2004.
- [6] Kobza, J.: *Numerické metody*. Univerzita Palackého, Olomouc, 1993.
- [7] Kojecká, J., Závodný, M.: *Příklady z diferenciálních rovnic I*. Univerzita Palackého, Olomouc, 2004.
- [8] Přikryl, P., Brandner, M.: *Numerické metody II*. Západočeská univerzita v Plzni, Plzeň, 2000.
- [9] Vitásek, E.: *Základy teorie numerických metod pro řešení diferenciálních rovnic*. Academia, Praha, 1994.