



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

ÚSTAV MATEMATIKY

INSTITUTE OF MATHEMATICS

**NUMERICKÁ ANALÝZA TUHÝCH SYSTÉMŮ
DIFERENCIÁLNÍCH ROVNIC**

NUMERICAL ANALYSIS OF STIFF DIFFERENTIAL EQUATIONS

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

Ondřej Pavelka

VEDOUCÍ PRÁCE

SUPERVISOR

doc. Ing. Petr Tomášek, Ph.D.

BRNO 2019

Zadání bakalářské práce

Ústav: Ústav matematiky
Student: **Ondřej Pavelka**
Studijní program: Aplikované vědy v inženýrství
Studijní obor: Matematické inženýrství
Vedoucí práce: **doc. Ing. Petr Tomášek, Ph.D.**
Akademický rok: 2018/19

Ředitel ústavu Vám v souladu se zákonem č. 111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma bakalářské práce:

Numerická analýza tuhých systémů diferenciálních rovnic

Stručná charakteristika problematiky úkolu:

Předpokládanou náplní práce je studium numerického řešení systémů obyčejných diferenciálních rovnic a specifik numerického řešení tuhých systémů.

Cíle bakalářské práce:

V rámci práce se provede analýza vybraných numerických metod.

Na vybraných úlohách (s reálnými aplikacemi) se v prostředí MATLAB provede diskuze nad získanými výsledky.

Seznam doporučené literatury:

RALSTON, A. Základy numerické matematiky, Academia, Praha, 1978.

ISERLES, A. A first course in the numerical analysis of differential equations, Cambridge University Press, Cambridge, 1996.

BUTCHER, J. C. Numerical methods for ordinary differential equations, Wiley, Chichester, 2003.

Termín odevzdání bakalářské práce je stanoven časovým plánem akademického roku 2018/19.

V Brně, dne 23. 10. 2018



Šlapal

prof. RNDr. Josef Šlapal, CSc.
ředitel ústavu

V. Katolický

doc. Ing. Jaroslav Katolický, Ph.D.
děkan fakulty

Abstrakt

Tato bakalářská práce se zabývá numerickým řešením systémů obyčejných diferenciálních rovnic. V práci jsou nejprve zavedeny a popsány jednotlivé numerické metody určené k řešení diferenciálních rovnic, poté je vyšetřována jejich stabilita. Hlavním cílem je analýza tuhosti vybraných systémů diferenciálních rovnic, výběr vhodné numerické metody a následné řešení v prostředí MATLAB.

Summary

This bachelor's thesis deals with numerical solution of ordinary differential equations. The first part of the thesis introduces and describes the numerical methods for ordinary differential equations. The next part is about stability of numerical methods. The aim of the thesis is to analyze the stiffness of the systems of differential equations, select a suitable numerical method and solve the systems in the MATLAB.

Klíčová slova

obyčejné diferenciální rovnice, numerické řešení diferenciálních rovnic, stabilita, tuhé systémy

Keywords

ordinary differential equation, numerical solution of differential equations, stability, stiff systems

PAVELKA, O. *Numerická analýza tuhých systémů diferenciálních rovnic*. Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, 2019. 42 s. Vedoucí doc. Ing. Petr Tomášek, Ph.D.

Prohlašuji, že jsem bakalářskou práci „Numerická analýza tuhých systémů diferenciálních rovnic“ vypracoval samostatně pod vedením doc. Ing. Petra Tomáška, Ph.D. s použitím materiálů uvedených v seznamu použité literatury.

Ondřej Pavelka

Rád bych tímto poděkoval svému vedoucímu doc. Ing. Petru Tomáškoví, Ph.D. za odborné vedení, cenné připomínky a rady při zpracovávání této bakalářské práce.

Ondřej Pavelka

Obsah

1	Úvod	2
2	Obyčejné diferenciální rovnice	3
2.1	Existence a jednoznačnost	4
3	Numerické metody řešení ODR	5
3.1	Eulerovy metody	5
3.2	Rungeovy-Kuttovy metody	7
3.3	Lineární vícekrokové metody	11
3.3.1	Adams-Bashforthovy metody	12
3.3.2	Metody zpětného derivování	13
4	Stabilita	14
4.1	Stabilita Rungeových-Kuttových metod	17
4.2	Stabilita vícekrokových metod	20
5	Implementace metod v prostředí MATLAB	24
6	Tuhé systémy	26
7	Závěr	35
8	Seznam použité literatury	36
9	Seznam použitých zkratk a symbolů	37
10	Přílohy	38

1 Úvod

Mnoho diferenciálních rovnic není řešitelných analyticky nebo je jejich řešení příliš časově náročné, proto se řeší numericky. Při studiu numerických metod určených k řešení soustav diferenciálních rovnic se můžeme setkat se soustavami, jejichž řešení explicitními metodami je téměř nemožné. Tyto soustavy označujeme za *tuhé* (anglicky *stiff systems of differential equations*). Hlavním cílem této práce je analýza vybraných numerických metod a studium tuhých systémů.

Práce je strukturována následovně. Druhá kapitola uvádí teoretický základ obyčejných diferenciálních rovnic. Třetí kapitola je věnována studiu numerických metod určených k jejich řešení. Ve čtvrté kapitole analyzujeme stabilitu vybraných numerických metod. V kapitole páté se seznámíme s prostředím MATLAB. Poslední část je zaměřena na charakterizaci tuhých systémů. Dále se v ní řeší tuhé problémy z fyzikálního a chemického prostředí a hledá se vhodná metoda jejich řešení. Součástí práce je CD, na kterém nalezneme skripty k jednotlivým úlohám.

2 Obyčejné diferenciální rovnice

Počáteční úlohou obyčejné diferenciální rovnice n -tého řádu (ODR n) rozumíme rovnici

$$y^{(n)} = f(x, y, y', y'', \dots, y^{(n-1)}), \quad (2.1)$$

s podmínkami

$$y(x_0) = \eta_0, \quad y'(x_0) = \eta_1, \quad y''(x_0) = \eta_2, \quad \dots, \quad y^{(n-1)}(x_0) = \eta_{n-1}, \quad (2.2)$$

kde funkce f je definovaná na oblasti $G \subset \mathbb{R}^{n+1}$, $[x_0, \eta_0, \dots, \eta_{n-1}] \in G$, $f : G \rightarrow \mathbb{R}$, $y^{(k)}$ značí k -tou derivaci funkce $y(x)$. Řešením počáteční úlohy na intervalu I rozumíme nalezení funkce $y(x)$, která je spojitá, má spojité derivace do řádu n , vyhovuje diferenciální rovnici ve všech $x \in I$ a všem počátečním podmínkám.

Speciálně pro $n = 1$ získáme obyčejnou diferenciální rovnici 1. řádu (ODR1) zadanou rovnicí a jednou počáteční podmínkou

$$y' = f(x, y), \quad y(x_0) = y_0. \quad (2.3)$$

Počáteční úlohou pro soustavu n diferenciálních rovnic prvního řádu chápeme systém rovnic

$$\begin{aligned} y_1' &= f_1(x, y_1, y_2, \dots, y_n), \\ y_2' &= f_2(x, y_1, y_2, \dots, y_n), \\ &\vdots \\ y_n' &= f_n(x, y_1, y_2, \dots, y_n), \end{aligned}$$

přičemž $y_i = y_i(x)$, doplněný o n počátečních podmínek

$$y_i(x_0) = \eta_i, \quad i = 1, \dots, n.$$

Úlohu můžeme zapsat ve vektorovém tvaru

$$\mathbf{y}' = \mathbf{f}(x, \mathbf{y}), \quad \mathbf{y}(x_0) = \mathbf{y}_0, \quad (2.4)$$

kde

$$\mathbf{y} = \begin{pmatrix} y_1(x) \\ y_2(x) \\ \vdots \\ y_n(x) \end{pmatrix}, \quad \mathbf{f}(x, \mathbf{y}) = \begin{pmatrix} f_1(x, y_1, y_2, \dots, y_n) \\ f_2(x, y_1, y_2, \dots, y_n) \\ \vdots \\ f_n(x, y_1, y_2, \dots, y_n) \end{pmatrix}, \quad \mathbf{y}_0 = \begin{pmatrix} \eta_1(x) \\ \eta_2(x) \\ \vdots \\ \eta_n(x) \end{pmatrix}.$$

Věta 2.1. Každou obyčejnou diferenciální rovnici n -tého řádu (2.1) s počátečními podmínkami (2.2) definovanou na intervalu I lze převést na soustavu n diferenciálních rovnic prvního řádu na stejném intervalu.

Důkaz. Označme $y_1 = y$, $y_2 = y'$, $y_3 = y''$, \dots , $y_n = y^{(n-1)}$, pak

$$\begin{aligned} y_1' &= y_2, \\ y_2' &= y_3, \\ y_3' &= y_4, \\ &\vdots \\ y_{n-1}' &= y_n, \\ y_n &= f(x, y_1, y_2, \dots, y_n). \end{aligned}$$

2.1. EXISTENCE A JEDNOZNAČNOST

Tímto jsme získali soustavu n obyčejných diferenciálních rovnic prvního řádu s počátečními podmínkami $y_1(x_0) = \eta_0, y_2(x_0) = \eta_1, \dots, y_{n-1}(x_0) = \eta_{n-2}, y_n(x_0) = \eta_{n-1}$. Poznamenejme, že první složka systému $y_1(x)$ je za těchto počátečních podmínek řešením počáteční úlohy (2.1) (2.2). \square

Vzhledem k tomu, že numerické metody jsou formulovány především pro počáteční úlohy systému ODR 1. řádu, má tato věta zásadní význam pro řešení úloh s rovnicemi n -tého řádu. Jak bude ilustrováno i na příkladech, řada problémů z technické praxe je primárně modelována rovnicemi vyšších řádů.

2.1 Existence a jednoznačnost

Uvažujme úlohu (2.4). Budeme vyšetřovat, zda řešení úlohy existuje a za jakých podmínek je určeno jednoznačně. Nejprve musíme zadefinovat následující pojem.

Definice 2.2 (Lipschitzovsky spojitá funkce). Funkci $f : I \subset \mathbb{R}^n \rightarrow \mathbb{R}$ nazveme lipschitzovsky spojitou v proměnné x_k , jestliže pro všechny pevně zvolené $x_1, x_2, \dots, x_{k-1}, x_{k+1}, \dots, x_n \exists L > 0$ takové, že pro každé α, β v okolí x_k platí

$$|f(x_1, \dots, x_{k-1}, \alpha, x_{k+1}, \dots, x_n) - f(x_1, \dots, x_{k-1}, \beta, x_{k+1}, \dots, x_n)| \leq L|\alpha - \beta|.$$

Věta 2.3 (Peanova věta). *Uvažujme počáteční úlohu (2.4). Jsou-li všechny funkce $f_i, i = 1, 2, \dots, n$, spojitě v okolí bodu (x_0, \mathbf{y}_0) , pak má úloha (2.4) v okolí x_0 aspoň jedno řešení.*

Věta 2.4 (Picardova věta). *Jsou-li splněny předpoklady předcházející věty a všechny složky $f_i, i = 1, 2, \dots, n$ jsou lipschitzovské v proměnných $y_i, i = 1, 2, \dots, n$ v okolí (x_0, \mathbf{y}_0) , pak má úloha (2.4) řešení určené jednoznačně.*

Picardova věta je silnější než Peanova věta, neboť nám zaručuje jednoznačnost řešení. Důkaz vět můžeme nalézt v [3].

3 Numerické metody řešení ODR

V této části se zaměříme na popis numerických metod určených k řešení počátečních úloh obyčejných diferenciálních rovnic. Základní zdroje pro tuto kapitolu jsou [1], [2], [4] a [8].

Při numerickém řešení počáteční úlohy (2.4) hledáme přibližné hodnoty $\mathbf{y}(x)$ v bodech $x_n, n = 0, 1, \dots, q$, které dostatečně hustě pokrývají interval $\langle a, b \rangle$. Zavedeme dělení

$$a = x_0 < x_1 < \dots < x_{q-1} < x_q = b$$

intervalu $\langle a, b \rangle$. Body x_n označme *uzly*, *délka kroku* $h_n = x_{n+1} - x_n > 0$ je vzdálenost dvou sousedních uzlů. Je-li dělení intervalu takové, že všechny délky kroků jsou stejné, tj. $h_n = h = (b - a)/q$, nazveme dělení *rovnoměrné* a index n vynecháváme. Dále označme \mathbf{y}_n jako aproximaci přesného řešení $\mathbf{y}(x_n)$ v bodě x_n .

Numerická metoda je algoritmus pro výpočet přibližných hodnot $\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3, \dots$ ze zadané rovnice a počátečních podmínek. Numerickou metodu nazveme *jednokrokovou*, jestliže pro výpočet \mathbf{y}_{n+1} je zapotřebí pouze předcházející hodnota \mathbf{y}_n . Potřebujeme-li pro výpočet \mathbf{y}_{n+1} více než jednu předcházející uzlovou hodnotu, označíme metodu jako *více-krokovou*. Metoda je *explicitní*, jestliže pro výpočet \mathbf{y}_{n+1} máme explicitní vzorec, v opačném případě hovoříme o *implicitní* metodě.

3.1 Eulerovy metody

Eulerova explicitní metoda

Jednou ze základních numerických metod pro výpočet úlohy (2.4) je Eulerova explicitní metoda (EE), kterou můžeme odvodit následujícím způsobem. Pro přibližný výpočet $\mathbf{y}(x)$ provedeme aproximaci $\mathbf{f}(x, \mathbf{y}(x)) \approx \mathbf{f}(x_0, \mathbf{y}(x_0))$, $x \in \langle x_0, x_0 + h_0 \rangle$ s dostatečně malou délkou kroku. Integrací rovnice (2.4) v mezích od x_0 do $x_1 = x_0 + h_0$ získáme

$$\mathbf{y}(x_1) = \mathbf{y}(x_0) + \int_{x_0}^{x_1} \mathbf{f}(\tau, \mathbf{y}(\tau)) d\tau \approx \mathbf{y}_0 + (x_1 - x_0) \mathbf{f}(x_0, \mathbf{y}(x_0)).$$

Jako přibližné řešení \mathbf{y}_1 dostáváme

$$\mathbf{y}_1 = \mathbf{y}_0 + h_0 \mathbf{f}(x_0, \mathbf{y}_0).$$

Proces můžeme opakovat. Obecný vztah pro Eulerovu explicitní metodu má tvar

$$\mathbf{y}_{n+1} = \mathbf{y}_n + h_n \mathbf{f}(x_n, \mathbf{y}_n), \quad n = 0, 1, \dots$$

Nyní si ukážeme některé vlastnosti Eulerovy explicitní metody. Bude nás zajímat, za jaké podmínky je metoda konvergentní, tzn. numerická metoda dostatečně aproximuje přesné řešení, jak rychle metoda konverguje a jaké chyby se při řešení dopouštíme. K tomu využijeme následujícího výrazu známého jako Landauova notace:

Označení. Nechť funkce $f(x)$ je definována na intervalu $\langle a, b \rangle$. Řekneme, že $f(x) \in \mathcal{O}(h^p)$, jestliže existuje $C > 0$ takové, že pro všechna $x \in \langle a, b \rangle$ platí $|f(x)| \leq Ch^p$.

3.1. EULEROVY METODY

Symbolem $\mathbf{l}e_n$ označme *lokální chybu*. Jedná se o chybu, která vzniká v jednom kroku. Eulerovu metodu můžeme zapsat ve tvaru $\mathbf{y}_{n+1} - [\mathbf{y}_n + h\mathbf{f}(x_n, \mathbf{y}_n)] = \mathbf{0}$. Nahrazením \mathbf{y}_k přesným řešením $\mathbf{y}(x_k)$ a rozvinutím do Taylorova polynomu prvního stupně získáme

$$\begin{aligned} \mathbf{l}e_n &= \mathbf{y}(x_{n+1}) - [\mathbf{y}(x_n) + h\mathbf{f}(x_n, \mathbf{y}(x_n))] \\ &= [\mathbf{y}(x_n) + h\mathbf{y}'(x_n) + \mathcal{O}(h^2)] - [\mathbf{y}(x_n) + h\mathbf{y}'(x_n)] = \mathcal{O}(h^2) \end{aligned}$$

Řekneme, že libovolná metoda

$$\mathbf{y}_{n+1} = \mathcal{Y}(\mathbf{f}, h, \mathbf{y}_0, \mathbf{y}_1, \dots, \mathbf{y}_n), \quad n = 0, 1, \dots,$$

kde \mathcal{Y} je formule konkrétní numerické metody, je řádu p , když

$$\mathbf{y}(x_{n+1}) - \mathcal{Y}(\mathbf{f}, h, \mathbf{y}_0, \mathbf{y}_1, \dots, \mathbf{y}_n) = \mathcal{O}(h^{p+1}) \quad (3.1)$$

platí pro každou analytickou funkci \mathbf{f} . Podobně i pro implicitní metody. Jinými slovy můžeme říct, že metoda je řádu p , když dává přesné řešení pro každou polynomiální funkci stupně nejvýše p . Řád metody nám sděluje informaci o jeho lokálním chování, mnohem podstatnější budou pro nás ale informace o globálním chování metody. Označme $\mathbf{e}_n = \mathbf{y}(x_n) - \mathbf{y}_n$ *globální chybu*, která vzniká hromaděním lokálních chyb. Globální chyba Eulerovy explicitní metody je $\mathcal{O}(h)$ a metoda je tedy prvního řádu.

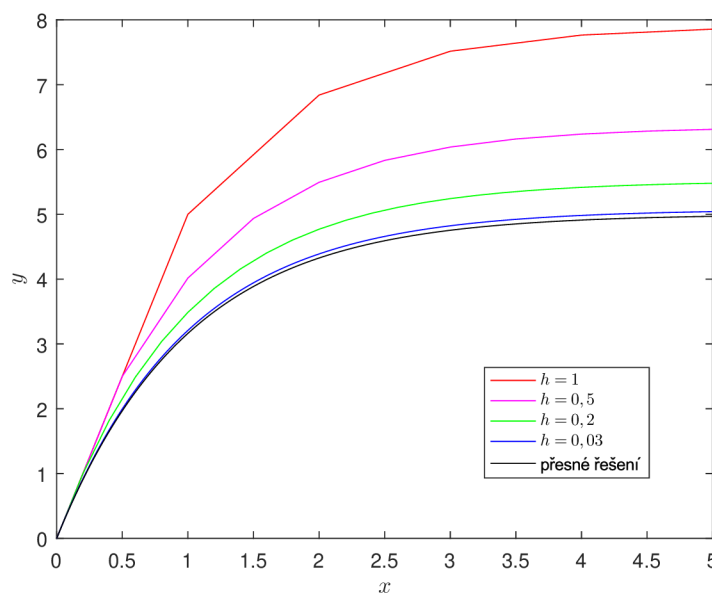
Definice 3.1 (Konvergence metody). Metodu prohlásíme za konvergentní, jestliže pro každou počáteční úlohu (2.4) s lischitzovskou funkcí a rovnoměrným dělením platí

$$\lim_{h \rightarrow 0^+} \max_{n=0,1,\dots,q} \|\mathbf{e}_n\| = 0.$$

Věta 3.2. *Eulerova explicitní metoda je konvergentní.*

Důkaz. Důkaz můžeme nalézt v [4, Theorem 1.1, s. 6]. □

Pro ilustraci aspoň uvedeme obrázek 3.1 řešení rovnice $y' = 5e^{-x}, y(0) = 0$ s různou délkou kroku.



Obrázek 3.1: Řešení Eulerovou explicitní metodou při použití různých délek kroků.

Eulerova implicitní metoda

Použijeme-li aproximaci $\mathbf{f}(x, \mathbf{y}(x)) \approx \mathbf{f}(x_1, \mathbf{y}(x_1))$ na intervalu $x \in \langle x_0, x_1 \rangle$, pak po integraci (2.4) a úpravě získáme vztah

$$\mathbf{y}_1 = \mathbf{y}_0 + h_0 \mathbf{f}(x_1, \mathbf{y}_1), \quad (3.2)$$

který zobecníme

$$\mathbf{y}_{n+1} = \mathbf{y}_n + h_n \mathbf{f}(x_{n+1}, \mathbf{y}_{n+1}), \quad n = 0, 1, \dots$$

a dostaneme obecný tvar Eulerovy implicitní metody (IE). Eulerova implicitní metoda je konvergentní, řádu 1 a je prakticky stejně přesná jako explicitní metoda. Nevýhodou při použití implicitní metody je nutnost řešit soustavu nelineárních rovnic, ale naopak velkou výhodou je její stabilita, kterou podrobně rozebereme v Kapitole 4. Zmíněnou soustavu nelineárních rovnic je možné řešit Newtonovou metodou s počáteční aproximací \mathbf{y}_n .

Lichoběžníková metoda

Eulerovy metody aproximují $\mathbf{f}(x, \mathbf{y}(x))$ na intervalu $\langle x_n, x_{n+1} \rangle$ konstantou, kterou určíme u explicitní metody z levého a implicitní metody z pravého krajního bodu intervalu. Pokud zvolíme jako aproximaci průměr hodnot v krajních bodech, tj.

$$\mathbf{f}(x, \mathbf{y}(x)) \approx \frac{1}{2} [\mathbf{f}(x_n, \mathbf{y}(x_n)) + \mathbf{f}(x_{n+1}, \mathbf{y}(x_{n+1}))], \quad x \in \langle x_n, x_{n+1} \rangle$$

získáme *lichoběžníkovou metodu*, zkráceně TR (*trapezoid rule*), pro kterou platí

$$\begin{aligned} \mathbf{y}(x_{n+1}) &= \mathbf{y}(x_n) + \int_{x_n}^{x_{n+1}} \mathbf{f}(\tau, \mathbf{y}(\tau)) d\tau \approx \\ &\mathbf{y}(x_n) + \frac{1}{2}(x_{n+1} - x_n) [\mathbf{f}(x_n, \mathbf{y}(x_n)) + \mathbf{f}(x_{n+1}, \mathbf{y}(x_{n+1}))], \end{aligned}$$

neboli

$$\mathbf{y}_{n+1} = \mathbf{y}_n + \frac{1}{2} h_n [\mathbf{f}(x_n, \mathbf{y}_n) + \mathbf{f}(x_{n+1}, \mathbf{y}_{n+1})].$$

Řád metody při rovnoměrném dělení odvodíme ze vztahu (3.1) pomocí Taylorova rozvoje

$$\begin{aligned} &\mathbf{y}(x_{n+1}) - \left\{ \mathbf{y}(x_n) + \frac{1}{2} h [\mathbf{f}(x_n, \mathbf{y}(x_n)) + \mathbf{f}(x_{n+1}, \mathbf{y}(x_{n+1}))] \right\} \\ &= \left\{ \mathbf{y}(x_n) + h \mathbf{y}'(x_n) + \frac{1}{2} h^2 \mathbf{y}''(x_n) + \mathcal{O}(h^3) \right\} \\ &\quad - \left\{ \mathbf{y}(x_n) + \frac{1}{2} h [\mathbf{y}'(x_n) + \mathbf{y}'(x_n) + h \mathbf{y}''(x_n) + \mathcal{O}(h^2)] \right\} = \mathcal{O}(h^3). \end{aligned}$$

Lichoběžníková metoda je již řádu 2, je konvergentní a globální chyba je $\mathcal{O}(h^2)$.

3.2 Rungeovy-Kuttovy metody**Explicitní Rungeovy-Kuttovy metody**

Explicitní Rungeovy-Kuttovy metody (ERK) vychází z integrace rovnice (2.4) v mezích x_n a $x_{n+1} = x_n + h$

$$\mathbf{y}_{n+1} = \mathbf{y}_n + \int_{x_n}^{x_{n+1}} \mathbf{f}(\tau, \mathbf{y}(\tau)) d\tau = \mathbf{y}_n + h \int_0^1 \mathbf{f}(x_n + h\tau, \mathbf{y}(x_n + h\tau)) d\tau.$$

3.2. RUNGEOVY-KUTTOVY METODY

Integrál můžeme nahradit sumou a získáme obecný tvar ERK stupně s

$$\mathbf{y}_{n+1} = \mathbf{y}_n + h \sum_{j=1}^s b_j \mathbf{f}(x_n + c_j h, \mathbf{y}(x_n + c_j h)), \quad n = 0, 1, \dots$$

Koeficienty b_1, b_2, \dots, b_s a c_1, c_2, \dots, c_s , které jsou nezávislé na funkci \mathbf{f} specifikují konkrétní metodu. Jelikož ale neznáme hodnoty \mathbf{y} v uzlech $x_n + c_1 h, x_n + c_2 h, \dots, x_n + c_s h$, musíme zavést aproximace. Označme aproximace $\mathbf{y}(x_n + c_j h)$ jako $\boldsymbol{\xi}_j$, $j = 1, 2, \dots, s$. Položíme-li $c_1 = 0$, získáme $\boldsymbol{\xi}_1 = \mathbf{y}(x_n) = \mathbf{y}_n$. Hlavní myšlenka ERK metody je, že každé $\boldsymbol{\xi}_j$, $j = 2, 3, \dots, s$, se dá vyjádřit jako lineární kombinace $\mathbf{f}(x_n, \boldsymbol{\xi}_1), \mathbf{f}(x_n + c_2 h, \boldsymbol{\xi}_2), \dots, \mathbf{f}(x_n + c_{j-1} h, \boldsymbol{\xi}_{j-1})$. Aproximace $\boldsymbol{\xi}_j$ můžeme určit předpisem

$$\begin{aligned} \boldsymbol{\xi}_1 &= \mathbf{y}_n \\ \boldsymbol{\xi}_2 &= \mathbf{y}_n + h a_{2,1} \mathbf{f}(x_n, \boldsymbol{\xi}_1) \\ \boldsymbol{\xi}_3 &= \mathbf{y}_n + h a_{3,1} \mathbf{f}(x_n, \boldsymbol{\xi}_1) + h a_{3,2} \mathbf{f}(x_n + c_2 h, \boldsymbol{\xi}_2) \\ &\vdots \\ \boldsymbol{\xi}_s &= \mathbf{y}_n + h \sum_{j=1}^{s-1} a_{s,j} \mathbf{f}(x_n + c_j h, \boldsymbol{\xi}_j) \\ \mathbf{y}_{n+1} &= \mathbf{y}_n + h \sum_{i=1}^s b_i \mathbf{f}(x_n + c_i h, \boldsymbol{\xi}_i), \end{aligned} \tag{3.3}$$

Matice $A = (a_{i,j})_{i,j=1,2,\dots,s}$, se nazývá *RK matice*, vektory

$$\mathbf{b} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_s \end{pmatrix}, \quad \mathbf{c} = \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_s \end{pmatrix}$$

jsou *RK váhy* a *RK uzly*. Koeficienty dané RK metody se zapisují do *RK tabulky* známé jako *Butcherova tabulka*, která je ve tvaru

$$\frac{\mathbf{c} \mid A}{\mid \mathbf{b}^T}.$$

Nabízí se otázka, jak vybrat vhodnou RK matici s rozumnými váhami a uzly. Odpověď na tuto otázku nám může poskytnout rozložení rovnic do Taylorových řad a následné porovnání. Toto odvození je technicky náročné, lze jej najít například v [8]. Z Taylorova rozvoje pro řád 4 plynou podmínky

$$\begin{aligned}
c_i &= a_{i,1} + a_{i,2} + \cdots + a_{i,i-1}, & i &= 2, 3, 4, \\
b_1 + b_2 + b_3 + b_4 &= 1, \\
b_2c_2 + b_3c_3 + b_4c_4 &= \frac{1}{2}, \\
b_2c_2^2 + b_3c_3^2 + b_4c_4^2 &= \frac{1}{3}, \\
b_3c_2a_{3,2} + b_4(c_2a_{4,2} + c_3a_{4,3}) &= \frac{1}{6}, \\
b_2c_2^3 + b_3c_3^3 + b_4c_4^3 &= \frac{1}{4}, \\
b_3c_2^2a_{3,2} + b_4(c_2^2a_{4,2} + c_3^2a_{4,3}) &= \frac{1}{12}, \\
b_3c_2c_3a_{3,2} + b_4(c_2a_{4,2} + c_3a_{4,3})c_4 &= \frac{1}{8}, \\
b_4c_2a_{3,2}a_{4,3} &= \frac{1}{24}.
\end{aligned} \tag{3.4}$$

Má-li funkce $f(x, \mathbf{y})$ spojitě všechny parciální derivace do řádu p včetně, pak se dá dokázat, že pro $s = 1, 2, 3, 4$ platí, že řád s -stupňové RK metody je nejvýše s . Jinými slovy můžeme říci, že řád může být maximálně srovnatelný se stupněm RK metody. Pro $s \geq 5$ již není možné dosáhnout řádu $p = s$. Viz [1] a [4].

Metody prvního řádu

Existuje pouze jedna explicitní RK metoda řádu jedna, a to je explicitní Eulerova metoda.

Metody druhého řádu

Pro případ $p = s = 2$ nám 2. řád metody zajistí podmínky

$$b_1 + b_2 = 1, \quad b_2c_2 = \frac{1}{2}, \quad a_{2,1} = c_2.$$

Prostřednictvím parametru θ lze splnění těchto podmínek zachytit volbou jednotlivých koeficientů v tabulce

$$\begin{array}{c|cc}
0 & & \\
\theta & \theta & \\
\hline
& 1 - \frac{1}{2\theta} & \frac{1}{2\theta}
\end{array} .$$

Zvolíme-li parametr $\theta = \frac{1}{2}$, získáme *modifikovanou Eulerovu metodu*, volbou $\theta = 1$ obdržíme *druhou modifikaci Eulerovy metody* a pro $\theta = \frac{2}{3}$ dostaneme *Ralstonovu metodu řádu 2*. [2]

Metody třetího řádu

Podmínky pro rovnice třetího řádu jsou

$$\begin{aligned}
b_1 + b_2 + b_3 &= 1, & b_2c_2 + b_3c_3 &= \frac{1}{2}, & b_2c_2^2 + b_3c_3^2 &= \frac{1}{3}, \\
b_3c_2a_{3,2} &= \frac{1}{6}, & c_2 &= a_{2,1}, & c_3 &= a_{3,1} + a_{3,2},
\end{aligned}$$

což je soustava 6 rovnic o 8 neznámých, tedy soustava se dvěma parametry volnosti. Volbou $c_2 = \frac{1}{2}$ a $c_3 = \frac{3}{4}$ získáme metodu známou jako *Ralstonova metoda řádu 3* (RK3), jejíž Butcherova tabulka vypadá následovně.

3.2. RUNGEOVY-KUTTOVY METODY

$$\begin{array}{c|cc} 0 & & \\ \frac{1}{2} & \frac{1}{2} & \\ \frac{3}{4} & 0 & \frac{3}{4} \\ \hline & \frac{2}{9} & \frac{1}{3} & \frac{4}{9} \end{array}$$

Metody čtvrtého řádu

Pro metody čtvrtého řádu už platí podmínky (3.4). Opět máme soustavu závislou na dvou parametrech. Nejvíce využívaná RK metoda čtvrtého řádu je *klasická Rungeova-Kuttova metoda* (RK4), kterou získáme volbou $c_2 = c_3 = \frac{1}{2}$.

$$\begin{array}{c|ccc} 0 & & & \\ \frac{1}{2} & \frac{1}{2} & & \\ \frac{1}{2} & 0 & \frac{1}{2} & \\ 1 & 0 & 0 & 1 \\ \hline & \frac{1}{6} & \frac{1}{3} & \frac{1}{3} & \frac{1}{6} \end{array}$$

Za povšimnutí stojí nuly nacházející se v RK matici, které zefektivní proces výpočtu.

Implicitní Rungeovy-Kuttovy metody

Hlavní myšlenka *implicitní Rungeovy-Kuttovy metody* (IRK) je umožnit vektorovým funkcím $\xi_1, \xi_2, \dots, \xi_s$ záviset na sobě obecněji než v případě (3.3). IRK můžeme popsat rovnicemi

$$\begin{aligned} \xi_i &= \mathbf{y}_n + h \sum_{j=1}^s a_{i,j} \mathbf{f}(x_n + c_j h, \xi_j), \quad i = 1, 2, \dots, s, \\ \mathbf{y}_{n+1} &= \mathbf{y}_n + h \sum_{i=1}^s b_i \mathbf{f}(x_n + c_i h, \xi_i). \end{aligned} \quad (3.5)$$

V tomto případě je $A = (a_{i,j})_{i,j=1,2,\dots,s}$ libovolná matice na rozdíl od ERK, kde byla pouze ryze dolní trojúhelníková (nuly na diagonále). Terminologie z ERK (RK váhy, RK uzly, stupeň metody, atd.) zůstává stejná. Hlavní výhodou implicitní metody je její stabilita, další potenciální výhodou IRK oproti ERK je, že řád p může být vyšší než stupeň s . Přírozenou nevýhodou implicitní metody zůstává nutnost řešit soustavu nelineárních rovnic. Jako ukázkou IRK uvedeme dvě metody druhého stupně, jejichž Butcherova tabulka je tvaru

$$\begin{array}{c|cc} 0 & 0 & 0 \\ 1 & \frac{1}{2} & \frac{1}{2} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array} \quad \text{a} \quad \begin{array}{c|cc} 0 & \frac{1}{4} & -\frac{1}{4} \\ \frac{2}{3} & \frac{1}{4} & \frac{5}{12} \\ \hline & \frac{1}{4} & \frac{3}{4} \end{array}.$$

První uvedená tabulka reprezentuje již známou lichoběžníkovou metodou řádu 2, druhá tabulka charakterizuje metodu nesoucí jméno *Radauova metoda* (RM2), která je řádu 3.

3.3 Lineární vícezkrokové metody

Metody popsané v minulé kapitole patřily mezi jednokrokové numerické metody. K výpočtu přibližného řešení v x_{n+1} využívaly pouze aproximací \mathbf{y}_n . U vícezkrokových metod se hodnota \mathbf{y}_{n+1} určí pomocí předchozích hodnot $\mathbf{y}_n, \mathbf{y}_{n-1}, \dots, \mathbf{y}_{n-(s-1)}$ a odpovídajících hodnot $\mathbf{f}(x_n, \mathbf{y}_n), \mathbf{f}(x_{n-1}, \mathbf{y}_{n-1}), \dots$. Obecný tvar lineární s-krokové metody má tvar

$$a_0 \mathbf{y}_n + a_1 \mathbf{y}_{n+1} + \dots + a_s \mathbf{y}_{n+s} = h[b_0 \mathbf{f}(x_n, \mathbf{y}_n) + b_1 \mathbf{f}(x_{n+1}, \mathbf{y}_{n+1}) + \dots + b_s \mathbf{f}(x_{n+s}, \mathbf{y}_{n+s})],$$

stručně

$$\sum_{m=0}^s a_m \mathbf{y}_{n+m} = h \sum_{m=0}^s b_m \mathbf{f}(x_{n+m}, \mathbf{y}_{n+m}), \quad n = 0, 1, \dots, \quad (3.6)$$

kde $a_m, b_m, m = 0, 1, \dots, s$, jsou konstanty nezávislé na h, n a \mathbf{f} . Je-li $b_s = 0$, vícezkroková metoda je *explicitní*, v opačném případě je metoda *implicitní*. Dále budeme předpokládat, že rovnice je v *normovaném tvaru*, tj. $a_s = 1$. Chceme-li použít s-krokovou metodu, potřebujeme znát s startovacích hodnot, které ale můžeme určit z počáteční podmínky užitím jiné ménězkrokové metody. Při určování metody hraje důležitou roli výběr koeficientů a_m a b_m , které vybíráme tak, abychom získali metodu rozumného řádu. V souladu s definicí řádu metody v předchozí kapitole můžeme vyslovit následující lemma.

Lemma 3.3. *Lineární vícezkroková metoda (3.6) je řádu $p \geq 1$ právě tehdy, když*

$$\sum_{m=0}^s a_m \mathbf{y}(x + mh) - h \sum_{m=0}^s b_m \mathbf{y}'(x + mh) = \mathcal{O}(h^{p+1}), \quad h \rightarrow 0$$

pro všechny dostatečně hladké funkce \mathbf{y} .

Lineární vícezkrokovou metodu (3.6) můžeme charakterizovat pomocí dvou polynomů

$$\rho(\omega) := \sum_{m=0}^s a_m \omega^m \quad \text{a} \quad \sigma(\omega) := \sum_{m=0}^s b_m \omega^m.$$

Věta 3.4. *Lineární vícezkroková metoda (3.6) je řádu $p \geq 1$ právě tehdy, když existuje $c \neq 0$ takové, že,*

$$\rho(\omega) - \sigma(\omega) \ln(\omega) = c(\omega - 1)^{p+1} + \mathcal{O}(|\omega - 1|^{p+2}).$$

Důkaz věty můžeme najít v [4, Theorem 2.1, s. 22]. Věta 3.4 je velmi užitečná pro určování řádu metody, neboť nejsou potřeba „speciální triky“ a prostým dosazením do vztahu určíme řád metody.

Definice 3.5. (Kořenová podmínka) Mějme polynom $p(x) = \sum_{i=0}^n a_i x^i$. Řekneme, že polynom p splňuje *kořenovou podmínku*, jestliže všechny jeho kořeny leží v komplexní rovině v jednotkovém kruhu se středem v počátku a každý kořen ležící na hranici je jednoduchý.

Věta 3.6. *Předpokládejme, že chyba startovacích hodnot $\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_{s-1}$ je nulová a $h \rightarrow 0^+$. Lineární vícezkroková metoda (3.6) je konvergentní tehdy a jen tehdy, když je řádu $p \geq 1$ a polynom $\rho(\omega)$ splňuje kořenovou podmínku.*

Důkaz věty nalezneme v [4, Theorem 2.2, s. 24]. Výše zmíněná věta se někdy nazývá *Dahlquistova věta o ekvivalenci* a metody, které vyhovují kořenové podmínce, se označují jako *D-stabilní*. Metoda (3.6) závisí na $2s + 1$ parametrech. Maximální řád konvergentní s-krokové metody je nejvýše $2\lfloor (s + 2)/2 \rfloor$ pro implicitní a s pro explicitní metodu. Tato podmínka je známa jako *první Dahlquistova bariéra*.

3.3. LINEÁRNÍ VÍCEKROKOVÉ METODY

3.3.1 Adams-Bashforthovy metody

Adams-Bashforthova metoda (ABs) je lineární s-kroková explicitní metoda, která má obecný tvar

$$\mathbf{y}_{n+s} = \mathbf{y}_{n+s-1} + h \sum_{m=0}^{s-1} b_m \mathbf{f}(x_{n+m}, \mathbf{y}_{n+m}), \quad (3.7)$$

kde koeficienty (váhy) b_m jsou nezávislé na n a délce kroku h . Váhy b_m jsou určeny vztahy [10]

$$\begin{aligned} b_{s-1} + b_{s-2} + b_{s-3} + \cdots + b_1 + b_0 &= 1 \\ b_{s-2} + 2b_{s-3} + \cdots + (s-2)b_1 + (s-1)b_0 &= -\frac{1}{2} \\ b_{s-2} + 2^2b_{s-3} + \cdots + (s-2)^2b_1 + (s-1)^2b_0 &= \frac{1}{3} \\ &\vdots \\ &\vdots \\ b_{s-2} + 2^{(s-1)}b_{s-3} + \cdots + (s-2)^{(s-1)}b_1 + (s-1)^{(s-1)}b_0 &= (-1)^{(s-1)}\frac{1}{s}. \end{aligned}$$

Pro $s = 1$ obdržíme Eulerovu explicitní metodu, pro $s = 2$ získáme

$$\mathbf{y}_{n+2} = \mathbf{y}_{n+1} + h \left[\frac{3}{2} \mathbf{f}(x_{n+1}, \mathbf{y}_{n+1}) - \frac{1}{2} \mathbf{f}(x_n, \mathbf{y}_n) \right], \quad (3.8)$$

pro $s = 3$ dostaneme

$$\mathbf{y}_{n+3} = \mathbf{y}_{n+2} + h \left[\frac{23}{12} \mathbf{f}(x_{n+2}, \mathbf{y}_{n+2}) - \frac{4}{3} \mathbf{f}(x_{n+1}, \mathbf{y}_{n+1}) + \frac{5}{12} \mathbf{f}(x_n, \mathbf{y}_n) \right].$$

Snadno ověříme, že Adams-Bashforthovy metody jsou D-stabilní. Pro libovolnou s-krokovou Adams-Bashforthovu metodu má polynom $\rho(\omega) = \omega^s - \omega^{s-1} = \omega^{s-1}(\omega - 1)$ kořeny 0 a 1, přičemž druhý kořen je jednoduchý, tudíž jsou metody D-stabilní. Řád dvou výše zmíněných metod můžeme určit podle věty 3.4. Polynomy charakterizující 2-krokovou metodu mají tvar

$$\rho(\omega) = \omega^2 - \omega, \quad \sigma(\omega) = \frac{3}{2}\omega - \frac{1}{2}.$$

Po dosazení substituce $\xi := \omega - 1$ a zapsání logaritmu pomocí řady, dostaneme výraz

$$\begin{aligned} \rho(\omega) - \sigma(\omega) \ln(\omega) &= (\omega^2 - \omega) - \left(\frac{3}{2}\omega - \frac{1}{2} \right) \ln(\omega) = \\ &= ((\xi + 1)^2 - (\xi + 1)) - \left(\frac{3}{2}(\xi + 1) - \frac{1}{2} \right) \ln(\xi + 1) = \\ &= (\xi + \xi^2) - \left(1 + \frac{3}{2}\xi \right) \left(1 - \frac{1}{2}\xi + \frac{1}{3}\xi^2 + \dots \right) = \frac{5}{12}\xi^3 + \mathcal{O}(\xi^4). \end{aligned}$$

Metoda je tedy řádu 2, což je v podstatě nejvyšší možný řád explicitní 2-krokové metody. Analogicky můžeme zjistit, že výše zmíněná 3-kroková metoda je třetího řádu.

Na závěr nutno zmínit, že k Adams-Bashforthovým metodám existují i implicitní protějšky nazývané Adams-Moultonovy metody (AM).

3.3.2 Metody zpětného derivování

Dalším typem lineárních vícekových metod jsou metody zpětného derivování. Charakteristické polynomy pro s -krokovou metodu mají tvar

$$\beta = \left(\sum_{m=1}^s \frac{1}{m} \right)^{-1}, \quad \sigma(\omega) = \beta \omega^s, \quad \rho(\omega) = \beta \sum_{m=1}^s \frac{1}{m} \omega^{s-m} (\omega - 1)^m. \quad (3.9)$$

Každá s -kroková metoda zpětného derivování (BDFs) je implicitní metoda řádu s . Nejjednodušší BDF je již známá Eulerova implicitní metoda. Další metody BDF jsou např.

$$s = 2 : \quad \mathbf{y}_{n+2} - \frac{4}{3}\mathbf{y}_{n+1} + \frac{1}{3}\mathbf{y}_n = \frac{2}{3}\mathbf{f}(x_{n+2}, \mathbf{y}_{n+2}),$$

$$s = 3 : \quad \mathbf{y}_{n+3} - \frac{18}{11}\mathbf{y}_{n+2} + \frac{9}{11}\mathbf{y}_{n+1} - \frac{2}{11}\mathbf{y}_n = \frac{6}{11}\mathbf{f}(x_{n+3}, \mathbf{y}_{n+3}),$$

Odvození je velmi jednoduché. Dosazením $s = 3$ do (3.9) získáme

$$\beta = \left(1 + \frac{1}{2} + \frac{1}{3} \right)^{-1} = \frac{6}{11}, \quad \sigma(\omega) = \frac{6}{11}\omega^3$$

a

$$\rho(\omega) = \frac{6}{11} \left[\frac{1}{1}\omega^2(\omega - 1) + \frac{1}{2}\omega(\omega - 1)^2 + \frac{1}{3}\omega(\omega - 1) \right] = \omega^3 - \frac{18}{11}\omega^2 + \frac{9}{11}\omega - \frac{2}{11}.$$

Věta 3.7 ([4, Theorem 2.4, s. 28]). *BDFs je konvergentní právě tehdy, když $1 \leq s \leq 6$.*

Metody zpětného derivování budou hrát významnou roli v řešení tuhých systémů, kterým je věnována Kapitola 6.

4 Stabilita

V této kapitole se budeme zabývat stabilitou řešení diferenciálních rovnic. Řešení budeme považovat za stabilní, jestliže po provedení malé změny (poruchy) v zadání získáme řešení, které se od původního řešení liší jen velmi málo. Omezíme se pouze na poruchy v počáteční podmínce \mathbf{y}_0 , funkci $\mathbf{f}(x, \mathbf{y})$ a x_0 budeme považovat za pevně určené. Informace jsou čerpány ze zdrojů [1], [4] a [7].

Definice 4.1. O řešení $\mathbf{y}_\xi(x)$ počáteční úlohy (2.4) s počáteční podmínkou $\mathbf{y}_0 = \boldsymbol{\xi}$ řekneme, že je *stabilní*, jestliže pro každé $\varepsilon > 0$ existuje $\delta > 0$ takové, že pro každé řešení $\mathbf{y}_\eta(x)$ úlohy (2.4) s počáteční podmínkou $\mathbf{y}_0 = \boldsymbol{\eta}$ splňující $\|\boldsymbol{\xi} - \boldsymbol{\eta}\| < \delta$ platí $\|\mathbf{y}_\xi(x) - \mathbf{y}_\eta(x)\| < \varepsilon$ pro $\forall x \in \langle x_0, \infty \rangle$. Pokud navíc $\|\mathbf{y}_\xi(x) - \mathbf{y}_\eta(x)\| \rightarrow 0$ pro $x \rightarrow \infty$, je řešení $\mathbf{y}_\xi(x)$ *asymptoticky stabilní*. V případě, že řešení nesplňuje podmínky stability, prohlásíme jej za *nestabilní*.

Věta 4.2. Mějme úlohu $\mathbf{y}' = \mathbf{A}\mathbf{y}$ s počáteční podmínkou $\mathbf{y}(x_0) = \boldsymbol{\eta}$, kde \mathbf{A} je čtvercová číselná matice. Úloha je *asymptoticky stabilní*, jestliže všechna vlastní čísla matice \mathbf{A} mají zápornou reálnou část, tj. $\operatorname{Re}(\lambda_k) < 0, k = 1, 2, \dots, d$.

V další části budeme uvažovat, že daná numerická metoda s konstantní délkou kroku $h > 0$ je aplikovaná na skalární diferenciální rovnici (Testovací úloha)

$$y' = \lambda y, \quad y(0) = 1, \quad (4.1)$$

kde $\lambda \in \mathbb{C}$. Analytické řešení rovnice (4.1) je $y(x) = e^{\lambda t}$, pro které platí, že $\lim_{t \rightarrow \infty} y(x) = 0$ právě tehdy, když $\operatorname{Re}(\lambda) < 0$. Řekneme, že *oblast absolutní stability* \mathcal{D} jsou taková čísla $h\lambda \in \mathbb{C}$, pro které je splněna *podmínka stability* $\lim_{n \rightarrow \infty} y_n = 0$. Jinými slovy, \mathcal{D} je množina všech $h\lambda$, pro které je Testovací úloha (4.1) asymptoticky stabilní.

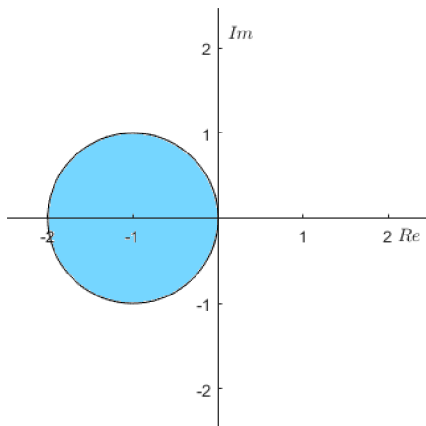
Nyní můžeme prozkoumat stabilitu jednotlivých numerických metod, počínaje Eulerovou explicitní metodou. Aplikací Eulerovy explicitní metody na Testovací úlohu (4.1) obdržíme

$$y_{n+1} = y_n + h\lambda y_n = (1 + h\lambda)y_n = (1 + h\lambda)^2 y_{n-1} = \dots = (1 + h\lambda)^{n+1} y_0, \quad n = 0, 1, \dots$$

Posloupnost $\{y_n\}_{n=0,1,\dots}$ je geometrická řada, která konverguje k nule, tj. $\lim_{n \rightarrow \infty} y_n = 0$ právě tehdy, když $|1 + h\lambda| < 1$. Z toho můžeme usoudit, že oblast absolutní stability

$$\mathcal{D}_{EE} = \{z \in \mathbb{C} : |1 + z| < 1\}$$

je vnitřek jednotkového kruhu se středem v $z = -1$.



Obrázek 4.1: Oblast absolutní stability Eulerovy explicitní metody.

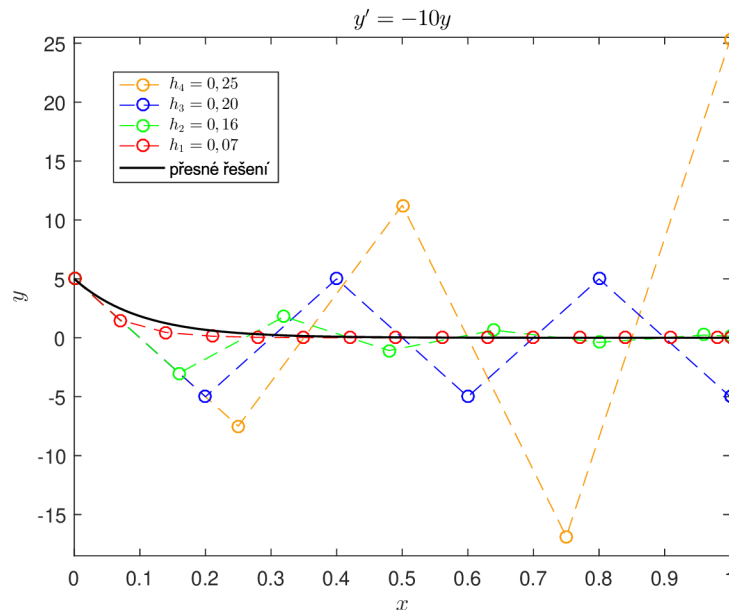
Co by se stalo, kdybychom zvolili délku kroku takovou, že $h\lambda \notin \mathcal{D}$? Na tuto otázku si odpovíme následujícím příkladem.

Příklad č. 1

Uvažujme počáteční úlohu

$$y' = -10y, \quad y(0) = 1,$$

kteřou budeme řešit Eulerovou explicitní metodou s konstantní délkou kroku $h_1 = 0,07$, $h_2 = 0,16$, $h_3 = 0,20$ a $h_4 = 0,25$. Aby byla splněna podmínka stability $\lim_{n \rightarrow \infty} y_n = 0$ musí platit $|1 + h\lambda| < 1$. V našem případě $\lambda = -10$, takže délka kroku h musí splňovat nerovnost $h < 0,2$. Na obrázku 4.2 jsou zakresleny výsledky pro jednotlivé délky kroků.



Obrázek 4.2: Řešení Eulerovou explicitní metodou při použití různých délek kroků.

Délka kroku h_1 (resp. h_2) je ostře menší než $0,2$. Numerické řešení se přibližuje k přesným hodnotám a globální chyba konverguje k nule. Zvolíme-li délku kroku na hranici stability, tj. $h_3 = 0,20$, získáme řešení oscilující s konstantní vzdáleností od $y = 0$. Poslední zvolená délka kroku $h_4 = 0,25$ je již větší než povolená hodnota $0,2$. Získané hodnoty se od přesného řešení vzdalují a globální chyba narůstá. Z tohoto důvodu je vždy nutné zvolit délku kroku takovou, aby $h\lambda \in \mathcal{D}$.

Dříve než se přesuneme k další metodě, prozkoumáme, jak by se situace změnila, kdyby Testovací úloha byla vektorová rovnice

$$\mathbf{y}' = \Lambda \mathbf{y}, \quad \mathbf{y}(0) = \mathbf{y}_0,$$

kde Λ je libovolná matice $d \times d$. Předpokládejme, že matice Λ má různá vlastní čísla, takže ji můžeme *diagonalizovat* na tvar $\Lambda = VDV^{-1}$, kde V je regulární matice vlastních vektorů a $D = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_d)$ skládající se z vlastních čísel Λ . Dá se ukázat, že existují vektory $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_d \in \mathbb{C}^d$ závisející pouze na \mathbf{y}_0 takové, že

$$\mathbf{y}_n = \sum_{k=1}^d (1 + h\lambda_k)^n \mathbf{x}_k, \quad n = 0, 1, \dots$$

Předpokládejme, že analytické řešení je asymptoticky stabilní. To se stane tehdy a jen tehdy, když $Re(\lambda_k) < 0$ pro všechna $k = 1, 2, \dots, d$. Abychom napodobili asymptotické chování pomocí Eulerovy explicitní metody, musí platit $|1 + h\lambda_k| < 1, k = 1, 2, \dots, d$, neboli všechny součiny $h\lambda_k$ musí ležet v \mathcal{D}_{EE} . To znamená, že pro zachování stability je někdy potřeba redukovat délku kroku.

V případě, že máme obecný ODR systém

$$\mathbf{y}' = \mathbf{f}(x, \mathbf{y}), \quad \mathbf{y}(x_0) = \mathbf{y}_0,$$

kde funkce \mathbf{f} je diferencovatelná vzhledem k \mathbf{y} , klademe požadavek, aby v n -tém kroku vlastní čísla *Jacobiho matice* $J_n = \partial \mathbf{f}(x_n, \mathbf{y}_n) / \partial \mathbf{y}$ spolu s délkou kroku splňovala

$$h\lambda_{n,1}, h\lambda_{n,2}, \dots, h\lambda_{n,d} \in \mathcal{D}_{EE}.$$

Vraťme se ke skalární testovací úloze (4.1) a prozkoumejte stabilitu Eulerovy implicitní metody. Rovnici

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

upravíme na tvar

$$y_{n+1} = \frac{1}{1 - h\lambda} y_n = \dots = \left(\frac{1}{1 - h\lambda} \right)^n y_0, \quad n = 0, 1, \dots$$

Získáme opět geometrickou řadu, která bude konvergentní právě tehdy, když $|1 - h\lambda| > 1$. Oblast absolutní stability Eulerovy implicitní metody

$$\mathcal{D}_{IE} = \{z \in \mathbb{C} : |1 - z| > 1\}$$

je vnějšek jednotkového kruhu se středem v bodě $z = 1$. Viz obrázek 4.3.

Pokračujme v našem vyšetřování numerických metod a prozkoumejte stabilitu lichoběžníkové metody aplikovanou na testovací úlohu (4.1).

$$y_{n+1} = y_n + \frac{1}{2}h(y_n + y_{n+1}),$$

odtud

$$y_{n+1} = \left(\frac{1 + \frac{1}{2}h\lambda}{1 - \frac{1}{2}h\lambda} \right) y_0 = \dots = \left(\frac{1 + \frac{1}{2}h\lambda}{1 - \frac{1}{2}h\lambda} \right)^n y_0, \quad n = 0, 1, \dots$$

I v tomto případě jsme získali geometrickou řadu, pro níž je podmínka konvergence $\left| \frac{1 + \frac{1}{2}h\lambda}{1 - \frac{1}{2}h\lambda} \right| < 1$. Snadno ověříme, že podmínka stability je splněna pro všechna z taková, jejíž reálná část je záporná, tj. $Re(z) < 0$.

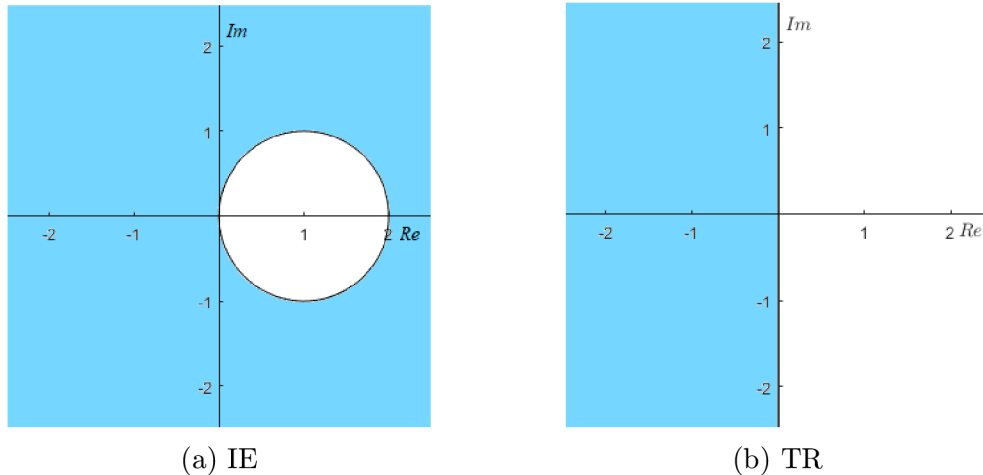
$$\mathcal{D}_{TR} = \{z \in \mathbb{C} : Re(z) < 0\}$$

Oblast absolutní stability Lichoběžníkové metody je celá záporná komplexní polorovina. Tato vlastnost je natolik podstatná, že si zaslouží pojmenování.

Definice 4.3. Řekneme, že metoda je *A-stabilní*, jestliže

$$\mathbb{C}^- := \{z \in \mathbb{C} : Re(z) < 0\} \subseteq \mathcal{D}.$$

Lichoběžníková metoda a Eulerova implicitní metoda jsou A-stabilní, zatímco Eulerova explicitní není. Pokud je metoda A-stabilní, délka kroku není závislá na podmínce stability a řídí se pouze požadovanou přesností.



Obrázek 4.3: Oblast absolutní stability Eulerovy implicitní metody (IE) a lichoběžníkové metody (TR).

4.1 Stabilita Rungeových-Kuttových metod

V následující kapitole se budeme zabývat stabilitou Rungeových-Kuttových metod. Aplikací implicitní Rungeovy-Kuttovy metody (3.5) na skalární Testovací úlohu (4.1) získáme

$$\xi_i = y_n + h \sum_{j=1}^s a_{i,j} f(x_n + c_j h, \xi_j) = y_n + h\lambda \sum_{j=1}^s a_{i,j} \xi_j \quad i = 1, 2, \dots, s. \quad (4.2)$$

Označme

$$\boldsymbol{\xi} = \begin{pmatrix} \xi_1 \\ \xi_2 \\ \vdots \\ \xi_s \end{pmatrix}, \quad \mathbf{1} = \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix},$$

pak (4.2) lze zapsat $\boldsymbol{\xi} = \mathbf{1}y_n + h\lambda A\boldsymbol{\xi}$. Řešení tohoto systému lineárních rovnic můžeme vyjádřit

$$\boldsymbol{\xi} = (I - h\lambda A)^{-1} \mathbf{1}y_n$$

za předpokladu, že matice $I - h\lambda A$ je regulární. Aproximace v následujícím uzlu lze pak vyjádřit explicitně

$$\begin{aligned} y_{n+1} &= y_n + h\lambda \sum_{i=1}^s b_i \xi_i = y_n + h\lambda \sum_{i=1}^s b_i (I - h\lambda A)^{-1} \mathbf{1}y_n = \\ &= [1 + h\lambda \mathbf{b}^T (I - h\lambda A)^{-1} \mathbf{1}] y_n, \quad n = 0, 1, \dots \end{aligned}$$

Označme \mathbb{P}_α množinu všech polynomů stupně nejvýše α a $\mathbb{P}_{\alpha/\beta}$ množinu všech racionálních funkcí p/q , kde $p \in \mathbb{P}_\alpha$ a $q \in \mathbb{P}_\beta$.

4.1. STABILITA RUNGEOVÝCH-KUTTOVÝCH METOD

Pro každou implicitní Rungeovu-Kuttovu metodu (3.5) aplikovanou na Testovací úlohu (4.1) existuje funkce stability $r \in \mathbb{P}_{s/s}$ taková, že

$$y_n = [r(h\lambda)]^n y_0, \quad n = 0, 1, \dots,$$

přičemž

$$r(z) := 1 + z\mathbf{b}^T(I - zA)^{-1}\mathbf{1}, \quad z \in \mathbb{C}. \quad (4.3)$$

Inverzní matici $(I - zA)^{-1}$ můžeme vyjádřit známou formulí

$$(I - zA)^{-1} = \frac{\text{adj}(I - zA)}{\det(I - zA)},$$

kde $\text{adj}(C)$ značí *adjungovanou* matici C . Snadno si ověříme, že $z\mathbf{b}^T(I - zA)^{-1}\mathbf{1} \in \mathbb{P}_s$ a $\det(I - zA) \in \mathbb{P}_s$, takže platí $r \in \mathbb{P}_{s/s}$. Nutno ještě připomenout, že pokud je metoda explicitní, matice A je ostře dolní trojúhelníková (hlavní diagonála nulová), matice $I - zA$ má na diagonále jedničky, proto $\det(I - zA) = 1$ a funkce stability je polynom $r \in \mathbb{P}_s$.

Lemma 4.4 ([4, Lemma 4.2, s. 60]). *Použijeme-li libovolnou Rungeovu-Kuttovu metodu na Testovací úlohu (4.1), získáme geometrickou posloupnost $y_n = [r(h\lambda)]^n, n = 0, 1, 2, \dots$, kde r je funkce stability. Pak platí*

$$\mathcal{D}_{RK} = \{z \in \mathbb{C} : |r(z)| < 1\}$$

Důsledek lemmatu 4.4 je, že explicitní Rungeova-Kuttova metoda nemůže být A-stabilní. Na obrázku 4.4 jsou znázorněny oblasti absolutní stability vybraných explicitních Rungeových-Kuttových metod.

Nyní si uvedeme lemma, pomocí kterého budeme moct snadno ověřit, zda implicitní RK metoda je A-stabilní.

Lemma 4.5 ([4, Lemma 4.3, s. 61]). *Nechť r je libovolná nekonstantní racionální funkce. Pak $|r(z)| < 1$ pro všechna $z \in \mathbb{C}^-$ právě tehdy, když všechny póly mají kladnou reálnou část a $|r(it)| \leq 1$ pro všechna $t \in \mathbb{R}$.*

Pomocí lemmatu prozkoumáme stabilitu následujících implicitních metod. Uvažujme metody zmíněné v předchozí kapitole, tj. Radauova metoda a lichoběžníková metoda, jejíž Butcherovy tabulky jsou tvaru

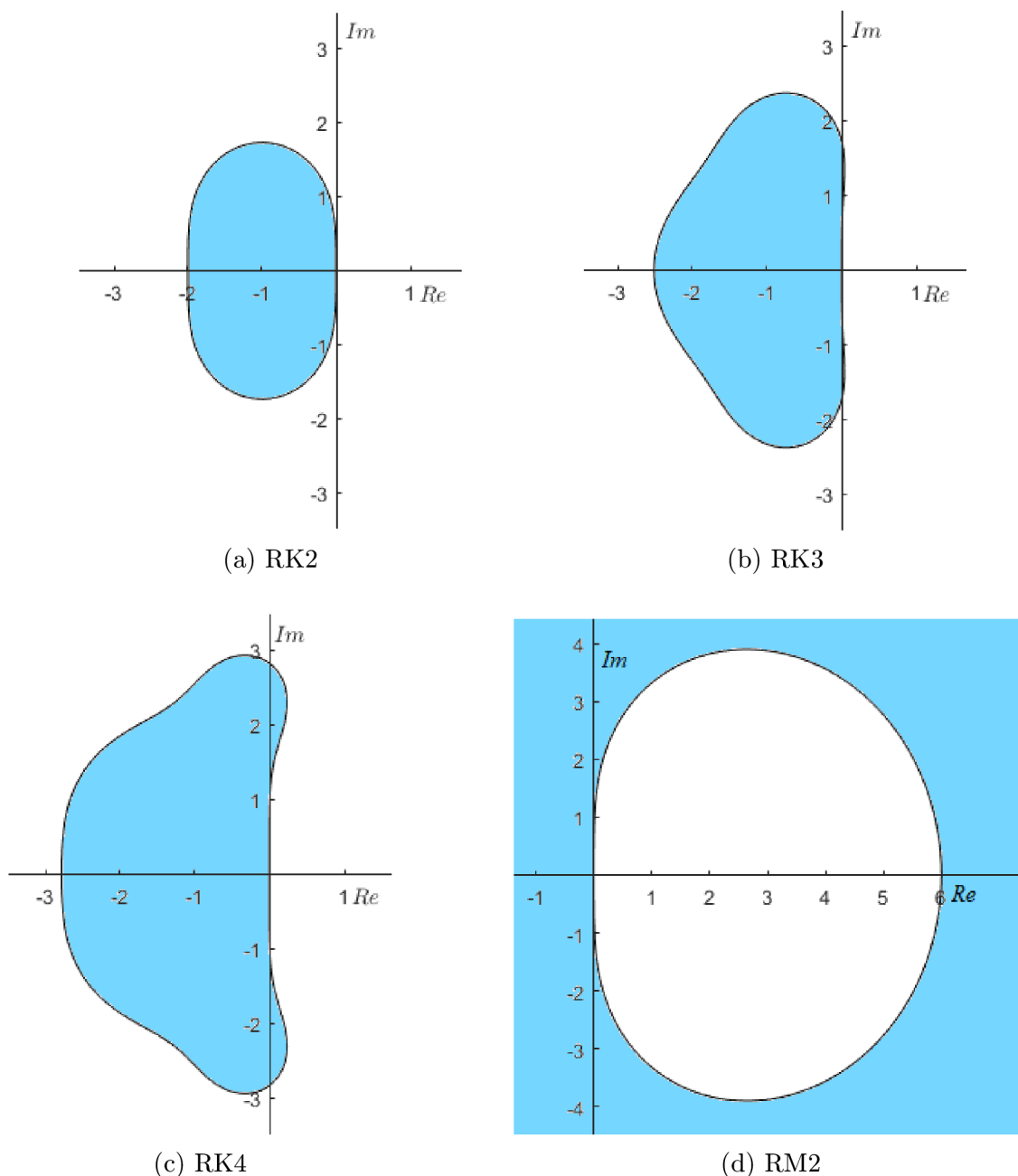
$$\begin{array}{c|cc} 0 & \frac{1}{4} & -\frac{1}{4} \\ \frac{2}{3} & \frac{1}{4} & \frac{5}{12} \\ \hline & \frac{1}{4} & \frac{3}{4} \end{array} \quad \text{a} \quad \begin{array}{c|cc} 0 & 0 & 0 \\ 1 & \frac{1}{2} & \frac{1}{2} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}.$$

Dosazením do (4.3) získáme funkce stability

$$r_1(z) = \frac{1 + \frac{1}{3}z}{1 - \frac{2}{3}z + \frac{1}{6}z^2} \quad \text{a} \quad r_2(z) = \frac{1 + \frac{1}{2}z}{1 - \frac{1}{2}z}.$$

Oba případy postupně rozebereme. V prvním případě jsou póly v bodech $2 \pm \sqrt{2}i$ nacházející se v pravé komplexní polorovině. Zbývá prověřit, zda $|r_1(it)| \leq 1, t \in \mathbb{R}$.

$$|r_1(it)| = \left| \frac{1 + \frac{1}{3}it}{1 - \frac{2}{3}it + \frac{1}{6}(it)^2} \right| = \frac{|1 + \frac{1}{3}it|}{|1 - \frac{2}{3}it - \frac{1}{6}t^2|} \leq 1, \quad t \in \mathbb{R}$$



Obrázek 4.4: Oblast absolutní stability Rungeovy-Kuttovy metody druhého řádu (RK2), Ralstonovy metody (RK3), klasické Rungeovy-Kuttovy metody (RK4) a Radauovy metody (RM2).

Nerovnici upravíme do ekvivalentního tvaru

$$\begin{aligned} \left|1 + \frac{1}{3}it\right|^2 &\leq \left|1 - \frac{2}{3}it - \frac{1}{6}t^2\right|^2, \\ 1 + \frac{1}{9}t^2 &\leq \left(1 - \frac{1}{6}t^2\right)^2 + \frac{4}{9}t^2, \\ 1 + \frac{1}{9}t^2 &\leq 1 + \frac{1}{9}t^2 + \frac{1}{36}t^4. \end{aligned}$$

Nerovnost je splněna pro všechna $t \in \mathbb{R}$, metoda je A-stabilní. Na obrázku 4.4 je zakreslená oblast absolutní stability. V druhém případě máme jediný pól $z = \frac{1}{2}$ ležící v pravé polorovině a $|r_2(it)| = 1$ pro všechna $t \in \mathbb{R}$, tudíž i tato metoda je A-stabilní.

4.2 Stabilita vícekových metod

V další části prozkoumáme stabilitu vícekových metod. Budeme vždy předpokládat, že všechny startovací hodnoty jsou přesné. Řešíme-li testovací úlohu (4.1) lineární více-krokovou metodou (3.6), získáme vztah

$$\sum_{m=0}^s a_m y_{n+m} = h\lambda \sum_{m=0}^s b_m y_{m+n}.$$

Rovnici upravíme do tvaru

$$\sum_{m=0}^s (a_m - h\lambda b_m) y_{m+n} = 0. \quad (4.4)$$

Rovnice (4.4) je příkladem lineární diferenční rovnice

$$\sum_{m=0}^s g_m x^{n+m} = 0, \quad n = 0, 1, \dots,$$

která má charakterický polynom

$$\eta(\omega) = \sum_{m=0}^s g_m \omega^m.$$

Lemma 4.6. *Charakteristický polynom (resp. polynom stability) obecné lineární více-krokové metody má tvar*

$$\eta(z, w) = \sum_{m=0}^s (a_m - z b_m) w^m, \quad z \in \mathbb{C},$$

a oblast absolutní stability lineární vícekové metody je množina

$$\mathcal{D}_{LVM} = \{z \in \mathbb{C} : \eta(z, w) = 0 \implies |w| < 1\}$$

Prozkoumejme oblast absolutní stability Adams-Bashforthovy 2-krokové metody (3.8) aplikovanou na Testovací úlohu (4.1). Charakteristický polynom je tvaru

$$\eta(z, w) = \frac{1}{2}z - \left(1 + \frac{3}{2}z\right)w + w^2,$$

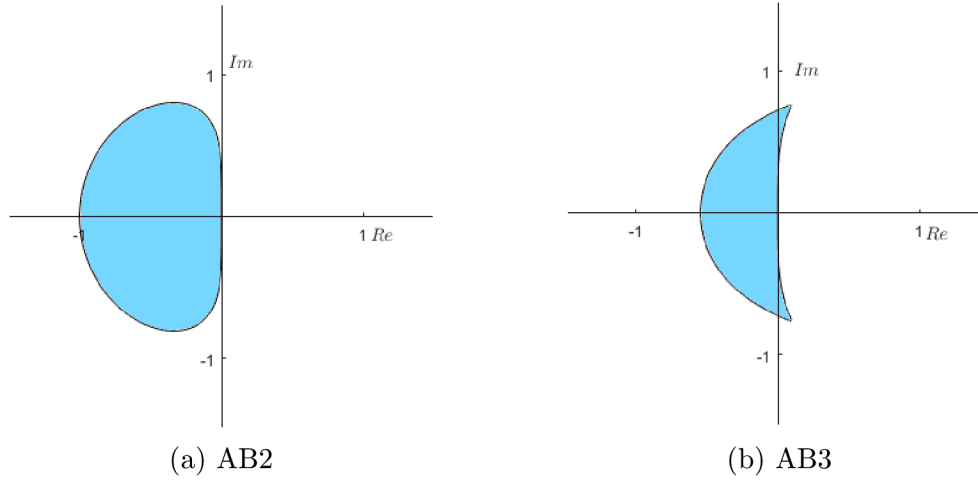
jehož kořeny jsou

$$w = \frac{1}{2} \left(1 + \frac{3}{2}z \pm \sqrt{1 + z + \frac{9}{4}z^2} \right),$$

tudíž oblast absolutní stability AB2 je

$$\mathcal{D}_{AB2} = \left\{ z \in \mathbb{C} : \left| \frac{1}{2} \left(1 + \frac{3}{2}z \pm \sqrt{1 + z + \frac{9}{4}z^2} \right) \right| < 1 \right\}.$$

Na první pohled je již zřejmé, že Adams-Bashforthova 2-kroková metoda není A-stabilní. Na obrázku 4.5 je znázorněna oblast absolutní stability AB2 a AB3. Poznamenejme, že oblast absolutní stability AB_s se zmenšuje s rostoucím s .



Obrázek 4.5: Oblast absolutní stability Adams-Bashforthových metod

Věta 4.7 ([4, Lemma 4.8, s. 66]). *Lineární vícezkroková metoda (3.6) je A-stabilní právě tehdy, když $b_s > 0$ a*

$$|w_1(it)|, |w_2(it)|, \dots, |w_{q(it)}(it)| \leq 1, \quad t \in \mathbb{R},$$

kde $w_1(z), w_2(z), \dots, w_{q(z)}(z)$ jsou kořeny $\eta(z, \cdot)$.

Věta 4.8. (Schurovo-Cohnovo kritérium) *Mějme polynom druhého stupně $aw^2 + bw + c$, kde $a, b, c \in \mathbb{C}, a \neq 0$. Oba kořeny leží v uzávěru jednotkového komplexního kruhu se středem $z = 0$ právě tehdy, když platí*

$$|a|^2 - |c|^2 \leq 0, \quad (|a|^2 - |c|^2)^2 - |a\bar{b} - b\bar{c}|^2 \leq 0, \quad c = a \neq 0 \implies |b|^2 \leq 2|a|. \quad (4.5)$$

Důkaz vycházející z Cohn-Lehmer-Schurova kritéria nalezneme v [1]. Pomocí výše zmíněných kritérií můžeme ověřit, že BDF1 a BDF2 jsou A-stabilní. BDF1 (neboli Eulerova implicitní metoda (3.2)) má charakteristický polynom

$$\eta(z, w) = 1 - (1 - z)w,$$

pro jehož jediný kořen

$$w(z) = \frac{1}{1 - z}$$

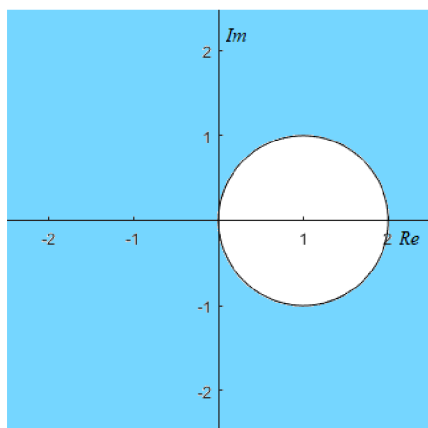
platí

$$|w(it)| = \left| \frac{1}{1 - it} \right| = \frac{|1|}{|1 - it|} = \frac{1}{\sqrt{1 + 1}} = \frac{1}{\sqrt{2}} \leq 1, \quad t \in \mathbb{R}.$$

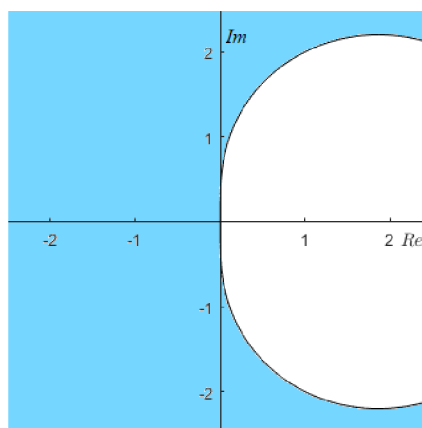
Koeficient $b_1 = 1 > 0$, tudíž můžeme metodu podle věty 4.7 prohlásit za A-stabilní. Analogicky prověříme stabilitu BDF2.

$$\eta(z, w) = \frac{1}{3} - \frac{4}{3}w + \left(1 - \frac{2}{3}z\right)w^2$$

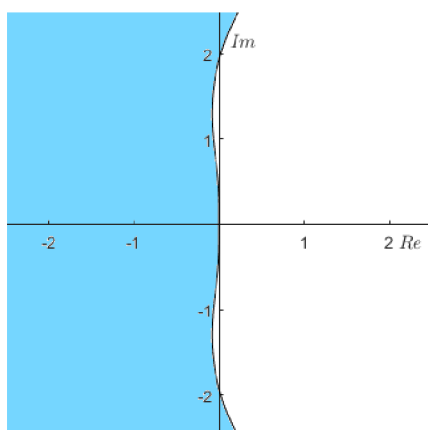
4.2. STABILITA VÍCEKROKOVÝCH METOD



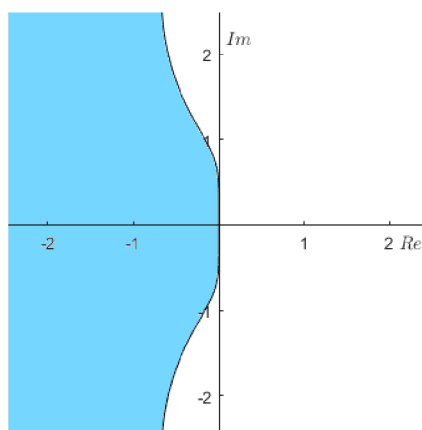
(a) BDF1



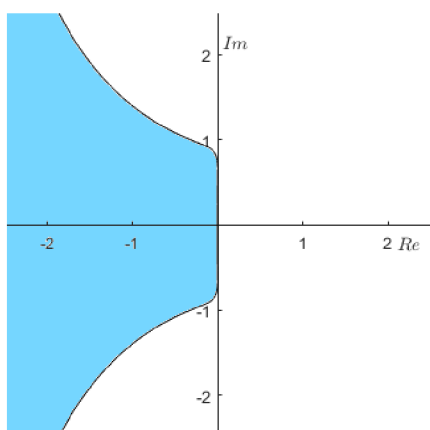
(b) BDF2



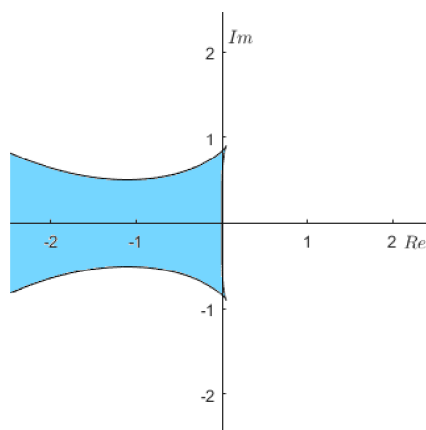
(c) BDF3



(d) BDF4



(e) BDF5



(f) BDF6

Obrázek 4.6: Oblast absolutní stability metod zpětného derivování (BDF)

Koeficient $b_2 = \frac{2}{3} > 0$. Zbývá potvrdit, že pro všechna $t \in \mathbb{R}$ žádný z kořenů $\eta(it, \cdot)$ neopustí jednotkový kruh. Využijeme Schurovo-Cohnovo kritérium, pro $a = 1 - \frac{2}{3}it, b = -\frac{4}{3}, c = \frac{1}{3}$ získáme

$$|a|^2 - |c|^2 = \left|1 - \frac{2}{3}it\right|^2 - \left|\frac{1}{3}\right|^2 = \frac{8}{9} + \frac{4}{9}t^2 \geq 0$$

a

$$(|a|^2 - |c|^2)^2 - |a\bar{b} - b\bar{c}|^2 = \left(\left|1 - \frac{2}{3}it\right|^2 - \left|\frac{1}{3}\right|^2\right)^2 - \left|\left(1 - \frac{2}{3}it\right)\left(-\frac{4}{3}\right) + \frac{4}{3} \cdot \frac{1}{3}\right|^2 = \frac{16}{81}t^4 \geq 0,$$

takže podmínka (4.5) je splněna a metoda je A-stabilní.

Lineární více krokové metody (3.6) řádu 3 a výše již nemohou být A-stabilní. Tato vlastnost je známá jako *Druhá Dahlquistova bariéra*. Zaměřme se na obrázek 4.6, z kterého je patrné, že metody BDFs, $s = 3, 4, 5$ nepatří mezi A-stabilní, ale pro každou oblast stability \mathcal{D} existuje $\alpha \in (0, \pi)$ takové, že nekonečný klín

$$\mathcal{V}_\alpha := \{\rho e^{i\varphi} : \rho > 0, |\rho - \pi| < \alpha\} \subseteq \mathbb{C}^-$$

je součástí \mathcal{D} . Pokud všechna vlastní čísla lineárního systému diferenciálních rovnic leží v \mathcal{V}_α , nezáleží na tom, jak daleko jsou od počátku, takže není nutno redukovat délku kroku pro zachování stability. Metody, pro které $\mathcal{V}_\alpha \subseteq \mathcal{D}$, nazýváme A(α)-stabilní. Všechny metody BDFs, $s = 1, 2, \dots, 6$ jsou A(α)-stabilní, v následující tabulce jsou ve stupních uvedeny α pro jednotlivé metody.

metoda	BDF1	BDF2	BDF3	BDF4	BDF5	BDF6
α	90°	90°	86°	73°	52°	18°

5 Implementace metod v prostředí MATLAB

V této části se budeme zabývat numerickým řešením v prostředí MATLAB. Základní informace o zmíněném softwaru nalezneme v [9]. V programu je zabudováno osm základních funkcí (řešičů) pro numerické řešení počátečního problému. Zaměříme se na čtyři z nich.

- ode23 - Explicitní Rungeova-Kuttova metoda, jejíž základem je Ralstonova metoda řádu 3. Patří mezi jednokrokové metody s omezenou absolutní oblastí stability.
- ode113 - Vícekroková metoda založena na kombinaci explicitní Adams-Bashforthovy a implicitní Adams-Moultonovy metody s proměnným řádem v rozmezí od 1 do 13. Metodu řadíme mezi metody s omezenou oblastí absolutní stability.
- ode23t - Implementace lichoběžníkové metody. Řadí se mezi jednokrokové metody s neomezenou oblastí stability.
- ode15s - Vícekroková metoda založena na metodě zpětného derivování proměnlivého řádu 1 až 5. Patří mezi α -stabilní metody.

Délka kroku

Řešiče implementované v programu MATLAB při řešení počátečního problému (2.3) používají proměnlivou délku kroku, která se zpravidla řídí podle následujících pravidel [2]:

- Uživatel zadá relativní toleranci (relativní přesnost) ε_r , absolutní toleranci (absolutní přesnost) ε_a . Defaultní hodnoty jsou $\varepsilon_r = 10^{-3}$ a $\varepsilon_a = 10^{-6}$. Definujeme $\varepsilon = \max\{\varepsilon_r \max\{|y_n|, |y_{n+1}|\}, \varepsilon_a\}$.
- Program poté určuje délku kroku tak, aby odhadovaná velikost lokální chyby v n -tém kroku $|est_n|$ byla přibližně rovna ε .
- Je-li splněn požadavek $|est_n| \leq \varepsilon$, krok považujeme za úspěšný, v opačném případě je krok neúspěšný a výpočet opakujeme s novou redukovanou délkou kroku.
- Maximální povolená délka jednoho kroku je rovna jedné desetíně délky intervalu, na kterém je úloha řešena, tj. $h_{max} = \frac{1}{10}(b - a)$, minimální přípustná délka intervalu je závislá na tzv. relativní přesnosti aritmetiky počítače označované ε_{mach} .

Velmi zjednodušeně řečeno, program se snaží zvolit co nejdelší délku kroku takovou, aby řešení bylo v toleranci. Analogicky pro soustavu rovnic. Poznamenejme, že jsme nastínili jen základní pravidla. Podrobnější informace lze získat studiem kódů jednotlivých metod.

Řád metody

Metody ode113 a ode15s mohou každou iterací změnit nejen délku kroku, ale také řády metody. Algoritmy s touto vlastností označujeme VSVO (*variable step variable order*). Naznačíme základní myšlenku řízení délky kroku a řádu metody při řešení počáteční úlohy (2.3). Přesnější informace můžeme opět získat studiem kódů [2].

- Předpokládejme, že jsme vypočítali y_{n+1} metodou řádu k s délkou kroku h .
- Určíme odhad est_n^k lokální chyby v n -tém kroku pro k -tý řád.
- Je-li $|est_n^k| \leq \varepsilon$, jedná se o úspěšný krok, v opačném případě je potřeba výpočet opakovat.
- Po úspěšném i neúspěšném kroku je potřeba určit nový řád a novou délku kroku.
- Vypočteme odhad lokální chyby a délku kroku pro řád $k - 1$, k a $k + 1$.
- Z vypočtených délek kroku vybereme nejdelší, tím určíme nový řád a novou délku kroku.

Řešení nelineárních rovnic

Metody ode23t a ode15s vyžadují řešení soustav nelineárních rovnic [2]. Uvažujme vektorovou rovnici ve tvaru

$$\mathbf{y}_{n+1} = h\gamma\mathbf{f}(x_{n+1}, \mathbf{y}_{n+1}) + \boldsymbol{\psi}, \quad (5.1)$$

kde γ konstanta určena metodou a $\boldsymbol{\psi}$ je vektor dané metody nezávislý na \mathbf{y}_{n+1} . Rovnici (5.1) budeme řešit zjednodušenou Newtonovou metodou. Počáteční aproximaci $\mathbf{y}_{n+1}^{(0)}$ určíme extrapolací z předchozích hodnot. Poté řešíme soustavu

$$(I - h\gamma J)(\mathbf{y}_{n+1}^{(s+1)} - \mathbf{y}_{n+1}^{(s)}) = h\gamma\mathbf{f}(x_{n+1}, \mathbf{y}_{n+1}^{(s)}) + \boldsymbol{\psi} - \mathbf{y}_{n+1}^{(s)},$$

přičemž I značí jednotkovou matici a $J = \partial\mathbf{f}(x_{n+1-k}, \mathbf{y}_{n+1-k})/\partial\mathbf{y}$ je Jacobiho matice, kde k je nějaké kladné celé číslo. Označíme

$$G = I - h\gamma J, \quad \mathbf{d}_s = \mathbf{y}_{n+1}^{(s+1)} - \mathbf{y}_{n+1}^{(s)}, \quad \mathbf{g}_s = h\gamma\mathbf{f}(x_{n+1}, \mathbf{y}_{n+1}^{(s)}) + \boldsymbol{\psi} - \mathbf{y}_{n+1}^{(s)}$$

a řešíme soustavu lineárních rovnic $G\mathbf{d}_s = \mathbf{g}_s$. Matice G je závislá na délce kroku h , matici J a typu metody. Soustavu $G\mathbf{d}_s = \mathbf{g}_s$ řešíme pomocí LU rozkladu. Program sám určuje, kdy je vhodné přepočítat Jacobiho matici a provést nový LU rozklad.

Statistika o průběhu řešení

Řešiče implementované v MATLABu poskytují tyto statistiky o průběhu řešení.

- nsteps - počet úspěšných kroků
- nfailed - počet neúspěšných kroků
- nfevals - počet vyhodnocení pravé strany \mathbf{f}
- npds - počet sestavení Jacobiho matice J
- ndecemps - počet provedených LU rozkladů $J = LU$
- nsolves - počet řešených soustav lineárních rovnic $LU\mathbf{d}_s = \mathbf{g}_s$

6 Tuhé systémy

V této kapitole se zaměříme na tuhé systémy (neboli tuhé problémy), způsoby jejich řešení a následně se budeme zabývat fyzikálními a chemickými úlohami, které vedou na tuhé systémy. Úlohy budeme řešit v prostředí MATLAB. V příloze jsou uvedeny skripty k jednotlivým příkladům. Hlavní zdroje pro tuto kapitolu jsou [1], [2] a [5].

Nejprve si vysvětleme, co se skrývá pod pojmem „tuhý systém“. V mnohé literatuře zaobírající se numerickými metodami se objevují definice tuhých systémů, které ale nejsou příliš rigorózní, takže místo korektní definice uvedeme charakteristické vlastnosti, kterými se tuhé systémy vyznačují. Mějme počáteční problém

$$\mathbf{y}' = \mathbf{f}(x, \mathbf{y}), \quad \mathbf{y}(x_0) = \mathbf{y}_0 \quad (6.1)$$

Charakteristika 1. [2] Počáteční problém (6.1) je *tuhý*, když počet kroků, který k jeho vyřešení s danou přesností potřebuje metoda s omezenou oblastí absolutní stability, je podstatně větší než počet kroků, který k jeho vyšetření potřebuje metoda s neomezenou absolutní oblastí stability při stejných nárocích na přesnost.

Charakteristika 2. Počáteční úlohu (6.1) nazveme *tuhou*, jestliže délka kroku vyžadovaná absolutní stabilitou je menší než délka kroku pro splnění přesnosti.

Na následujícím příkladu budeme ilustrovat analýzu tuhosti za použití uvedených charakteristik.

Příklad č. 1

Mějme soustavu diferenciálních rovnic ve tvaru

$$\begin{aligned} y_1' &= -1001y_1 - 1000y_2 & y_1(0) &= 1 & x &\in \langle 0, 5 \rangle. \\ y_2' &= y_1 & y_2(0) &= -1 \end{aligned} \quad (6.2)$$

Soustavu řešíme s přednastavenými tolerancemi. Analytické řešení této soustavy je $y_1 = e^{-x}$, $y_2 = -e^{-x}$. Soustavu postupně necháme řešit v prostředí MATLAB jednotlivými metodami, do tabulky zapíšeme počty úspěšně provedených kroků.

metoda	ode23	ode113	ode15s	ode23t
počet kroků	1839	2967	26	35
oblast absolutní stability	omezená		neomezená	

Vidíme, že metody s neomezenou oblastí stability potřebovaly k řešení úlohy podstatně méně kroků než metody s omezenou oblastí stability, což je v souladu se všemi uvedenými charakteristikami. Metoda ode15s zdolala interval $\langle 0, 5 \rangle$ 26 kroky, zatímco metoda ode113 provedla na stejném intervalu 2967 kroků.

Poznamenejme, že úlohu (6.2) jsme řešili na poměrně krátkém intervalu. Mnoho reálných úloh (např. z oblasti chemické kinetiky) se řeší na intervalu délky 10^{10} . Pokud bychom na takový problém použili metodu s omezenou oblastí stability, pak by i moderní výpočetní stroje nedokázaly vyřešit úlohu v rozumném čase.

Abychom mohli vyslovit další charakteristiku tuhého systému, musíme nejdříve zadefinovat pojem *spektrální poloměr*.

Definice 6.1. *Spektrální poloměr* matice \mathbf{A} (značíme $\rho(\mathbf{A})$) definujeme jako největší absolutní hodnotu z vlastních čísel, tj.

$$\rho(\mathbf{A}) = \max\{|\lambda_1|, |\lambda_2|, \dots, |\lambda_d|\}$$

Charakteristika 3. [2] Problém je tuhý, jestliže všechna vlastní čísla Jacobiho matice $J(x, \mathbf{y}) = \partial \mathbf{f}(x_n, \mathbf{y}_n) / \partial \mathbf{y}$ mají zápornou reálnou část a součin spektrálního poloměru Jacobiho matice s délkou intervalu integrace $\langle a, b \rangle$ je velký, tj.

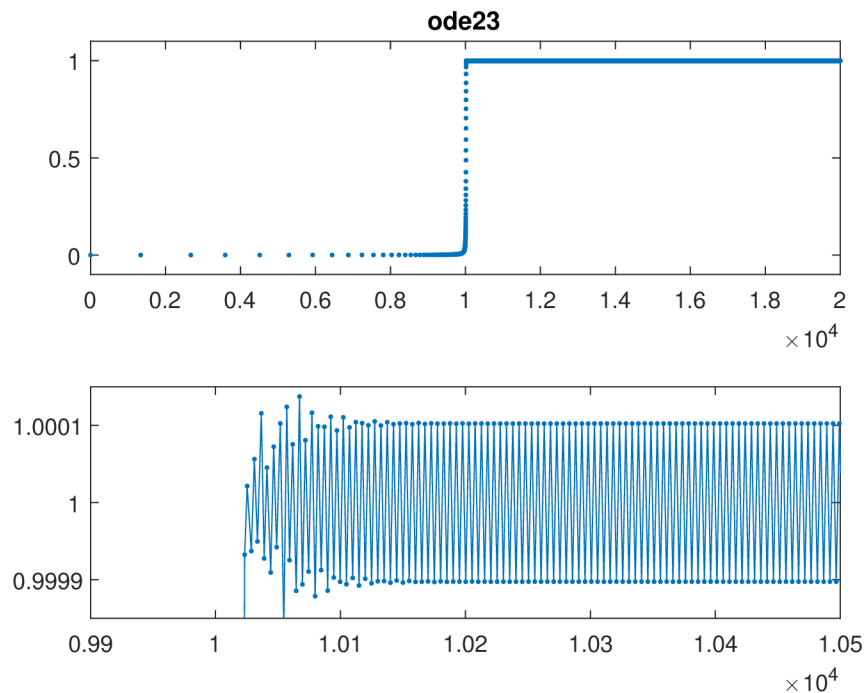
$$\max_{x \in (a, b)} \rho(J(x, \mathbf{y}(x))) (b - a) \gg 1$$

Příklad č. 2

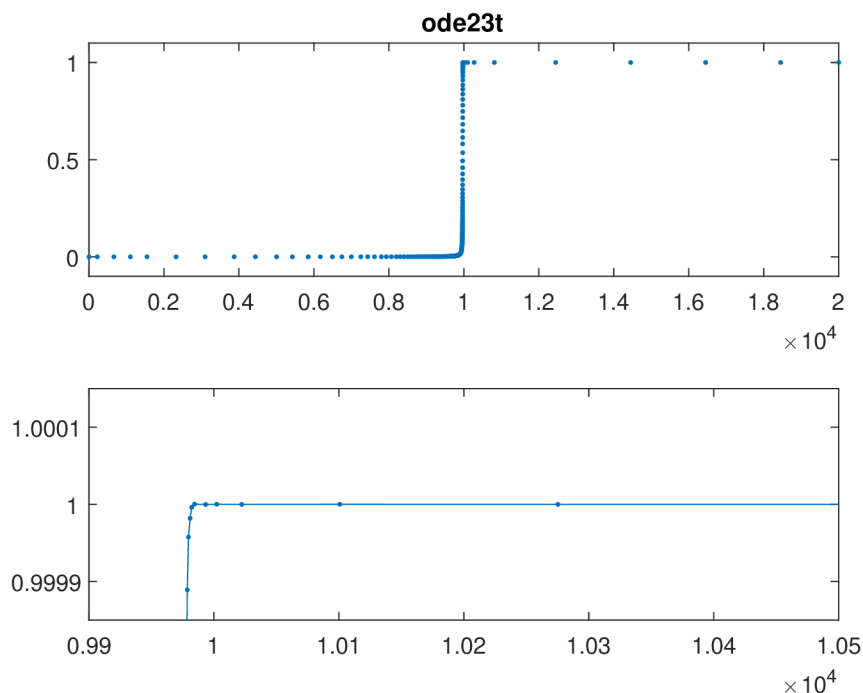
Uvedme si další příklad ze série tuhých problémů, a to model šíření plamene při hoření, viz [2], [6]. Když zapálíme zápalku, oheň se šíří všemi směry a zvětšuje se, dokud nedosáhne kritické velikosti, v níž nastává rovnováha mezi množstvím dodávaného a spalovaného kyslíku. Tento děj můžeme zjednodušeně popsat skalární diferenciální rovnicí

$$y' = y^2 - y^3, \quad y(0) = \delta, \quad x \in (0, 2/\delta), \quad (6.3)$$

kde $y(x)$ reprezentuje poloměr ohnivé koule, výrazy y^2 (resp. y^3) je úměrný velikosti plochy (resp. objemu) koule. Počáteční poloměr δ volíme dostatečně malý. Děj sledujeme po dobu přímo úměrnou $1/\delta$. Úlohu budeme řešit v prostředí MATLAB metodami ode23 a ode23t. Zvolíme $\delta = 10^{-4}$, absolutní přesnost $\varepsilon_a = 10^{-7}$, relativní přesnost $\varepsilon_r = 10^{-4}$.



Obrázek 6.1: Úloha (6.3) řešená ode23, nahoře celý výpočet, dole detail



Obrázek 6.2: Úloha (6.3) řešená ode23t, nahoře celý výpočet, dole detail

Z obrázku 6.1 a 6.2 vidíme, že pro $x \in (0, 9900)$ je $y(x) < 10^{-2}$, poté funkce prudce roste a v intervalu $x \in (10030, 20000)$ je $y(x)$ prakticky rovno jedné. Spektrální poloměr Jacobiho matice je $\rho(J) = |2y - 3y^2|$. Na intervalu $(0, 9900)$ je spektrální poloměr téměř nulový a výraz $|2y - 3y^2|_{9900}$ charakterizující tuhost poměrně malý, proto se zde nejedná o tuhý problém a délka kroku je řízena požadovanou přesností. Na intervalu $(9900, 10030)$ řešení prudce roste a spektrální poloměr se zvětšuje. Je nutné zkrátit délku kroku, aby byla zachována přesnost. V poslední části, tj. na intervalu $(10030, 20000)$, je spektrální poloměr přibližně roven 1, výraz $|2y - 3y^2|_{(20000 - 10300)}$ je dostatečně velký, jedná se o tuhý problém a délka kroku je řízena stabilitou. V následující tabulce uvedeme pro jednotlivé subintervaly a metody statistické údaj nsteps/nfevals (počet úspěšně provedených kroků/počet vyhodnocení pravé strany \mathbf{f}).

interval	(0,9900)	(9900,10030)	(10030,20000)	(0,20000)
ode23	48/145	63/204	3967/11907	4078/12256
ode23t	85/170	100/217	7/12	192/399

Z tabulky vidíme, že na prvních dvou intervalech byly obě metody prakticky stejně úspěšné, zlom ale nastal v poslední části, kde metoda ode23 provedla 3967 kroků, zatímco metoda ode23t stejný interval zdolala jen 7 kroky. Vidíme, že metoda s omezenou oblastí stability není vhodná pro řešení tuhých systémů.

Příklad č. 3

V tomto příkladu se budeme zabývat chemickou kinetikou [1]. Uvažujme tři chemické látky, které označíme písmeny A, B a C, mezi kterými probíhají chemické reakce



kde k_i , $i = 1, 2, 3$ značí veličinu *reakční rychlost*, která určuje, jak rychle se reaktanty přeměňují na produkty. Reakční rychlost je závislá na mnoha parametrech (teplota, tlak, světlo, apod.) a určuje se experimentálně. Označme $y_1(t)$, $y_2(t)$ a $y_3(t)$ koncentrace látek A, B a C v čase t . Rovnice (6.4a) představuje rozpad látky A za vzniku B, přičemž velikost rychlosti změny koncentrace je rovna $k_1 y_1$ (koncentrace A klesá, B roste). V druhém vztahu, tj. (6.4b), probíhá přeměna B na A za přítomnosti katalyzátoru C, velikost rychlosti změny koncentrace je daná součinem $k_2 y_2 y_3$. Poslední rovnice (6.4c) popisuje vznik látky C, jejíž růst koncentrace je určen vztahem $k_3 y_2^2$. Všechny tři reakce probíhají současně, jejich sloučením získáme soustavu diferenciálních rovnic

$$\begin{aligned} y_1' &= -k_1 y_1 + k_2 y_2 y_3, \\ y_2' &= k_1 y_1 - k_2 y_2 y_3 - k_3 y_2^2, \\ y_3' &= k_3 y_2^2. \end{aligned}$$

Za předpokladu, že by všechny tři reakční rychlosti byly stejného řádu, nejednalo by se o nijak složitou úlohu. Komplikace ale nastávají, když se hodnoty k_1 , k_2 a k_3 začnou „extrémně“ lišit. Zvolíme-li hodnoty reakčních rychlostí

$$k_1 = 0,04, \quad k_2 = 10^4, \quad k_3 = 3 \cdot 10^7$$

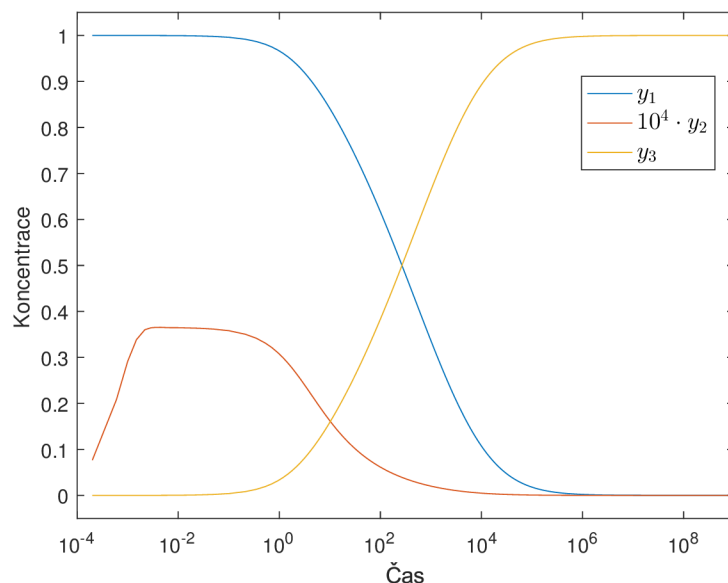
a počáteční podmínky

$$y_1(0) = 1, \quad y_2(0) = 0, \quad y_3(0) = 0,$$

získáme úlohu, která je známou jako *Robertsonův problém*. Jacobiho matice pro Robertsonův problém je tvaru

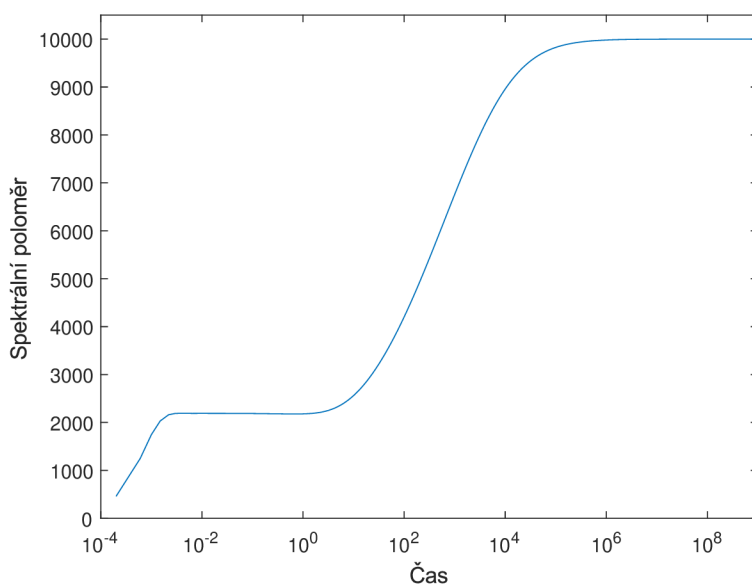
$$J(t, \mathbf{y}) = \begin{pmatrix} -0,04 & 10^4 y_3 & 10^4 y_2 \\ 0,04 & -10^4 y_3 - 6 \cdot 10^7 y_2 & -10^4 y_2 \\ 0 & 6 \cdot 10^7 y_2 & 0 \end{pmatrix}.$$

Z fyzikální podstaty plyne, že součet koncentrace je v každém okamžiku roven 1 a z rovnic (6.4a) - (6.4c) je patrné, že koncentrace látky C roste, proto $y_1 \rightarrow 0$, $y_2 \rightarrow 0$ a $y_3 \rightarrow 1$ pro $t \rightarrow \infty$. Toto tvrzení dokládá obrázek 6.3, který znázorňuje jednotlivé koncentrace v závislosti na čase $t \in \langle 0, 10^9 \rangle$. Koncentrace látky B je po celou dobu velmi malá, pro vizualizaci je v grafu její hodnota zvětšena 10^4 krát.



Obrázek 6.3: Robertsonův problém - koncentrace, ode23t

Vlastní čísla Jacobiho matice v počátku jsou $\{0; 0; -0,04\}$, pro $(y_1, y_2, y_3) \rightarrow (0, 0, 1)$ jsou vlastní čísla $\{0; 0; -10000\}$ a spektrální poloměr $\rho(J) = 10000$. Na obrázku 6.4 je znázorněn průběh spektrálního poloměru v čase. Můžeme tedy říci, že pro dlouhé intervaly se Robertsonův problém jeví tuhý.



Obrázek 6.4: Robertsonův problém - spektrální poloměr

Robertsonův problém jsme řešili metodami ode23t, ode15s a ode23 s přednastavenými přesnostmi. Metodou ode23 se nám problém nepodařilo vyřešit, neboť jen na překonání intervalu $(0, 1000)$ potřebovala 2 284 054 kroků. Naopak metoda ode23t (resp. ode15s) určená k řešení tuhých systému zvládla celou úlohu vyřešit 214 (resp. 212) kroky.

Příklad č. 4

V následující úloze navážeme na problematiku chemických reakcí z minulého příkladu a prozkoumáme podstatu vzniku ozónu v atmosféře [5]. Předpokládejme, že zemská atmosféra je uzavřená, má konstantní teplotu a objem, pak vzájemné působení volného kyslíku O, dvouatomového kyslíku O₂ a ozónu O₃ můžeme popsat reakcemi



Reakční rychlost, k_1 resp. k_2 můžeme považovat za konstantní, reakce (6.5c) a (6.5d) jsou závislé na míře slunečního záření, které způsobuje štěpení dvouatomového kyslíku a ozónu. Opět zavedme označení $y_1(t)$, $y_2(t)$ a $y_3(t)$ pro koncentrace látek O, O₂ a O₃ v čase t . Zaměříme-li se na určitou oblast atmosféry v dané nadmořské výšce, můžeme proces vzniku ozónu zapsat pomocí systému diferenciálních rovnic

$$\begin{aligned} y_1' &= -k_1 y_1 y_2 & -k_2 y_1 y_3 & +2k_3 y_2 & +k_4 y_3, \\ y_2' &= -k_1 y_1 y_2 & +2k_2 y_1 y_3 & -k_3 y_2 & +k_4 y_3, \\ y_3' &= k_1 y_1 y_2 & -k_2 y_1 y_3 & & -k_4 y_3, \end{aligned}$$

kde

$$k_1 = 1,63 \cdot 10^{-16}, \quad k_2 = 4,66 \cdot 10^{-16}.$$

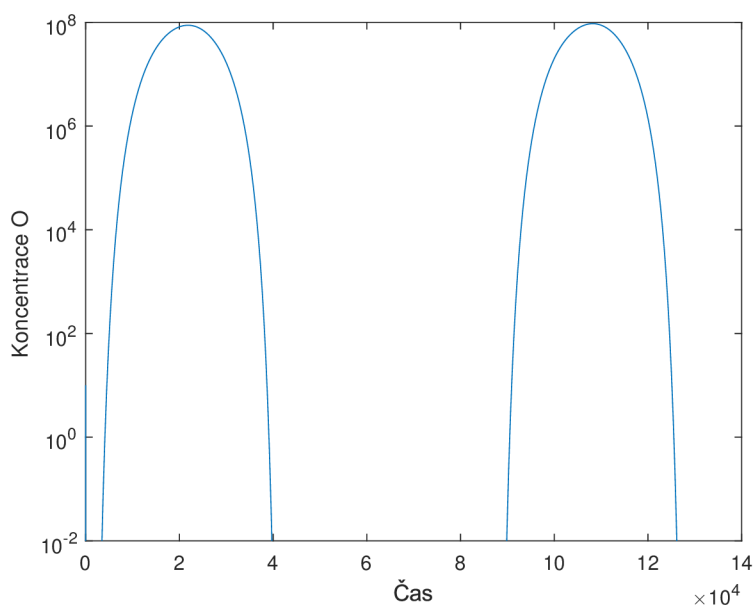
Zbývající dvě reakční rychlosti závislé na slunečním svitu modelujeme funkcí času

$$k_i(t) = \begin{cases} \exp(-c_i/\sin\omega t), & \sin\omega t > 0, \\ 0, & \sin\omega t \leq 0, \end{cases} \quad i = 3, 4.$$

Abychom vhodně napodobili střídání dne a noci, uvažujeme rovnodennost a volíme $\omega = 2\pi/86400$ (perioda 24 hodin), $c_3 = 22,62$, $c_4 = 7,601$. Počátek odpovídá východu slunce (šestá hodina ranní). Hodnota k_3 a k_4 roste, dokud nedosáhne v čase $t = 21600$ (poledne) svého maxima, poté klesá do nuly, kterou dosáhne v čase $t = 43200$ (západ slunce, šestá hodina večerní). Do východu slunce dalšího dne jsou hodnoty k_3 a k_4 nulové. Pozorováním byly zjištěny počáteční hodnoty

$$y_1(0) = 10^1, \quad y_2(0) = 3,7 \cdot 10^{16}, \quad y_3(0) = 10^{12}.$$

Úlohu budeme řešit metodami ode23t, ode15s a ode113 s přesnostmi $\varepsilon_a = 10^{-6}$, $\varepsilon_r = 10^{-4}$ na intervalu $t \in \langle 0, 129600 \rangle$ odpovídající délce 1,5 dne. Koncentrace O₂ a O₃ jsou po celou dobu prakticky konstantní, naopak koncentrace jednoatomového kyslíku O se během dne mění v závislosti na slunečním svitu. Na obrázku 6.5 je zobrazen průběh koncentrace jednoatomového kyslíku během 1,5 dne. Poznamenejme, že svislá osa grafu je v logaritmickém měřítku a čas je počítán v sekundách.



Obrázek 6.5: Ozón v atmosféře - koncentrace jednoatomového kyslíku O , ode15s

Během celého výpočtu jsou všechna vlastní čísla Jacobiho matice záporná, prakticky konstantní. Konkrétně $\lambda_1 \approx -6,03$, zbylá dvě vlastní čísla jsou téměř nulová. Součin spektrálního poloměru Jacobiho matice a délky intervalu, na kterém problém řešíme, je velké číslo, proto můžeme problematiku vzniku ozónu v atmosféře zařadit mezi tuhé problémy. Na závěr uvedeme statistiku pro jednotlivé řešiče.

statistický údaj	ode23t	ode15s	ode113
počet úspěšných kroků	1568	556	385936
počet neúspěšných kroků	106	54	21909
počet vyhodnocení pravé stany \mathbf{f}	2196	842	793782
počet sestavení Jacobiho matice	4	5	
počet provedených LU rozkladů	398	118	
počet řešených soustav lineárních rovnic	2179	812	

Z uvedených dat plyne, že metoda ode15s zvládla danou úlohu vyřešit velmi efektivně, naopak metoda ode113 s omezenou oblastí stability je pro danou úlohu nevhodná.

Příklad č. 5

V posledním příkladu prozkoumáme Van der Polův oscilátor [1]. Jedná se o oscilátor s nelineárním tlumením modelující oscilační procesy v elektrických soustavách, který je popsán vztahem

$$y'' + \mu(y^2 - 1)y' + y = 0, \quad \mu > 0, \quad (6.6)$$

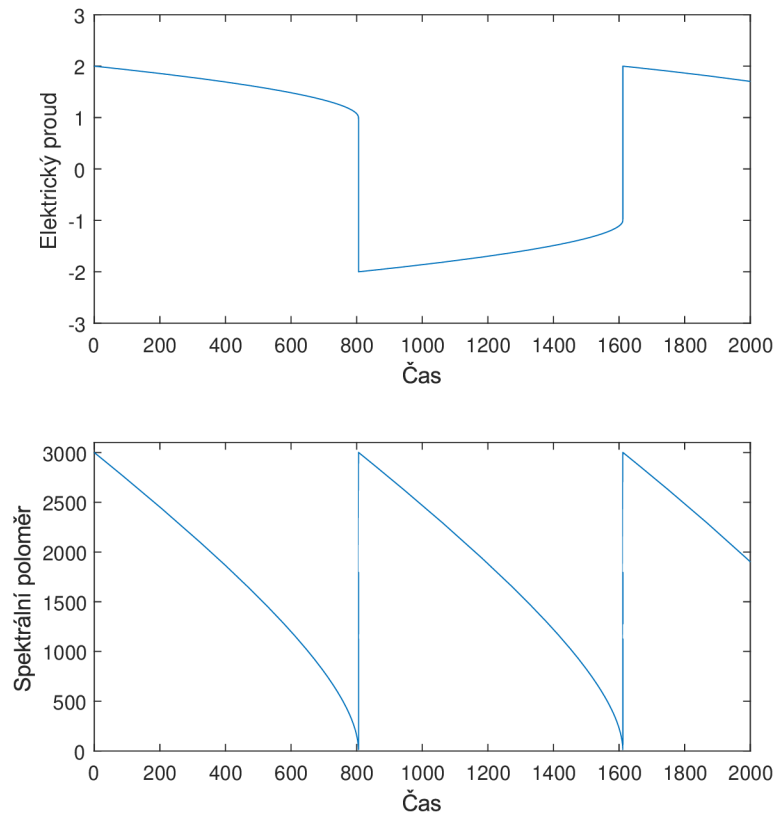
kde y značí velikost elektrického proudu procházející obvodem v čase t a μ je konstanta udávající tlumení. Rovnici (6.6) zapíšeme jako systém dvou diferenciálních rovnic

$$\begin{aligned} y_1' &= y_2, \\ y_2' &= -\mu(y_1^2 - 1)y_2 - y_1, \quad \mu > 0. \end{aligned} \quad (6.7)$$

Příslušná Jacobiho matice je tvaru

$$J(t, \mathbf{y}) = \begin{pmatrix} 0 & 1 \\ -2\mu y_1 y_2 - 1 & -\mu(y_1^2 - 1) \end{pmatrix}.$$

Vlastní čísla Jacobiho matice jsou nejen závislá na konstantě μ , ale mění se i v čase. Zvolme $\mu = 1000$, počáteční hodnoty $y(0) = 2$, $y'(0) = 0$ a řešme úlohu na intervalu $\langle 0, 2000 \rangle$ metodami ode23t, ode15s a ode23 s přednastavenými tolerancemi. Na obrázku 6.6 je zakreslen průběh elektrického proudu a spektrálního poloměru v závislosti na čase.



Obrázek 6.6: Van der Polův oscilátor - závislost velikosti elektrického proudu a spektrálního poloměru na čase, ode15s

Van der Polův oscilátor je periodický s periodou $T \approx 1612s$. Zaměřme se na interval $(0, 806)$ odpovídající půlce periody a porovnejme na něm jednotlivé metody z hlediska počtu kroků. Data zapíšeme do tabulky

interval	(0,5)	(5,780)	(780,806)	(0,806)
ode23t	31	21	221	273
ode15s	35	19	210	264
ode23	5962	565846	2518	574326

Metody ode23t a ode15s si v průběhu celého řešení počínaly prakticky stejně. První část, tj. interval $(0, 5)$, překonala metoda ode23t 31 kroky, zatímco ode23 potřeboval 5962 kroků. Velikost elektrického proudu na intervalu $(5, 780)$ klesala velmi pomalu. Metody určené pro tuhé systémy potřebovaly přibližně 20 kroků, naopak metoda ode23 jich vyžadovala přes půl milionu. V posledním intervalu $(780, 806)$ nastala prudká změna velikosti proudu a délka kroků musela být redukována pro zachování přesnosti. Celkově potřebovala metoda ode23 přibližně 2 000 krát více kroků než zbylé dvě metody, proto i tento problém můžeme označit za tuhý.

7 Závěr

Cílem této práce bylo popsat a analyzovat vybrané numerické metody a zaměřit se na numerické řešení tuhých systémů. Nejprve jsme uvedli teoretický základ obyčejných diferenciálních rovnic a jejich systémů, poté jsme se zaměřili na popis numerických metod určených k jejich řešení. Numerické metody jsme rozdělili na jednokrokové a vícekové, explicitní a implicitní. Metody jsme postupně odvozovali, určovali jejich řády a konvergence. Následně jsme metody zkoumali z hlediska stability. Zjistili jsme, že oblast absolutní stability explicitních metod je zpravidla podstatně menší než u implicitních metod. Pouze implicitní metody mohou mít neomezenou oblast absolutní stability. Také jsme zjistili, že oblast absolutní stability se obvykle s rostoucím řádem metody zmenšuje.

Zbylá část práce se věnuje tuhým systémům a jejich řešením v prostředí MATLAB. Tuhé systémy nemají (prozatím) přesnou definici a k jejich popisu se používají charakteristické vlastnosti. Nejznámější charakteristika říká, že u tuhých systémů potřebuje metoda s omezenou oblastí stability mnohonásobně více kroků než metoda s neomezenou oblastí stability. Následně jsme se věnovali fyzikálním a chemickým úlohám vedoucím na tuhé systémy. Úlohy jsme řešili metodami s omezenou i neomezenou oblastí stability a porovnávali je podle statistik o průběhu řešení. V příloze jsou uvedeny použité skripty. Příklad č. 3 se nepodařilo vyřešit metodou s omezenou oblastí stability z důvodu nedostatečné výpočetní techniky, v ostatních příkladech potřebovaly metody s omezenou oblastí stability podstatně více kroků než ty s neomezenou při stejných požadavcích na přesnost. Z toho můžeme usoudit, že metody s omezenou oblastí stability jsou nevhodné pro řešení tuhých systémů.

8 Seznam použité literatury

- [1] BUTCHER, J. C. *Numerical methods for ordinary differential equations*. Hoboken, NJ: J. Wiley, c2003. ISBN 04-719-6758-0.
- [2] ČERMÁK, L. *Numerické metody pro řešení diferenciálních rovnic* [online]. Brno, Vysoké učení technické: Ústav matematiky, Fakulta strojního inženýrství, 2017 [cit. 2019-05-22]. Dostupné z: http://mathonline.fme.vutbr.cz/download.aspx?id_file=1295
- [3] FRANČŮ, J. *Obyčejné diferenciální rovnice* [online]. Brno, Vysoké učení technické: Ústav matematiky, Fakulta strojního inženýrství, 2012 [cit. 2019-05-22]. Dostupné z: http://www.mat.fme.vutbr.cz/download.aspx?id_file=3358
- [4] ISERLES, A. *A first course in the numerical analysis of differential equations*. 2006. New York: Cambridge University Press, 1996. ISBN 05-215-5655-4.
- [5] KAHANER, D., C. B. MOLER, S. NASH a G. E. FORSYTHE. *Numerical methods and software*. Englewood Cliffs, N.J.: Prentice Hall, 1989. ISBN 01-362-7258-4.
- [6] MOLER, C. *Stiff Differential Equations* [online]. In: . 2003 [cit. 2019-05-22]. Dostupné z: <https://nl.mathworks.com/company/newsletters/articles/stiff-differential-equations.html>
- [7] NECHVÁTAL, L. *Laplaceova transformace a stabilita řešení soustav diferenciálních rovnic* [online]. Brno, Vysoké učení technické: Ústav matematiky, Fakulta strojního inženýrství, 2014 [cit. 2019-05-22]. Dostupné z: http://mathonline.fme.vutbr.cz/download.aspx?id_file=1306
- [8] RALSTON, A. *Základy numerické matematiky*. 2. vyd. Praha: Academia, 1978, 635 s.
- [9] RYBIČKA, J. *LATEX pro začátečníky*. 3. vyd. Brno: Konvoj, 2003. ISBN 80-730-2049-1.
- [10] SEIDU B. *International Journal of Computational and Applied Mathematics* [online]. 2011, 2011(3) [cit. 2019-05-22]. ISSN 1819-4966. Dostupné z: https://www.researchgate.net/publication/249655976_A_Matrix_System_for_Computing_the_Coefficients_of_the_Adams_Bashforth-Moulton_Predictor-Corrector_formulae

9 Seznam použitých zkratek a symbolů

ODR	obyčejná diferenciální rovnice
EE	Eulerova explicitní metoda
IE	Eulerova implicitní metoda
TR	lichoběžníková metoda
ERK	explicitní Rungeova-Kuttova metoda
IRK	implicitní Rungeova-Kuttova metoda
RK3	Ralstonova metoda řádu 3
RK4	klasická Rungeova-Kuttova metoda
RM2	Radauova metoda řádu 2
AB	Adams-Bashforthova metoda
AM	Adams-Moultonova metoda
BDF	metoda zpětného derivování
\mathbb{R}	množina reálných čísel
\mathbb{C}	množina komplexních čísel
$\operatorname{Re}(z)$	reálná část komplexního čísla
$\operatorname{Im}(z)$	imaginární část komplexního čísla
\mathbb{P}_α	množina všech polynomů stupně nejvýše α
$\mathbb{P}_{\alpha/\beta}$	množina všech racionálních funkcí p/q , kde $p \in \mathbb{P}_\alpha$ a $q \in \mathbb{P}_\beta$
\mathcal{D}	oblast absolutní stability
I	jednotková matice
$\det(A)$	determinant matice A
$ \cdot $	absolutní hodnota
$\ \cdot\ $	Eukleidovská norma
$\lfloor \cdot \rfloor$	dolní celá část
(\cdot, \cdot)	otevřený interval
$\langle \cdot, \cdot \rangle$	uzavřený interval

10 Přílohy

Příklad č. 1

```
clear; clc; close all;

soustava = @(x,y) [y(2); -1000*y(1)-1001*y(2)];
y0 = [1, -1]; %pocatecni podminky
konec = 5; %koncovy bod intervalu

sol1 = ode23(soustava,[0 konec],y0); %reseni ode23
sol2 = ode113(soustava,[0 konec],y0); %reseni ode113
sol3 = ode15s(soustava,[0 konec],y0); %reseni ode15s
sol4 = ode23t(soustava,[0 konec],y0); %reseni ode23t

%pocet kroku pro ode23, ode113, ode115s, ode23t
nsteps = [sol1.stats.nsteps, sol2.stats.nsteps,
          sol3.stats.nsteps, sol4.stats.nsteps]
```

Příklad č. 2

```
clear; clc; close all;

funkce = @(x,y) 1*y.^2 - y.^3 ;
delta = 0.0001 ;
konec = 2/delta; %koncovy bod intervalu
options = odeset('RelTol',1e-4,'AbsTol',1e-7); %urceni toleranci

sol1 = ode23(funkce,[0 konec],delta,options); %reseni ode23
sol2 = ode23t(funkce,[0 konec],delta,options); %reseni ode23t

nsteps = [sol1.stats.nsteps, sol2.stats.nsteps] %pocet kroku
nfevals = [sol1.stats.nfevals, sol2.stats.nfevals]
          %pocet vyhodnoceni prave strany f

figure
subplot(2,1,1);
    plot(sol1.x,sol1.y,'.')
    title('ode23')
    axis([0 konec -0.1 1.1])
subplot(2,1,2);
    plot(sol1(1).x,sol1(1).y,'.-')
    axis([1/delta*0.99 1/delta*1.05 1-1.5*delta 1+1.5*delta])

figure
subplot(2,1,1);
    plot(sol2(1).x,sol2(1).y,'.')
    axis([1/delta*0.99 1/delta*1.05 1-1.5*delta 1+1.5*delta])
```



```

    title('ode23t')
    axis([0 konec -0.1 1.1])
subplot(2,1,2);
    plot(sol2(1).x,sol2(1).y,'.-')
    axis([1/delta*0.99 1/delta*1.05 1-1.5*delta 1+1.5*delta])

```

Příklad č. 3

```

clear; clc; close all;

soustava = @(x,y) [
-0.04*y(1)+10000*y(2).*y(3);
0.04*y(1)-10000*y(2).*y(3)-3*10^7*y(2).*y(2);
3*10^7*y(2).*y(2) ];

J = @(x,y) [-0.04,          10000*y(3),          10000*y(2);
            0.04, -10000*y(3)-6*10^7*y(2),      -10000*y(2);
            0,           6*10^7*y(2),           0];
            %Jacobiho matice

konec = 10^9 ; %koncovy bod intervalu

sol1 = ode23t(soustava,[0 konec],[1 0 0]); %reseni ode23t
sol2 = ode15s(soustava,[0 konec],[1 0 0]); %reseni ode15s
sol3 = ode23(soustava,[0 konec],[1 0 0]); %reseni ode23

nsteps = [sol1.stats.nsteps, sol2.stats.nsteps, sol3.stats.nsteps ];
          %pocet kroku

spektralni_polomer = zeros(1,sol1.stats.nsteps);
for i = 1: sol1.stats.nsteps+1
    xx = sol1.x(i); %cas v i-tem kroku
    yy = [sol1.y(1,i), sol1.y(2,i), sol1.y(3,i)];
        %koncentrace v i-tem kroku
    Jac = J(xx,yy); %vypocet Jacobiho matice v i-tem kroku
    spektralni_polomer(i) = max(abs(eig(Jac)));
        %spektralni polomer Jacobiho matice v i-tem kroku
end

figure
plot(sol1.x,sol1.y(1,:),'-',sol1.x,10^4*sol1.y(2,:),'-',
      sol1.x,sol1.y(3,:),'-')
set(gca, 'XScale','log', 'Xlim',[10^-4 10^9],
        'Ylim', [-0.05 1.05])
xlabel('Cas');
ylabel('Koncentrace');
set(legend('$y_1$', '$10^4 \cdot y_2$', '$y_3$'),'Interpreter',
      'latex', 'FontSize',12);

```

```

figure
plot(sol1.x, spektralni_polomer)
set(gca, 'XScale', 'log', 'Xlim', [10^-4 10^9],
        'Ylim', [0 10500] )
xlabel('Cas');
ylabel('Spektralni polomer');

```

Příklad č. 4

```

clear; clc; close all;

k1 = 1.63 * 10^-16; %reakcni rychlost k1
k2 = 4.66 * 10^-16; %reakcni rychlost k2
c3 = 22.62;
c4 = 7.601;
omega = 2*pi/86400;

k3 = @(t) exp(-c3./sin(omega*t)).* (sign(sin(omega*t))/2+ 0.5) ;
      %reakcni rychlost k3
k4 = @(t) exp(-c4./sin(omega*t)).* (sign(sin(omega*t))/2+ 0.5) ;
      %reakcni rychlost k4

soustava = @(x,y)
[-k1*y(1)*y(2) - k2*y(1)*y(3) + 2*k3(t)*y(2) + k4(t)*y(3);
 -k1*y(1)*y(2) + 2*k2*y(1)*y(3) - k3(t)*y(2) + k4(t)*y(3);
 k1*y(1)*y(2) - k2*y(1)*y(3) - k4(t)*y(3)];

J = @(x,y)
[ -k1*y(2) - k2*y(3), -k1*y(1) + 2*k3(t), -k2*y(1) + k4(t);
 -k1*y(2) + 2*k2*y(3), -k1*y(1) - k3(t), 2*k2*y(1) + k4(t);
 k1*y(2) - k2*y(3), k1*y(1), -k2*y(1) - k4(t)];
      %Jacobiho matice

y0 = [10^1 3.7 * 10^16 10^12]; %pocatecni podminky
konec = 24*3600*1.5; %koncovy bod intervalu
options = odeset('RelTol',1e-4,'AbsTol',1e-6); %nastaveni toleranci

sol1 = ode23t(soustava,[0 konec],y0,options); %reseni ode23t
sol2 = ode15s(soustava,[0 konec],y0,options); %reseni ode15s
sol3 = ode113(soustava,[0 konec],y0,options); %reseni ode113

nsteps = [sol1.stats.nsteps sol2.stats.nsteps sol3.stats.nsteps];

sol1.stats %vypsani statistik ode23t
sol2.stats %vypsani statistik ode15s
sol3.stats %vypsani statistik ode113

```

```

spektralni_polomer = zeros(1,nsteps(1)+1);
for i = 1: nsteps(1)+1
    xx = sol1.x(i); %cas v i-tem kroku
    yy = [sol1.y(1,i), sol1.y(2,i), sol1.y(3,i)];
    %konentrace v i-tem kroku
    Jac = J(xx,yy); %vypocet Jacobiho matice v i-tem kroku
    vlc = eig(Jac);
    spektralni_polomer(i) = max(abs(vlc));
    %spektralni polomer Jacobiho matice v i-tem kroku
end

```

```

fprintf('Spektralni polomwr Jacobiho matice je mezi %f a %f.\n',min(spektralni_polomer), max(spektralni_polomer))

```

```

figure

```

```

    plot(sol1.x, sol1.y(1,:))
    set(gca, 'YScale', 'log', 'Ylim', [10^-2 10^8])
    xlabel('Cas');
    ylabel('Koncentrace O');

```

Příklad č. 5

```

clear; clc; close all;

```

```

my = 1000; %tlumici konstanta

```

```

soustava = @(x,y) [y(2); my*(1-y(1)^2)*y(2)-y(1)]; %soustva
J = @(x,y) [ 0, 1;
            -2*my*y(1)*y(2)-1, -my*(y(1)^2-1)]; %Jacobiho matice

```

```

konec = 3000; %koncovy bod intervalu
y0 = [2 0]; %pocatecni podminky

```

```

sol1 = ode23t(soustava,[0 konec],y0); %reseni ode23t
sol2 = ode15s(soustava,[0 konec],y0); %reseni ode15s
sol3 = ode23(soustava,[0 konec],y0); %reseni ode23

```

```

nsteps = [sol1.stats.nsteps, sol2.stats.nsteps, sol3.stats.nsteps]

```

```

pocet_kroku = zeros(3,4); %tabulka s pocty kroku na intervalech
pocet_kroku(1,1) = length(find(sol1.x<5));
pocet_kroku(2,1) = length(find(sol2.x<5));
pocet_kroku(3,1) = length(find(sol3.x<5));
pocet_kroku(1,2) = length(find(sol1.x>5 & sol1.x<780));
pocet_kroku(2,2) = length(find(sol2.x>5 & sol2.x<780));
pocet_kroku(3,2) = length(find(sol3.x>5 & sol3.x<780));
pocet_kroku(1,3) = length(find(sol1.x>780 & sol1.x<806));
pocet_kroku(2,3) = length(find(sol2.x>780 & sol2.x<806));

```

```

pocet_kroku(3,3) = length(find(sol3.x>780 & sol3.x<806));
pocet_kroku(1,4) = length(find(sol1.x<806));
pocet_kroku(2,4) = length(find(sol2.x<806));
pocet_kroku(3,4) = length(find(sol3.x<806));
pocet_kroku

spektralni_polomer = zeros(1,nsteps(2)+1);
    for i = 1: nsteps(1)+1
        xx = sol1.x(i);
        yy = [sol1.y(1,i), sol1.y(2,i)];
        Jac = J(xx , yy);
        spektralni_polomer(i) = max(abs(eig(Jac)));
        %spektralni polomer Jacobiho matice v i-tem kroku
    end

figure
subplot(2,1,1)
    plot(sol1.x, sol1.y(1,:))
    xlabel('Cas');
    ylabel('Elektricky proud');
subplot(2,1,2)
    plot(sol1.x, spektralni_polomer)
    set(gca, 'Ylim', [0 3100])
    xlabel('Cas');
    ylabel('Spektralni polomer');

```