



Barycentrická racionální interpolace a její aplikace
Barycentric rational interpolation and its applications

Jiří Cingel

Bakalářská práce
Vedoucí práce: Ing. Marie Sadowská, Ph. D.
Ostrava, 2022

Zadání bakalářské práce

Student:

Jiří Cingel

Studijní program:

B0541A170008 Výpočetní a aplikovaná matematika

Téma:

Barycentrická Lagrangeova interpolace a její aplikace
Barycentric Lagrange interpolation and its applications

Jazyk vypracování:

čeština

Zásady pro vypracování:

Polynomiální interpolace je často používaná numerická metoda pro nahrazení dané funkce vhodným polynomem, přičemž Lagrangeova interpolace je jednou z jejích základních variant. V této práci si ukážeme variantu Lagrangeovy interpolace, tzv. barycentrickou Lagrangeovu interpolaci, o které je známo, že je nejen elegantní, ale také rychlá a stabilní. Poskytuje taktéž větší míru flexibility. Budeme se zabývat především propojením barycentrické a racionální interpolace, kdy hledáme interpolant (tj. nahrazující funkci) ve tvaru podílu dvou polynomů.

Seznam doporučené odborné literatury:

[1] J.-P. Berrut, Lloyd N. Trefethen: Barycentric Lagrange Interpolation. SIAM Review 46, 501-517, 2004

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **Ing. Marie Sadowská, Ph.D.**

Datum zadání: 01.09.2021

Datum odevzdání: 30.04.2022

prof. RNDr. Jiří Bouchala, Ph.D.
vedoucí katedry

prof. Ing. Jan Platoš, Ph.D.
děkan fakulty

Zde bych rád poděkoval Ing. Marii Sadowské, Ph. D. za aktivní zodpovídání dotazů, trpělivost a ochotu, kterou mi při tvorbě této bakalářské práce poskytla.

Abstrakt

Pro nalezení přibližných hodnot funkce na daném intervalu se v numerice používá metoda s názvem polynomiální interpolace. Jeden ze známých způsobů její realizace je Lagrangeova interpolace. Cílem této práce je seznámení se s barycentrickou variantou Lagrangeovy interpolace, která nám poskytuje lepší výpočetní vlastnosti, jako je stabilita, flexibilita a menší časová náročnost. Tato varianta se pak dá dále rozšířit pomocí barycentrické racionální interpolace, jejíž interpolant má tvar zlomku s polynomem v čitateli i jmenovateli.

Klíčová slova

Polynomiální interpolace, Lagrangeova interpolace, Newtonova interpolace, barycentrická Lagrangeova interpolace, racionální interpolace, barycentrická racionální interpolace

Abstract

In numerical mathematics, the method used for approximating function values on a given interval is called polynomial interpolation. One of its implementations is known as Lagrange interpolation. The goal of this thesis is to learn about the barycentric form of Lagrange interpolation, which provides better computational properties such as stability, flexibility and lesser complexity. This variant can be expanded further with the use of barycentric rational interpolation, which interpolant takes form of a fraction with a polynomial in numerator and denominator.

Keywords

Polynomial interpolation, Lagrange interpolation, Newton interpolation, barycentric Lagrange interpolation, rational interpolation, barycentric rational interpolation

Seznam značení

| | |
|--------------------------------------|---|
| \mathbb{P}_n | prostor všech reálných polynomů stupně nejvýše n |
| $\mathbb{R}_{m,n}$ | prostor všech racionálních funkcí P/Q , kde $P \in \mathbb{P}_m$, $Q \in \mathbb{P}_n$ |
| $\mathbb{C}^n(\langle a, b \rangle)$ | prostor všech n -krát spojitě derivovatelných funkcí na $\langle a, b \rangle$ |
| $O(f(n))$ | asymptotická složitost v závislosti na n |
| $\ f\ _\infty$ | $\max_{\xi \in \langle a, b \rangle} f(\xi) $, maximová norma |
| sgn | funkce signum, znaménková funkce |
| $\vec{0}$ | nulový sloupcový vektor |
| \setminus | množinový rozdíl |
| \triangle | konec příkladu |
| \square | konec důkazu |

Seznam zkratk

| | |
|------|--|
| FLOP | operace v plovoucí řádově čárce (FLoating-point OPeration) |
| SLR | systém lineárních rovnic |

Obsah

| | | |
|----------|---|-----------|
| 1 | Úvod | 3 |
| 2 | Polynomiální interpolace | 4 |
| 2.1 | Lagrangeova interpolace | 4 |
| 2.2 | Newtonova forma interpolace | 5 |
| 2.3 | Existence a jednoznačnost interpolačního polynomu | 7 |
| 2.4 | Odhad chyby interpolace | 8 |
| 2.4.1 | Ekvidistantní uzly | 9 |
| 2.4.2 | Čebyševovy uzly | 10 |
| 2.5 | Numerické experimenty | 11 |
| 3 | Barycentrická Lagrangeova interpolace | 15 |
| 3.1 | Výhodnější formulace Lagrangeovy interpolace | 15 |
| 3.2 | Barycentrický tvar Lagrangeovy interpolace | 16 |
| 3.3 | Barycentrické váhy | 17 |
| 3.3.1 | Ekvidistantní uzly | 17 |
| 3.3.2 | Čebyševovy uzly | 18 |
| 3.4 | Numerické experimenty | 18 |
| 4 | Racionální interpolace | 20 |
| 4.1 | Úvod k racionální interpolaci | 20 |
| 4.2 | Existence a jednoznačnost racionální interpolace | 22 |
| 4.3 | Řešitelnost úlohy racionální interpolace | 23 |
| 4.4 | Metoda inverzních diferencí | 24 |
| 4.5 | Numerické experimenty | 27 |
| 5 | Barycentrická racionální interpolace | 30 |
| 5.1 | Barycentrický tvar racionální interpolace | 30 |
| 5.2 | Odstranění bodů nespojitosti pomocí Berrutových vah | 32 |
| 5.3 | Floaterovy-Hormannovy váhy | 34 |
| 5.3.1 | Ekvidistantní uzly | 37 |
| 5.4 | Numerické experimenty | 38 |
| 6 | Závěr | 41 |

| | |
|----------------------------------|-----------|
| <i>OBSAH</i> | 2 |
| Literatura | 42 |
| Příloha A: Kódy v MATLABU | 43 |

Kapitola 1

Úvod

Existuje řada metod k nalezení řešení úlohy interpolace, tj. k nalezení funkce, jejíž graf prochází zadanou množinou bodů. Každá z nich disponuje svými výhodami i nevýhodami, mohou se lišit zejména výpočetní náročností, stabilitou či flexibilitou. Základní metody interpolace (např. Lagrangeova interpolace nebo Newtonova interpolace) nám poskytují kvality, se kterými se občas nemusíme spokojit. V této práci si ukážeme metody a formy interpolace, které tyto vlastnosti postupně vylepšují.

Barycentrická Lagrangeova interpolace je flexibilní a stabilní [3]. Nezávislost barycentrických vah na složce y nám umožňuje jejich znovupoužití pro několik souborů dat s totožným rozdělením uzlů na vodorovné ose. U polynomiální interpolace hraje rozložení uzlů velkou roli na velikosti maximální chyby interpolace, přičemž Čebyševovo rozdělení nám poskytuje nejlepší výsledky. Barycentrická racionální interpolace obecně nepotřebuje speciální rozložení uzlů pro dobré výsledky. Správnou volbou barycentrických vah pak můžeme zabránit existenci bodů nespojitosti [6] a tyto váhy lze dále volit tak, abychom interpolant vůči interpolované funkci více zpřesnili [7].

V kapitole 2 si definujeme pojem polynomiální interpolace a budeme se zabývat dvěma jejími metodami, a to Lagrangeovou interpolací a Newtonovou interpolací. Ukážeme si, že každé úloze nalezení polynomiální interpolace náleží právě jedno řešení. Vysvětlíme si, jak odhadnout maximální chybu interpolace a jak ji lze pomocí vybraných rozložení uzlů ovlivnit. Kapitola 3 bude věnována vylepšení Lagrangeovy interpolace do barycentrického tvaru, který nám poskytne řadu výhod oproti metodám z kapitoly 2. Definujeme si tzv. barycentrické váhy a ukážeme, jak vypadají pro konkrétní rozložení uzlů. Čtvrtá kapitola objasní úlohu racionální interpolace. Předvedeme si, že pokud řešení této úlohy existuje, tak není v jistém smyslu jednoznačné. Odvodíme si také způsob nalezení řešení pomocí metody inverzních diferencí. V poslední kapitole definujeme pojem barycentrické racionální interpolace a ukážeme si, že vhodnou volbou barycentrických vah se zbavíme nechtěných bodů nespojitosti v řešení. Konec každé kapitoly obsahuje numerické pokusy daného způsobu interpolace. V příloze A jsou zmíněné metody interpolace implementovány v MATLABu.

Kapitola 2

Polynomiální interpolace

V této kapitole si ukážeme základy polynomiální interpolace, které můžeme nalézt v knihách W. Shen [1], J. Stoera a R. Bulirsch [4].

Mějme jako vstupní data dáno $n + 1$ bodů (x_i, y_i) , kde všechna x_i jsou navzájem různá ($n \in \mathbb{N}$, $i \in \{0, 1, \dots, n\}$). Cílem interpolace je nalézt vhodnou funkci f , která těmito body prochází, tedy pro každé x_i a y_i platí tzv. interpolační podmínky

$$f(x_i) = y_i, \quad i = 0, 1, \dots, n. \quad (2.1)$$

V metodách interpolace, kterými se budeme v této kapitole zabývat, je hledaná funkce ve tvaru reálného polynomu, jehož stupeň je nejvýše n :

$$f(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0 \in \mathbb{P}_n.$$

2.1 Lagrangeova interpolace

Definujme pro každé $i \in \{0, 1, \dots, n\}$ Lagrangeovy báze funkce L_i následovně:

$$L_i(x) = \prod_{\substack{k=0 \\ k \neq i}}^n \frac{x - x_k}{x_i - x_k}. \quad (2.2)$$

Tyto funkce disponují důležitou vlastností. Lze vidět, že dosazením x_i za x získáme po vykrácení zlomku hodnotu 1. Dosazením jiné hodnoty ze vstupních dat nám však vyjde 0 z důvodu nulovosti jednoho členu v čitateli, tj.

$$L_i(x_j) = \begin{cases} 1, & j = i, \\ 0, & j \neq i, \end{cases} \quad i, j \in \{0, 1, \dots, n\}.$$

Tato kvalita nám umožní sestavit Lagrangeův interpolační polynom ve tvaru

$$L(x) = \sum_{i=0}^n L_i(x) y_i. \quad (2.3)$$

Funkční hodnota $L(x_j)$ v jednom ze zadaných bodů pak bude obsahovat jenom j -tý člen sumy, ostatní členy budou nulové. Jelikož je navíc tento člen roven y_j , interpolační polynom splňuje podmínky (2.1).

Tato metoda má nevýhodu v tom, že pokud bychom získali přídatná vstupní data a chtěli je zakomponovat do naší již sestavené interpolace, musíme všechny báze funkce vypočítat znovu, tj. metoda není flexibilní. Pokaždé tedy musíme počítat s náročností na výpočet v řádu $O(n^2)$ FLOP. Následující metoda tento problém mít však nebude.

2.2 Newtonova forma interpolace

Začneme od příkladu s jedním vzorkem dat (x_0, y_0) . Interpolační polynom volíme jako konstantní funkci s hodnotou y_0 :

$$N_0(x) = y_0.$$

Pro následující vzorek (x_1, y_1) přidáme nový člen polynomu tak, aby v bodě x_0 byla zachována původní hodnota y_0 :

$$N_1(x) = N_0(x) + a_1(x - x_0).$$

Ze členu $(x - x_0)$ je zachování hodnoty v x_0 zřejmé. Nyní potřebujeme zvolit číslo a_1 tak, aby byla splněna interpolační podmínka $N_1(x_1) = y_1$, čili dosadíme x_1 a dostaneme

$$\begin{aligned} N_1(x_1) &= N_0(x_1) + a_1(x_1 - x_0), \\ a_1 &= \frac{N_1(x_1) - N_0(x_1)}{x_1 - x_0} \\ &= \frac{y_1 - y_0}{x_1 - x_0}. \end{aligned}$$

Obdobným postupem pokračujeme dále. Pro nový vzorek (x_2, y_2) opatříme polynom členem, který nezmění hodnoty v žádném předchozím x_i , $i \in \{0, 1\}$, tedy

$$N_2(x) = N_1(x) + a_2(x - x_0)(x - x_1).$$

Z interpolační podmínky $N_2(x_2) = y_2$ vypočteme a_2 :

$$N_2(x_2) = N_1(x_2) + a_2(x_2 - x_0)(x_2 - x_1),$$

a proto

$$\begin{aligned}
 a_2 &= \frac{N_2(x_2) - N_1(x_2)}{(x_2 - x_0)(x_2 - x_1)} \\
 &= \frac{y_2 - y_0 - \frac{y_1 - y_0}{x_1 - x_0}(x_2 - x_0)}{(x_2 - x_0)(x_2 - x_1)} \\
 &= \frac{y_2 - y_0 - \frac{y_1 - y_0}{x_1 - x_0}(x_2 - x_1 + x_1 - x_0)}{(x_2 - x_0)(x_2 - x_1)} \\
 &= \frac{y_2 - y_0 - \frac{y_1 - y_0}{x_1 - x_0}(x_2 - x_1) - \frac{y_1 - y_0}{x_1 - x_0}(x_1 - x_0)}{(x_2 - x_0)(x_2 - x_1)} \\
 &= \frac{y_2 - y_0 - \frac{y_1 - y_0}{x_1 - x_0}(x_2 - x_1) - y_1 + y_0}{(x_2 - x_0)(x_2 - x_1)} \\
 &= \frac{y_2 - y_1 - \frac{y_1 - y_0}{x_1 - x_0}(x_2 - x_1)}{(x_2 - x_0)(x_2 - x_1)} \\
 &= \frac{\frac{y_2 - y_1}{x_2 - x_1} - \frac{y_1 - y_0}{x_1 - x_0}}{x_2 - x_0}.
 \end{aligned}$$

Stejně bychom postupovali při přidávání dalších dat. Obecně můžeme zapsat rekurentní vztah pro Newtonův interpolační polynom s $k + 1$ vzorky:

$$\begin{aligned}
 N_0(x) &= a_0 = y_0, \\
 N_k(x) &= N_{k-1}(x) + a_k(x - x_0)(x - x_1) \cdots (x - x_{k-1}),
 \end{aligned} \tag{2.4}$$

kde koeficient a_k je roven

$$a_k = \frac{y_k - N_{k-1}(x_k)}{(x_k - x_0)(x_k - x_1) \cdots (x_k - x_{k-1})}, \quad k \in \mathbb{N}.$$

Sepsáním vztahů (2.4) do jednoho vzorce získáme předpis

$$N_k(x) = a_0 + a_1(x - x_0) + \cdots + a_k(x - x_0)(x - x_1) \cdots (x - x_{k-1}), \tag{2.5}$$

jehož vyčíslení má složitost $O(k^2)$ FLOP. Postupným vytýkáním členů $(x - x_i)$ dostaneme tzv. vnořený tvar polynomu

$$N_k(x) = a_0 + (x - x_0)(a_1 + (x - x_1)(a_2 + \cdots + (x - x_{k-1})(a_k))), \tag{2.6}$$

který se používá při výpočtech, jelikož jeho vyčíslení stojí oproti (2.5) jen $O(k)$ FLOP.

Předvedme si nyní algoritmus, kterým se dají vypočíst koeficienty a_i , $i \in \{0, 1, \dots, k\}$. Pro tento účel si definujme speciální funkci f , tzv. podílovou dife-

renci:

$$\begin{aligned}
 f[x_0] &= y_0, \\
 f[x_0, x_1] &= \frac{f[x_1] - f[x_0]}{x_1 - x_0}, \\
 f[x_0, x_1, x_2] &= \frac{f[x_1, x_2] - f[x_0, x_1]}{x_2 - x_0}, \\
 f[x_0, x_1, x_2, x_3] &= \frac{f[x_1, x_2, x_3] - f[x_0, x_1, x_2]}{x_3 - x_0}, \\
 &\vdots \\
 f[x_0, x_1, \dots, x_k] &= \frac{f[x_1, x_2, \dots, x_k] - f[x_0, x_1, \dots, x_{k-1}]}{x_k - x_0}.
 \end{aligned}$$

Vytvoříme trojúhelníkovou tabulku $(k+1) \times (k+1)$ (viz tabulku 2.1), ve které budeme zleva po sloupcích vyčíslovat hodnoty podílových diferencí. Hodnoty na diagonále pak budou odpovídat koeficientům a_i , $i \in \{0, 1, \dots, k\}$.

| | | | | | |
|----------|----------|-------------------|----------------------------|----------|---------------------------|
| x_0 | $f[x_0]$ | | | | |
| x_1 | $f[x_1]$ | $f[x_0, x_1]$ | | | |
| x_2 | $f[x_2]$ | $f[x_1, x_2]$ | $f[x_0, x_1, x_2]$ | | |
| \vdots | \vdots | \vdots | \vdots | \ddots | |
| x_k | $f[x_k]$ | $f[x_{k-1}, x_k]$ | $f[x_{k-2}, x_{k-1}, x_k]$ | \dots | $f[x_0, x_1, \dots, x_k]$ |

Tabulka 2.1: Tabulka podílových diferencí

Sestavení kompletní tabulky vyžaduje $O(k^2)$ FLOP.

Zmíňme ještě detaily o flexibilitě. Při přidávání nového vzorku (x_{k+1}, y_{k+1}) k souboru již zpracovaných bodů zvýšíme rozměr tabulky o 1. Nový řádek naplníme hodnotami podílových diferencí, na vzniklém diagonálním prvku pak získáme koeficient a_{k+1} . K původnímu interpolačnímu polynomu pak jen přičteme člen $a_{k+1}(x-x_0)(x-x_1)\dots(x-x_k)$. Složitost sestavení nového polynomu použitím vnořeného tvaru (2.6) pak činí $O(k)$ FLOP.

2.3 Existence a jednoznačnost interpolačního polynomu

Ukažme nyní, že polynomiální interpolant na zvoleném souboru dat vždy existuje a je unikátní.

Existence interpolantu plyne z platnosti vztahu (2.1) jak u Lagrangeovy metody, tak u Newtonovy metody. Umíme pomocí nich sestavit polynom vyhovující interpolačním podmínkám (2.1), proto musí existovat.

Nyní předpokládejme, že existují dva různé interpolační polynomy $p, q \in \mathbb{P}_n$ pro stejná vstupní data. Jelikož prokládáme $n+1$ bodů, tak funkce

$$r(x) = p(x) - q(x)$$

musí mít alespoň $n + 1$ kořenů, a to v bodech x_i , protože platí:

$$p(x_i) = q(x_i), \quad i \in \{0, 1, \dots, n\}.$$

Funkce r jistě náleží do \mathbb{P}_n . Podle základní věty algebry ovšem identicky nulová funkce z \mathbb{P}_n má právě n kořenů (včetně násobností). Proto r musí být identicky nulová, tj. pro všechna $x \in \mathbb{R}$ platí:

$$\begin{aligned} 0 &= p(x) - q(x), \\ p(x) &= q(x). \end{aligned}$$

Polynomy p a q tedy nemohou být různé. Ukázali jsme si tedy, že interpolační polynom existuje jednoznačně.

2.4 Odhad chyby interpolace

Následující věta nám popíše chování chyby, které se dopouštíme při interpolování funkce polynomem.

Věta 2.4.1. *Mějme funkci $f \in \mathbb{C}^{n+1}(\langle a, b \rangle)$. Mějme polynom $p \in \mathbb{P}_n$ interpolující f v bodech (x_i, y_i) , $i \in \{0, 1, \dots, n\}$, kde všechna $x_i \in \langle a, b \rangle$ jsou navzájem různá. Definujme chybovou funkci $e = f - p$. Pak pro všechna $x \in \langle a, b \rangle$ existuje číslo $\xi \in \langle a, b \rangle$ takové, že*

$$e(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!} \prod_{i=0}^n (x - x_i). \quad (2.7)$$

Důkaz. Pro $x = x_i$ je tvrzení triviální, jelikož $e(x_i) = 0$. Zaměříme se tedy na případ, kdy $x \neq x_i$. Označme

$$\begin{aligned} w(t) &= \prod_{i=0}^n (t - x_i) \in \mathbb{P}_{n+1}, \\ \phi(t) &= f(t) - p(t) - \frac{f(x) - p(x)}{w(x)} w(t), \end{aligned}$$

kde $x \in \langle a, b \rangle$ je libovolné fixní číslo.

Zřejmě, $\phi \in \mathbb{C}^{n+1}(\langle a, b \rangle)$. Všimněme si, že dohromady $n + 2$ kořenů funkce ϕ jsou v bodech x_0, x_1, \dots, x_n a také v x . Derivací ϕ snížíme počet kořenů o jeden, což zjistíme, když na každé dva sousední kořeny aplikujeme Rolleovu větu. $(n + 1)$ -ní derivací ϕ vznikne funkce $\phi^{(n+1)}$ s jedním kořenem v $\langle a, b \rangle$. Tento kořen nyní označme symbolem ξ . Platí

$$\begin{aligned} \phi^{(n+1)}(t) &= \left(f(t) - p(t) - \frac{f(x) - p(x)}{w(x)} w(t) \right)^{(n+1)} \\ &= f^{(n+1)}(t) - p^{(n+1)}(t) - \frac{f(x) - p(x)}{w(x)} w^{(n+1)}(t) \\ &= f^{(n+1)}(t) - 0 - \frac{f(x) - p(x)}{w(x)} (n+1)!. \end{aligned}$$

Dosadíme ξ :

$$0 = f^{(n+1)}(\xi) - \frac{f(x) - p(x)}{w(x)}(n+1)!.$$

Osamostatněním $f(x) - p(x)$ dostaneme

$$\begin{aligned} f(x) - p(x) &= \frac{f^{(n+1)}(\xi)}{(n+1)!} w(x) \\ &= \frac{f^{(n+1)}(\xi)}{(n+1)!} \prod_{i=0}^n (x - x_i). \end{aligned}$$

□

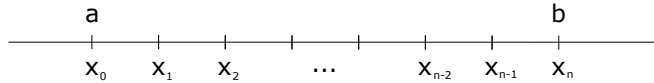
Vztah (2.7) nám pomůže při minimalizaci odchylky interpolantu od skutečných hodnot funkce f , jelikož nám říká, na čem tato chyba závisí. V další části se budeme věnovat jednomu z těchto faktorů, a to konkrétně optimálnímu rozložení uzlů x_i , které ovlivňuje hodnotu součinu

$$\prod_{i=0}^n (x - x_i). \quad (2.8)$$

2.4.1 Ekvidistantní uzly

Uzly s ekvidistantním rozložením na intervalu $\langle a, b \rangle$ rozumíme hodnoty x_i , jejichž vzdálenosti mezi každými dvěma sousedy jsou stejné:

$$x_i = a + ih, \quad h = \frac{b-a}{n}, \quad i \in \{0, 1, \dots, n\}.$$



Obrázek 2.4.1: Ekvidistantní uzly

Ukažme si nyní, jak toto rozložení ovlivňuje hodnotu součinu (2.8).

Věta 2.4.2. *Mějme na intervalu $\langle a, b \rangle$ zadáno $n+1$ ekvidistantních uzlů x_i s krokem $h = \frac{b-a}{n}$. Pak pro $x \in \langle a, b \rangle$ platí:*

$$\prod_{i=0}^n |x - x_i| \leq \frac{1}{4} h^{n+1} n!. \quad (2.9)$$

Důkaz. Omezme se na výběr bodu x z krajního intervalu $\langle x_0, x_1 \rangle$, jelikož není těžké si rozmyslet, že v něm je součin (2.8) maximální. Můžeme pak omezit hodnoty činitelů:

$$|x - x_2| \leq 2h, \quad |x - x_3| \leq 3h, \quad \dots, \quad |x - x_n| \leq nh.$$

Pro x_0 a x_1 se můžeme shora omezit na h , existuje však lepší odhad. Analýzou funkce

$$|(x - x_0)(x - x_1)|$$

lze lehce zjistit, že na intervalu $\langle x_0, x_1 \rangle$ nabývá maxima $\frac{(x_1 - x_0)^2}{4} = \frac{h^2}{4}$. Proto

$$|(x - x_0)(x - x_1)| \leq \frac{h^2}{4}.$$

Můžeme tedy psát

$$\prod_{i=0}^n |x - x_i| \leq \frac{h^2}{4} \cdot 2h \cdot 3h \cdot \dots \cdot nh \leq \frac{1}{4} h^{n+1} n!.$$

□

Dle (2.9) a (2.7) tedy můžeme v případě použití ekvidistantních uzlů shora omezit chybu na

$$|e(x)| \leq \frac{h^{n+1}}{4(n+1)} \|f^{(n+1)}\|_\infty, \quad x \in \langle a, b \rangle. \quad (2.10)$$

Jednou z nevýhod volby těchto uzlů je, že na krajích intervalu $\langle a, b \rangle$ můžeme pozorovat velká vychýlení interpolantu od reálných hodnot funkce f , jak již bylo naznačeno v důkazu. Abychom se vyhnuli tomuto jevu, můžeme použít následující rozložení uzlů, které chybu rovnoměrně rozprostře.

2.4.2 Čebyševovy uzly

Pro začátek se omezme na interval $\langle -1, 1 \rangle$. Čebyševovy uzly 1. druhu mají tvar

$$\bar{x}_i = \cos \frac{(2i+1)\pi}{2(n+1)}, \quad i \in \{0, 1, \dots, n\}. \quad (2.11)$$

Transformací uzlů do $\langle a, b \rangle$ získáme

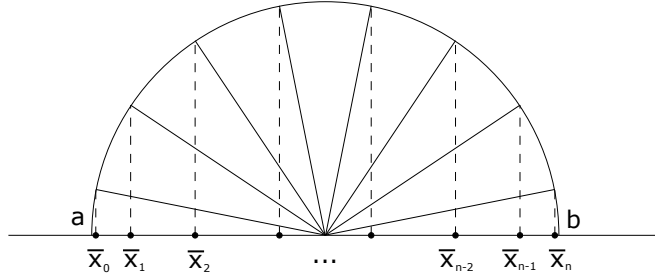
$$\bar{x}_i = \frac{a+b}{2} + \frac{a-b}{2} \cos \frac{(2i+1)\pi}{2(n+1)}, \quad i \in \{0, 1, \dots, n\}.$$

Na obrázku 2.4.2 můžeme pozorovat shlukování uzlů u krajních bodů intervalu $\langle a, b \rangle$. Tato vlastnost pomáhá řešit problém ekvidistantních uzlů - velké hodnoty chyby na krajích jsou kompenzovány častějším vzorkováním.

Čebyševovy uzly 2. druhu se liší jen v distribuci argumentu kosinu na intervalu $\langle 0, \pi \rangle$, obsahuje i jeho koncové body narozdíl od (2.11):

$$\bar{x}_i = \frac{a+b}{2} + \frac{a-b}{2} \cos \frac{i\pi}{n}, \quad i \in \{0, 1, \dots, n\}.$$

Pro oba tyto typy uzlů na intervalu $\langle -1, 1 \rangle$ platí následující věta, kterou si již uvedeme bez důkazu.



Obrázek 2.4.2: Čebyševovy uzly

Věta 2.4.3. *Mějme na intervalu $\langle -1, 1 \rangle$ zadáno $n + 1$ Čebyševových uzlů \bar{x}_i . Pak pro $x \in \langle -1, 1 \rangle$ platí:*

$$\prod_{i=0}^n |x - \bar{x}_i| \leq 2^{-n}.$$

Zároveň Čebyševovy uzly nejlépe minimalizují výše uvedený součin ze všech možných rozložení uzlů x_0, x_1, \dots, x_n v $\langle -1, 1 \rangle$:

$$\max_{x \in \langle -1, 1 \rangle} \prod_{i=0}^n |x - \bar{x}_i| = 2^{-n} \leq \max_{x \in \langle -1, 1 \rangle} \prod_{i=0}^n |x - x_i|. \quad (2.12)$$

Důkaz. Viz např. stranu 17 v [2]. □

Volbou těchto uzlů se tedy chyba interpolace na $\langle -1, 1 \rangle$ omezí na

$$|e(x)| \leq \frac{\|f^{(n+1)}\|_\infty}{2^n (n+1)!}, \quad x \in \langle -1, 1 \rangle. \quad (2.13)$$

2.5 Numerické experimenty

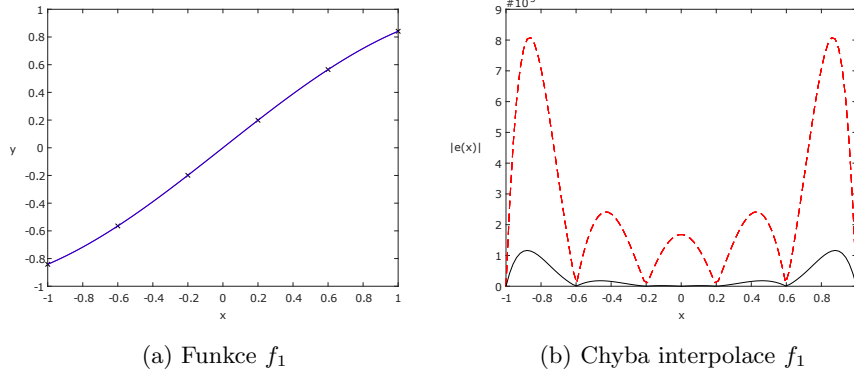
V této sekci budeme zkoumat chybu interpolace konkrétních rozložení uzlů. Každou vybranou funkci si navzorkujeme ve vybraných uzlech a budeme sledovat, jak je jí interpolace na základě těchto bodů věrná. Interpolanty jsme vyčíslovali ve 100 ekvidistantních bodech rozložených na intervalu $\langle -1, 1 \rangle$. Maximální naměřená absolutní chyba je počítána jako největší absolutní chyba přes všechny vyčíslované hodnoty interpolantu.

V levém grafu je modrou čarou uvedena funkce, kterou interpolujeme. Červeně je znázorněn interpolační polynom, který však nemusí být viditelný z důvodu malého řádu chyby (bude zřetelný v příkladech 2.5.3 a 2.5.4). Na pravé straně je plnou černou čarou vynesena naměřená chyba, červenou přerušovanou čarou je vymezen odhad chyby.

Příklad 2.5.1. $f_1(x) = \sin x$, 6 ekvidistantních uzlů, $x_0 = -1, x_5 = 1$

Maximová norma: $\|f_1^{(6)}\|_\infty = \|-\sin x\|_\infty = -\sin(-1) \approx 8,41 \cdot 10^{-1}$

Maximální teoretická chyba: $1,44 \cdot 10^{-4}$
 Maximální naměřená chyba: $1,63 \cdot 10^{-5}$



Obrázek 2.5.1: Grafy k příkladu 2.5.1

Na obrázku 2.5.1(b) můžeme vidět již zmíněný trend ekvidistantních uzlů, tj. na krajích intervalu je chyba největší. Nicméně jsme vyvázli s chybou poměrně malého řádu $1,63 \cdot 10^{-5}$, pouhým okem tedy nelze rozeznat funkci f_1 od jejího interpolantu.

Pomocí spočtené maximové normy $\|f_1^{(6)}\|_\infty$ a vztahu (2.10) dostaneme odhad chyby interpolace:

$$|e(x)| \leq \frac{\left(\frac{2}{5}\right)^6}{4 \cdot 6} \cdot 8,41 \cdot 10^{-1} \approx 1,44 \cdot 10^{-4}, \quad x \in \langle -1, 1 \rangle.$$

△

Příklad 2.5.2. $f_2(x) = e^x \sin x$, 9 Čebyševových uzlů 1. druhu

Maximová norma: $\|f_2^{(9)}\|_\infty = \|16e^x(\sin x + \cos x)\|_\infty = 16e(\sin 1 + \cos 1)$

Maximální teoretická chyba: $6,47 \cdot 10^{-7}$

Maximální naměřená chyba: $2,17 \cdot 10^{-7}$

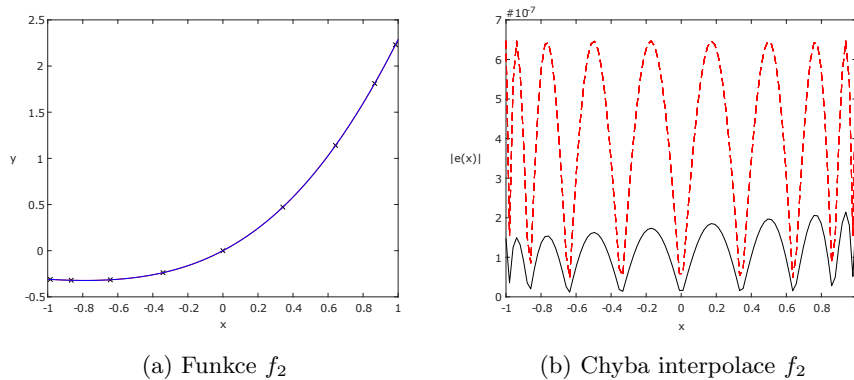
Obrázek 2.5.2(b) ilustruje rovnoměrné rozložení chyby intepolace pomocí Čebyševových uzlů.

Analýzou funkce $|f_2^{(9)}(x)| = 16|e^x(\sin x + \cos x)|$ zjistíme, že na intervalu $\langle -1, 1 \rangle$ je spojitá a nabývá jednoho lokálního extrému, a to $f_2^{(9)}(-\frac{\pi}{4}) = 0$. V tomto bodě se ale nachází minimum, my hledáme maximum. To pak musí být v jednom z krajních bodů intervalu, konkrétně v $x = 1$:

$$\|f_2^{(9)}\|_\infty = f_2^{(9)}(1) = 16e(\sin 1 + \cos 1) \approx 6,01 \cdot 10^1.$$

Dle předpisu (2.13) nalezneme odhad chyby:

$$|e(x)| \leq \frac{6,01 \cdot 10^1}{2^8 \cdot 9!} \approx 6,47 \cdot 10^{-7}, \quad x \in \langle -1, 1 \rangle.$$



Obrázek 2.5.2: Grafy k příkladu 2.5.2

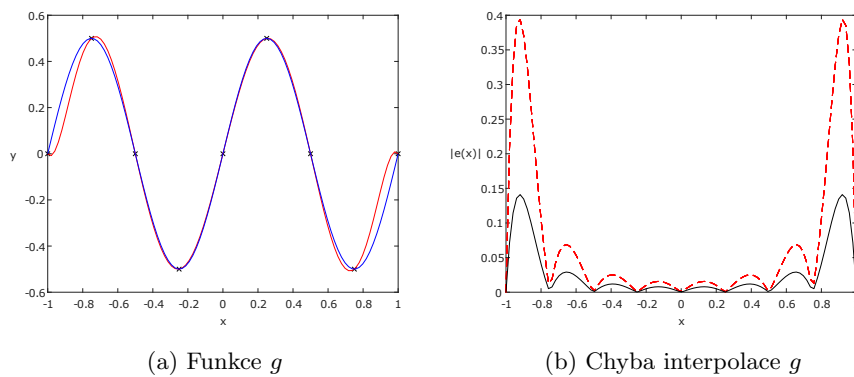
△

Příklad 2.5.3. $g(x) = \sin(\pi x) \cos(\pi x)$, 9 ekvidistntních uzlů, $x_0 = -1, x_8 = 1$

Maximová norma: $\|g^{(9)}\|_\infty = \|256\pi^9 \cos(2\pi x)\|_\infty = 256\pi^9$

Maximální teoretická chyba: $8,09 \cdot 10^{-1}$

Maximální naměřená chyba: $1,41 \cdot 10^{-1}$



Obrázek 2.5.3: Grafy k příkladu 2.5.3

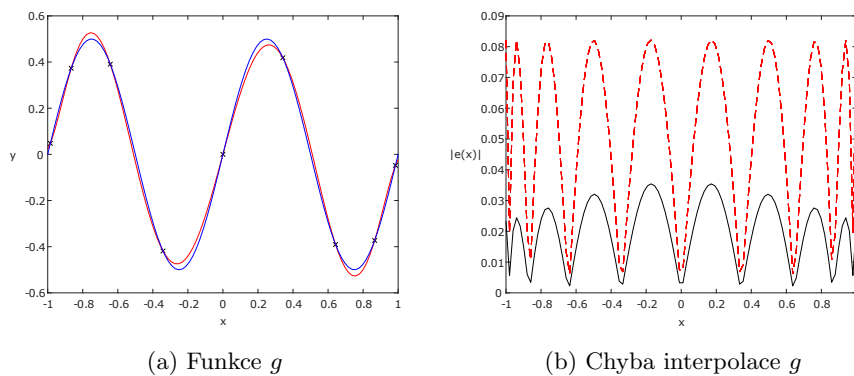
△

Příklad 2.5.4. $g(x) = \sin(\pi x) \cos(\pi x)$, 9 Čebyševových uzlů 1. druhu

Maximová norma: $\|g^{(9)}\|_\infty = \|256\pi^9 \cos(2\pi x)\|_\infty = 256\pi^9$

Maximální teoretická chyba: $8,21 \cdot 10^{-2}$

Maximální naměřená chyba: $3,54 \cdot 10^{-2}$



Obrázek 2.5.4: Grafy k příkladu 2.5.4

Pro příklady 2.5.3 a 2.5.4 byla vybrána stejná funkce g . Oproti prvním dvěma ukázkám však chyba interpolace o několik řádů vzrostla. Důvodem je vysoká hodnota maximové normy $(n+1)$ -ní, tedy deváté derivace funkce g . Jelikož se pohybujeme na intervalu $\langle -1, 1 \rangle$, lze usoudit, že

$$\|256\pi^9 \cos(2\pi x)\|_\infty = 256\pi^9 \approx 7,63 \cdot 10^6,$$

což např. pro Čebyševovy uzly dá dle (2.13) horní mez absolutní chyby

$$|e(x)| \leq \frac{7,63 \cdot 10^6}{2^8 \cdot 9!} \approx 8,21 \cdot 10^{-2}.$$

V neposlední řadě si můžeme také povšimnout, že s Čebyševovými uzly jsme dosáhli skoro o jeden řád menší chyby oproti ekvidistantním uzlům, což jsme očekávali, viz (2.12).

△

Kapitola 3

Barycentrická Lagrangeova interpolace

Vraťme se nyní zpět k Lagrangeově metodě. Oproti Newtonově přístupu postrádá na flexibilitě, tuto vlastnost lze ovšem získat jiným pohledem na Lagrangeovu interpolaci. Informace o tomto postupu budeme převážně čerpat z článku J.-P. Berruta a L. N. Trefethena [3].

3.1 Výhodnější formulace Lagrangeovy interpolace

Přeformulujme si vyjádření interpolantu (2.3). Nejprve si všimněme, že čitatel Lagrangeovy báze funkce L_i (2.2) se dá zapsat jako

$$\ell(x) = \prod_{j=0}^n (x - x_j)$$

vydělený členem $x - x_i$. Pokud poté definujeme tzv. barycentrické váhy w_i pomocí

$$w_i = \frac{1}{\prod_{\substack{j=0 \\ j \neq i}}^n (x_i - x_j)}, \quad i \in \{0, 1, \dots, n\}, \quad (3.1)$$

můžeme zapsat i -tou Lagrangeovu bázeovou funkci jako

$$L_i(x) = \frac{\ell(x)}{x - x_i} w_i, \quad i \in \{0, 1, \dots, n\}.$$

Lagrangeův interpolační polynom (2.3) pak má tvar

$$L(x) = \sum_{i=0}^n \frac{\ell(x)}{x - x_i} w_i y_i,$$

ve kterém můžeme ze sumy vytknout čitatele $\ell(x)$, jelikož nezávisí na sumační proměnné i . Dostáváme barycentrický tvar Lagrangeovy interpolace

$$L(x) = \ell(x) \sum_{i=0}^n \frac{w_i}{x - x_i} y_i. \quad (3.2)$$

Nyní se podívejme, jak jsme na tom se složitostí. Barycentrické váhy w_i dohromady zaberou $O(n^2)$ FLOP pro jejich vyčíslení, výraz $\ell(x)$ má náročnost vyčíslení $O(n)$ FLOP. Výpočet hodnoty interpolačního polynomu $L(x)$ při znalosti těchto objektů pak zabere $O(n)$ FLOP. Celkově tedy počítáme s náročností $O(n^2)$ FLOP.

Narozdíl od předchozího pohledu na Lagrangeovu interpolaci je nám nyní umožněno efektivně přidávat nové vzorky dat. S každým novým vzorkem (x_{n+1}, y_{n+1}) potřebujeme podělit každou i -tou váhu výrazem $x_i - x_{n+1}$, $i = 0, 1, \dots, n$, a vypočítat jednu novou váhu w_{n+1} . Na obě operace potřebujeme $O(n)$ FLOP. Celkově nám tedy přidání nového prvku zabere $O(n)$ FLOP.

Oproti Newtonově metodě získáváme navíc nezávislost barycentrických vah w_i , $i = 0, 1, \dots, n$, na hodnotách y_j ze vstupních dat (x_j, y_j) , $j = 0, 1, \dots, n$. Této nezávislosti lze například využít v situaci, kde máme dva soubory dat, kde se vzorky liší pouze v y_j - není nutné znovu napočítávat barycentrické váhy w_i .

3.2 Barycentrický tvar Lagrangeovy interpolace

Vztah (3.2) lze zapsat ještě elegantněji a výhodněji. Interpolujeme-li konstantní funkci 1, dostaneme

$$1 = \sum_{i=0}^n L_i(x) = \ell(x) \sum_{k=0}^n \frac{w_k}{x - x_k}.$$

Jelikož interpolant konstantní funkce je tatáž konstantní funkce, můžeme jedničkou podělit původní předpis (3.2). Tímto se vykrátí společný faktor $\ell(x)$ a získáme druhý způsob zápisu barycentrického tvaru Lagrangeovy interpolace:

$$L(x) = \frac{\sum_{k=0}^n \frac{w_k}{x - x_k} y_k}{\sum_{k=0}^n \frac{w_k}{x - x_k}}. \quad (3.3)$$

V literatuře se tento vztah uvádí jako „pravá“ barycentrická formulace [3]. Stejně jako (3.2), tak i (3.3) má stejné výpočetní náročnosti. Navíc ale vyjádření ve tvaru zlomku umožní krácení společných faktorů nezávislých na k v čitateli a jmenovateli, čehož využijeme v následující sekci.

3.3 Barycentrické váhy

Při použití vybraných rozložení uzlů lze váhy w_i vyjádřit explicitně. Ukažme pro již uvedené distribuce uzlů, jak tyto váhy vypadají.

3.3.1 Ekvidistantní uzly

Věta 3.3.1. *Barycentrické váhy w_i při použití ekvidistantní distribuce $n + 1$ uzlů lze zjednodušeně vyjádřit pomocí*

$$w_i = (-1)^i \binom{n}{i} \quad (3.4)$$

pro $i \in \{0, 1, \dots, n\}$.

Důkaz. Za účelem přehlednosti se zaměříme na převrácenou hodnotu váhy. Každé dva ekvidistantní body x_i a x_j můžeme nahradit součty $a + ih$ a $a + jh$, kde h je vzdálenost mezi dvěma libovolnými sousedními uzly. Pro převrácenou váhu w_i pak dle vztahu (3.1) platí:

$$\frac{1}{w_i} = \prod_{\substack{j=0 \\ j \neq i}}^n (a + ih - a - jh) = \prod_{\substack{j=0 \\ j \neq i}}^n h(i - j) = h^n \prod_{\substack{j=0 \\ j \neq i}}^n (i - j).$$

Člen h^n se ve vzorci (3.3) po vytknutí (jeho převrácené hodnoty) ze sumy vykrátí, lze jej tedy vynechat. Můžeme dále psát:

$$\prod_{\substack{j=0 \\ j \neq i}}^n (i - j) = \prod_{j=0}^{i-1} (i - j) \cdot \prod_{j=i+1}^n (i - j).$$

Není těžké si uvědomit, že

$$\prod_{j=0}^{i-1} (i - j) = i!, \quad \prod_{j=i+1}^n (i - j) = (-1)^{n-i} (n - i)!.$$

Pokračováním ve vyjádření získáme

$$\frac{1}{w_i} = i! \cdot (-1)^{n-i} (n - i)! = (-1)^{n-i} \cdot \frac{n!}{\binom{n}{i}} = (-1)^n \cdot (-1)^{-i} \cdot \frac{n!}{\binom{n}{i}}.$$

Členy $(-1)^n$ a $n!$ ve vztahu (3.3) taktéž vykrátí. Nakonec nám tedy zbyde

$$\frac{1}{w_i} = (-1)^{-i} \frac{1}{\binom{n}{i}} \Rightarrow w_i = (-1)^i \binom{n}{i}.$$

□

Váhy je z hlediska počtu FLOP výhodné počítat pomocí kombinačních čísel dle principu sčítání v Pascalově trojúhelníku tzn. musíme si sestavit Pascalův trojúhelník až po n -tý řádek. Počet FLOP tohoto postupu patří řádově do stejné složitosti $O(n^2)$ jako u vyčíslení pomocí faktoriálů. Při sčítání je ovšem koeficient u n^2 pouze čtvrtinový oproti použití vzorečku s faktoriály.

3.3.2 Čebyševovy uzly

Věta 3.3.2. *Barycentrické váhy w_i při použití Čebyševových uzlů 1. druhu mohou být vyjádřeny jako*

$$w_i = (-1)^i \sin \frac{(2i+1)\pi}{2n+2}$$

pro $i \in \{0, 1, \dots, n\}$. Pro Čebyševovy uzly 2. druhu lze váhy vyjádřit jako

$$w_i = (-1)^i \delta_i, \quad \delta_i = \begin{cases} 1/2, & i = 0 \vee i = n, \\ 1, & \text{jinak,} \end{cases}$$

pro $i \in \{0, 1, \dots, n\}$.

Důkaz. Viz např. stranu 506 v [3]. □

3.4 Numerické experimenty

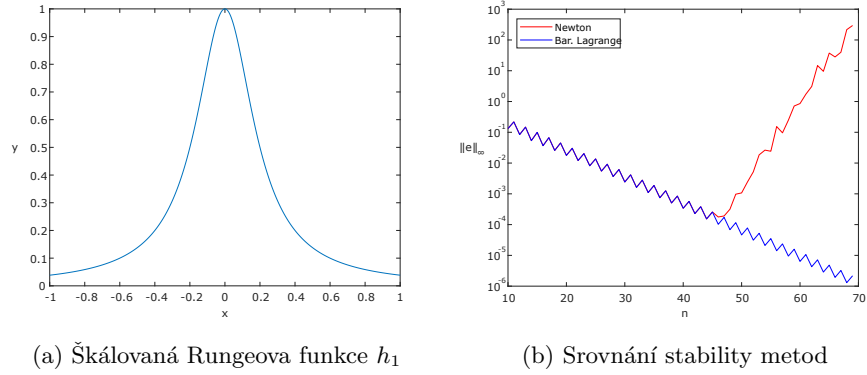
V této části nyní srovnáme dvě metody, a to barycentrickou Lagrangeovu metodu a Newtonovu metodu. Bude nás zajímat hlavně stabilita. Interpolace bude tvořena pomocí vzorků sledovaných v Čebyševových uzlech 2. druhu na intervalu $(-1, 1)$. Počet vykreslovacích bodů, v nichž na sledovaném úseku počítáme naměřené chyby interpolace, je 1000.

Grafy na obrázcích 3.4.1(a) a 3.4.2(a) znázorňují funkci, která je interpolována. V grafech 3.4.1(b) a 3.4.2(b) je sledována maximální naměřená absolutní chyba, která je závislá na počtu vzorků. Pro přehlednější zobrazení chyby použijeme logaritmickou škálu svislé osy. Červenou barvou je zaznačena chyba Newtonovy metody, modrá čára pak odpovídá chybě barycentrické Lagrangeovy metody.

Příklad 3.4.1. *Pro první experiment jsme vybrali škálovanou Rungeovu funkci*

$$h_1(x) = \frac{1}{1 + 25x^2}.$$

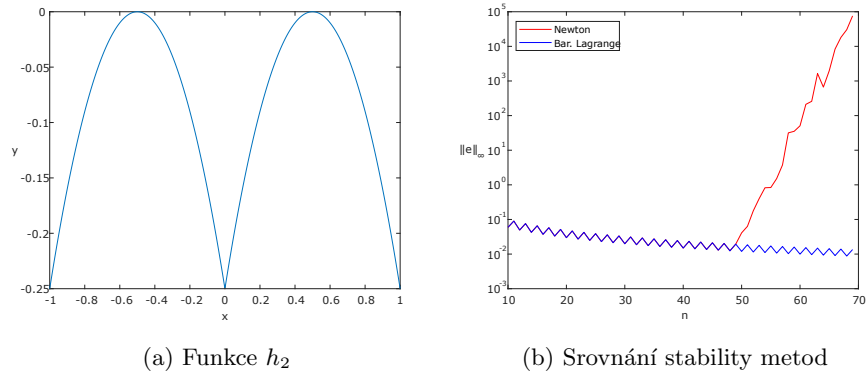
V grafu 3.4.1(b) můžeme sledovat, že do 45 vzorků se interpolace chová tak, jak očekáváme, interpolant je jednoznačný, tudíž zachovává i maximální chybu. Od tohoto momentu se ale začne navyšovat chyba Newtonovy metody. To je způsobeno velkým řádem koeficientů a_k , které mají jako následek nepřesnost výpočtu kvůli zaokrouhlovacím chybám. Newtonova metoda se tedy jeví jako výrazně méně stabilní.



Obrázek 3.4.1: Grafy k příkladu 3.4.1

△

Příklad 3.4.2. Podívejme se ještě na vývoj maximální chyby funkce $h_2(x) = |x| - (x + 0,5)^2 + x$. Zde se projevuje podobný jev jako v předchozím příkladě,



Obrázek 3.4.2: Grafy k příkladu 3.4.2

Newtonova metoda začíná od 50 vzorků kolabovat. Chvilu, odkdy začne vznikat nepřesnost, se odvíjí od samotných hodnot souřadnic vzorků, ty se totiž podílejí na sestavení koeficientů a_k . Podotkněme také, že chyba interpolace klesá velmi pomalu, což může být přičiněno samotné funkcí h_2 , protože není diferencovatelná v bodě 0 a existuje zde tzv. „hrot“. Všimněme si, že pokud bychom odstranili z předpisu funkce h_2 absolutní hodnotu, dostaneme polynom z \mathbb{P}_2 , což by znamenalo nulovost chyby při použití tří a více vzorků.

△

Kapitola 4

Racionální interpolace

Podívejme se nyní na zcela jiný přístup k interpolování vstupních dat. Může se stát, že nebudeme chtít interpolant ve tvaru polynomiální funkce, ale v podobě zlomkového výrazu. S tímto přístupem avšak musíme pracovat obezřetněji, ne vždy totiž může takovýto předpis existovat. V této kapitole budeme čerpat především z knihy J. Stoera a R. Bulirsche [4].

4.1 Úvod k racionální interpolaci

Mějme jako vstupní data dány body (x_i, y_i) , $i = 0, 1, 2, \dots$, kde všechna x_i jsou navzájem různá. Racionální interpolací rozumíme nalezení předpisu funkce R ve tvaru zlomku, která obsahuje mnohočleny v čitateli i ve jmenovateli a splňuje interpolační podmínky:

$$R(x) = \frac{P(x)}{Q(x)} = \frac{a_0 + a_1x + \dots + a_\mu x^\mu}{b_0 + b_1x + \dots + b_\nu x^\nu}, \quad \mu, \nu \in \mathbb{N} \cup \{0\}, \quad (4.1)$$

$$R(x_i) = y_i, \quad i = 0, 1, 2, \dots \quad (4.2)$$

Úlohu nalezení R splňující podmínky (4.1) a (4.2) zkráceně označme pomocí symbolu (A). Vidíme, že $P \in \mathbb{P}_\mu$ a $Q \in \mathbb{P}_\nu$.¹ Celý předpis je tedy určen $\mu + \nu + 2$ koeficienty. Jelikož v tomto zlomku můžeme vždy krátit nenulovým faktorem $\rho \in \mathbb{R}$, snižuje se počet neznámých hodnot na $\mu + \nu + 1$. Ty získáme z interpolačních podmínek, jejichž rozepsáním dostaneme

$$P(x_i) - y_i Q(x_i) = 0, \quad i = 0, 1, \dots, \mu + \nu, \quad (4.3)$$

což lze dále vyjádřit pomocí koeficientů a_j, b_k , $j = 0, 1, \dots, \mu$, $k = 0, 1, \dots, \nu$ takto:

$$a_0 + a_1x_i + \dots + a_\mu x_i^\mu - y_i(b_0 + b_1x_i + \dots + b_\nu x_i^\nu) = 0, \quad i = 0, 1, \dots, \mu + \nu. \quad (S)$$

¹Všimněme si, že pokud $\nu = 0$, mluvíme o klasické polynomiální interpolaci.

SLR (S) můžeme poté zapsat a řešit maticově:

$$\begin{pmatrix} 1 & x_0 & \dots & x_0^\mu & -y_0 & -y_0 x_0 & \dots & -y_0 x_0^\nu \\ 1 & x_1 & \dots & x_1^\mu & -y_1 & -y_1 x_1 & \dots & -y_1 x_1^\nu \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{\mu+\nu} & \dots & x_{\mu+\nu}^\mu & -y_{\mu+\nu} & -y_{\mu+\nu} x_{\mu+\nu} & \dots & -y_{\mu+\nu} x_{\mu+\nu}^\nu \end{pmatrix} \begin{pmatrix} a_0 \\ \vdots \\ a_\mu \\ b_0 \\ \vdots \\ b_\nu \end{pmatrix} = \vec{0}.$$

Uveďme nyní úmluvu, která nám umožní se v textu stručněji vyjadřovat. Řekneme, že řešení SLR (S) řeší úlohu (A), pokud řešení $a_0, \dots, a_\mu, b_0, \dots, b_\nu$ úlohy (S) definuje koeficienty funkce R , jež řeší (A). Dále řekneme-li, že funkce R řeší (S), myslíme tím, že příslušné koeficienty R řeší (S).

Nahrazení úlohy (A) úlohou (S) potřebuje jistou dávku opatrnosti, může totiž nastat situace, kdy se dopátráme k řešení (S), které nebude řešit (A). Mimo to mohou existovat body nespojitosti na místech, které by mohly vadit, např. mezi vzorky. Na následujících příkladech si tyto situace popíšeme.

Příklad 4.1.1. Pro vzorky $(0, 2), (1, 1), (2, 1)$ a volbu $\mu = 0$ a $\nu = 2$ dostáváme homogenní SLR

$$\begin{aligned} a_0 - 2b_0 &= 0, \\ a_0 - (b_0 + b_1 + b_2) &= 0, \\ a_0 - (b_0 + 2b_1 + 4b_2) &= 0. \end{aligned}$$

Jedno z možných řešení soustavy sestává z koeficientů $a_0 = 4$, $b_0 = 2$, $b_1 = 3$, $b_2 = -1$. Dostáváme pak racionální funkci, která splňuje podmínku (4.2)

$$R(x) = \frac{4}{2 + 3x - x^2}.$$

△

Příklad 4.1.2. Pro vzorky $(0, 2), (1, 1), (2, 1)$ a volbu $\mu = 1$ a $\nu = 1$ dostáváme homogenní SLR

$$\begin{aligned} a_0 - 2b_0 &= 0, \\ a_0 + a_1 - (b_0 + b_1) &= 0, \\ a_0 + 2a_1 - (b_0 + 2b_1) &= 0. \end{aligned}$$

Vyřešením soustavy můžeme získat koeficienty $a_0 = 0$, $a_1 = 1$, $b_0 = 0$, $b_1 = 1$. Racionální funkce má tvar

$$R(x) = \frac{x}{x} = 1.$$

Při kontrole funkce R zjišťujeme, že vstupní bod $(0, 2)$ není interpolován.

△

Příklad 4.1.3. Pro vzorky $(-1, -1), (-\frac{1}{3}, -\frac{1}{243}), (\frac{1}{3}, \frac{1}{243}), (1, 1)$ a volbu $\mu = 2$ a $\nu = 1$ dostáváme homogenní SLR

$$\begin{aligned} a_0 - a_1 + a_2 + (b_0 - b_1) &= 0, \\ a_0 - \frac{1}{3}a_1 + \frac{1}{9}a_2 + \frac{1}{243}(b_0 - \frac{1}{3}b_1) &= 0, \\ a_0 + \frac{1}{3}a_1 + \frac{1}{9}a_2 - \frac{1}{243}(b_0 + \frac{1}{3}b_1) &= 0, \\ a_0 + a_1 + a_2 - (b_0 + b_1) &= 0. \end{aligned}$$

Jedno z řešení má koeficienty $a_0 = -10, a_1 = 0, a_2 = 91, b_0 = 0, b_1 = 81$. Racionální funkci zapíšeme jako

$$R(x) = \frac{-10 + 91x^2}{81x}.$$

Hned vidíme, že R splňuje (4.2), ale je nespojitá v bodě 0, který leží v intervalu $\langle -1, 1 \rangle$, na němž data interpolujeme. Této nespojistosti se chceme jistě vyhnout.

△

4.2 Existence a jednoznačnost racionální interpolace

Než se dostaneme k vysvětlení, za jakých podmínek řešení (S) řeší (A), uvedme si nejprve tvrzení k existenci a jednotnou reprezentaci hledané funkce.

Věta 4.2.1. Homogenní SLR (S) má vždy netriviální řešení tvaru

$$R(x) = \frac{P(x)}{Q(x)}, \quad Q \not\equiv 0.$$

Tj. všechna netriviální řešení (S) jsou racionálními funkcemi.

Důkaz. SLR (S) obsahuje $\mu + \nu + 1$ rovnic a $\mu + \nu + 2$ neznámých. Homogenní soustavy s větším počtem neznámých než rovnic mají vždy netriviální řešení. Platnost $Q \not\equiv 0$ dále ukážeme sporem.

Předpokládejme, že $Q \equiv 0$. Vztah (4.3) pak říká, že P má $\mu + \nu + 1$ kořenů v bodech $x_i, i = 0, 1, \dots, \mu + \nu$. Stupeň polynomu P je ale nejvýše μ . Jelikož základní věta algebry říká, že identicky nenulová funkce z \mathbb{P}_μ má právě μ kořenů (včetně násobností), musí platit, že $P \equiv 0$. Zde vzniká kontradikce s netriviálností řešení (S). □

Nyní víme, že existuje řešení úlohy (S) ve tvaru racionální funkce. Pro pochopení „unikátnosti“ řešení si musíme uvědomit, že jeden racionální výraz může být napsán více způsoby. Jelikož tvary téhož zlomku lze získat krácením nebo rozšiřováním, je k věci tyto zápisy seskupit pomocí vztahu ekvivalence. Ekvivalentní racionální výrazy pak reprezentují stejnou racionální funkci.

Definice 4.2.1. Mějme dva racionální výrazy R_1, R_2 ve tvaru

$$R_1(x) = \frac{P_1(x)}{Q_1(x)}, \quad R_2(x) = \frac{P_2(x)}{Q_2(x)}, \quad Q_1 \not\equiv 0, \quad Q_2 \not\equiv 0.$$

Řekneme, že R_1 a R_2 jsou ekvivalentní, pokud

$$P_1(x)Q_2(x) = P_2(x)Q_1(x).$$

Píšeme $R_1 \sim R_2$. Jinak řečeno, R_1 a R_2 reprezentují tutéž racionální funkci.

Věta 4.2.2. Mějme dvě netriviální řešení R_1, R_2 soustavy (S). Pak platí:

$$\frac{P_1(x)}{Q_1(x)} = R_1(x) \sim R_2(x) = \frac{P_2(x)}{Q_2(x)}.$$

Důkaz. Definujme mnohočlen $P(x) = P_1(x)Q_2(x) - P_2(x)Q_1(x)$. Pak P má $\mu + \nu + 1$ různých kořenů, protože

$$\begin{aligned} P(x_i) &= P_1(x_i)Q_2(x_i) - P_2(x_i)Q_1(x_i) \\ &= y_i Q_1(x_i)Q_2(x_i) - y_i Q_2(x_i)Q_1(x_i) \\ &= 0, \quad i = 0, 1, \dots, \mu + \nu. \end{aligned}$$

Dle základní věty algebry musí pak být $P \in \mathbb{P}_{\mu+\nu}$ identicky nulová funkce. Z toho vyplývá, že $R_1 \sim R_2$. \square

Věty 4.2.1 a 4.2.2 společně nesou informaci, že každá úloha (A) vždy vede k jednoznačné racionální funkci, která je reprezentována jakýmkoli racionálním výrazem R , jehož koeficienty řeší SLR (S). Tato funkce pak buď splňuje interpolační podmínky (4.2), a tedy její koeficienty řeší (A), nebo (A) nemá řešení. O řešitelnosti (A) budeme mluvit v sekci níže.

4.3 Řešitelnost úlohy racionální interpolace

Pro začátek si definujme vlastnost racionálního výrazu, kterou v této části budeme potřebovat.

Definice 4.3.1. Racionální výraz je nesoudělný, jestliže čitatel i jmenovatel není dělitelný tímtož mnohočlenem různým od ± 1 .

Příklad 4.3.1. Racionální výrazy

$$\frac{9x}{3(x-1)}, \quad \frac{4(x-1)(x-2)}{5(x-2)(x+10)}, \quad \frac{10x(x+1)(x+2)(x+3)}{20(x+2)(x+3)}$$

nejsou nesoudělné. Zlomky lze v uvedeném pořadí krátit pomocí mnohočlenů

$$3, \quad (x-2), \quad 10(x+2)(x+3).$$

Po vykrácení uvedenými mnohočleny získáme nesoudělné racionální výrazy

$$\frac{3x}{x-1}, \quad \frac{4(x-1)}{5(x+10)}, \quad \frac{x(x+1)}{2}.$$

△

Jak lze vidět v příkladu 4.1.2, v řešení může existovat tzv. nedostupný bod, který byl vynechán. Jeho přítomnost je samozřejmě nežádoucí, podmínka (4.2) není splněna. Pokud tedy nedostupné body neexistují, úloha (A) je řešitelná díky větám 4.2.1 a 4.2.2. Shrňme tyto informace do pozorování níže.

Pozorování 4.3.1. Mějme řešení $R(x) = \frac{P(x)}{Q(x)}$ úlohy (S). Pro $i = 0, 1, \dots, \mu + \nu$ rozlišíme dvě situace:

- 1) $Q(x_i) \neq 0 \Rightarrow R(x_i) = y_i$, bod (x_i, y_i) je interpolován,
- 2) $Q(x_i) = 0 \Rightarrow (x_i, y_i)$ může být nedostupný.

V případě 2) musí jistě platit $P(x_i) = 0$ dle (4.3). To ale znamená, že $P(x)$ a $Q(x)$ obsahují společný faktor $(x - x_i)$, jinak řečeno nejsou nesoudělné.

Nyní konečně použijeme nesoudělnost z definice 4.3.1:

Věta 4.3.1. Mějme nesoudělné netriviální řešení R soustavy (S). Pak pro R neexistují nedostupné body a R řeší úlohu (A).

4.4 Metoda inverzních diferencí

Vyčíslování koeficientů a_i, b_j přímo pomocí matice SLR (S) není v praxi používáno z důvodu její špatné podmíněnosti. Proto se využívají alternativní postupy řešení tohoto problému. Uveďme si jeden z těchto možných postupů, a to metodu inverzních diferencí.

Jak již z názvů vyplývá, budeme využívat tzv. inverzní difference ϕ definované jako

$$\begin{aligned} \phi(x_0) &= y_0, \\ \phi(x_0, x_1) &= \frac{x_0 - x_1}{\phi(x_0) - \phi(x_1)}, \\ \phi(x_0, x_1, x_2) &= \frac{x_1 - x_2}{\phi(x_0, x_1) - \phi(x_0, x_2)}, \\ &\vdots \\ \phi(x_0, \dots, x_{k-2}, x_{k-1}, x_k) &= \frac{x_{k-1} - x_k}{\phi(x_0, \dots, x_{k-2}, x_{k-1}) - \phi(x_0, \dots, x_{k-2}, x_k)}. \end{aligned}$$

Funkce ϕ nám bude užitečná v následujícím postupu. Nechť $P^\mu \in \mathbb{P}_\mu$ a $Q^\nu \in \mathbb{P}_\nu$. Budeme hledat řešení $R^{n,n}$ ve tvaru

$$R^{n,n}(x) = \frac{P^n(x)}{Q^n(x)},$$

pro které platí podmínky (4.2), tedy

$$R^{n,n}(x_i) = y_i, \quad i = 0, 1, \dots, 2n.$$

Předpis pro R budeme postupně konkretizovat pomocí vstupních dat tak, aby interpolační podmínky (4.2) byly zachovány. Pro první vzorek (x_0, y_0) provedeme tuto úpravu:

$$\begin{aligned} \frac{P^n(x)}{Q^n(x)} &= y_0 + \frac{P^n(x)}{Q^n(x)} - y_0 \\ &= y_0 + \frac{P^n(x) - y_0 Q^n(x)}{Q^n(x)}. \end{aligned}$$

Čitatel zlomku má určitě kořen v bodě x_0 dle interpolačních podmínek (4.3). Můžeme tedy vytknout faktor $(x - x_0)$, a proto:

$$\begin{aligned} \frac{P^n(x)}{Q^n(x)} &= y_0 + (x - x_0) \frac{P^{n-1}(x)}{Q^n(x)} \\ &= y_0 + \frac{x - x_0}{Q^n(x)/P^{n-1}(x)}. \end{aligned}$$

Aby byly splněny interpolační podmínky (4.3), musí platit

$$\frac{Q^n(x_i)}{P^{n-1}(x_i)} = \frac{x_i - x_0}{y_i - y_0} = \phi(x_0, x_i), \quad i = 1, 2, \dots, 2n. \quad (4.4)$$

Nyní pomocí dalšího vzorku (x_1, y_1) přepíšeme $\frac{Q^n(x)}{P^{n-1}(x)}$:

$$\begin{aligned} \frac{Q^n(x)}{P^{n-1}(x)} &= \phi(x_0, x_1) + \frac{Q^n(x)}{P^{n-1}(x)} - \phi(x_0, x_1) \\ &= \phi(x_0, x_1) + \frac{Q^n(x) - \phi(x_0, x_1)P^{n-1}(x)}{P^{n-1}(x)}. \end{aligned}$$

Pomocí nově získaných podmínek (4.4) můžeme z čitatele zlomku vytknout faktor $(x - x_1)$:

$$\begin{aligned} \frac{Q^n(x)}{P^{n-1}(x)} &= \phi(x_0, x_1) + (x - x_1) \frac{Q^{n-1}(x)}{P^{n-1}(x)} \\ &= \phi(x_0, x_1) + \frac{x - x_1}{P^{n-1}(x)/Q^{n-1}(x)} \end{aligned}$$

a musí platit:

$$\frac{P^{n-1}(x_i)}{Q^{n-1}(x_i)} = \frac{x_i - x_1}{\phi(x_0, x_i) - \phi(x_0, x_1)} = \phi(x_0, x_1, x_i), \quad i = 2, 3, \dots, 2n.$$

Řešení $R^{n,n}$ pak můžeme zapsat jako

$$\begin{aligned} R^{n,n}(x) &= \frac{P^n(x)}{Q^n(x)} \\ &= y_0 + \frac{x - x_0}{Q^n(x)/P^{n-1}(x)} \\ &= y_0 + \frac{x - x_0}{\phi(x_0, x_1) + \frac{x - x_1}{P^{n-1}(x)/Q^{n-1}(x)}}. \end{aligned}$$

V algoritmu dále pomocí dalších vzorků snižujeme stupně polynomů P^{n-1} a Q^{n-1} v uvedeném pořadí, až se dostaneme k nulovým stupňům. Postupně se takto budeme zanořovat do zlomku. Na konci algoritmu dostaneme řešení ve tvaru

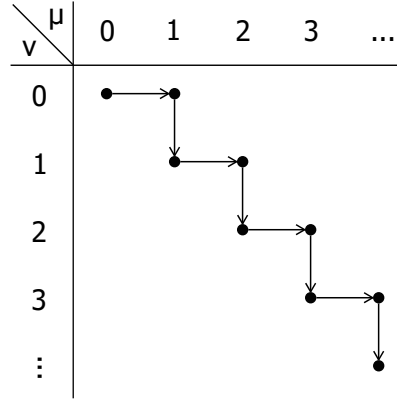
$$\begin{aligned} R^{n,n}(x) = y_0 + & \frac{x - x_0}{\phi(x_0, x_1) + \frac{x - x_1}{\phi(x_0, x_1, x_2) + \frac{x - x_2}{\phi(x_0, x_1, x_2, x_3) + \\ & \vdots \\ & + \frac{x - x_{2n-1}}{\phi(x_0, \dots, x_{2n})}}. \end{aligned}$$

Tento pokračující zlomek lze „předčasně ukončit“ a získáme tak výrazy $R^{\mu,\mu}$ a $R^{\mu+1,\mu}$, $\mu = 0, 1, \dots, n-1$. Každý následující výraz pak interpoluje o jeden vstupní bod více než předchozí:

$$\begin{aligned} R^{0,0}(x) &= y_0 \\ R^{1,0}(x) &= y_0 + \frac{x - x_0}{\phi(x_0, x_1)} \\ R^{1,1}(x) &= y_0 + \frac{x - x_0}{\phi(x_0, x_1) + \frac{x - x_1}{\phi(x_0, x_1, x_2)}} \\ &\vdots \end{aligned}$$

Tímto postupem alternujeme mezi navyšováním stupně čitatele a jmenovatele, viz obrázek 4.4.1.

Pro výpočet je výhodné si sestavit trojúhelníkovou tabulku inverzních diferencí (viz tabulku 4.1), která obsahuje všechny důležité napočtené hodnoty na posunuté hlavní diagonále.

Obrázek 4.4.1: Alternování inkrementace stupňů μ a ν

| i | x | y | $\phi(x_0, x_i)$ | $\phi(x_0, x_1, x_i)$ | \dots | $\phi(x_0, x_1, \dots, x_i)$ |
|----------|----------|----------|------------------|-----------------------|----------|------------------------------|
| 0 | x_0 | y_0 | | | | |
| 1 | x_1 | y_1 | $\phi(x_0, x_1)$ | | | |
| 2 | x_2 | y_2 | $\phi(x_0, x_2)$ | $\phi(x_0, x_1, x_2)$ | | |
| \vdots | \vdots | \vdots | \vdots | \vdots | \ddots | |
| k | x_k | y_k | $\phi(x_0, x_k)$ | $\phi(x_0, x_1, x_k)$ | \dots | $\phi(x_0, x_1, \dots, x_k)$ |

Tabulka 4.1: Tabulka inverzních diferencí

Podobně jako u Newtonovy metody získáváme stejnou výpočetní náročnost $O(n^2)$ FLOP pro vytvoření tabulky. Na sestavení interpolačního polynomu při její znalosti potřebujeme již jen $O(n)$ FLOP, stejnou náročnost má také přidání nového prvku a sestavení nového výrazu s tímto přidaným prvkem. Metoda je tedy flexibilní.

Jedna nevýhoda této metody je, že se ve jmenovateli inverzní difference může objevit nula, což znehodnotí celý výpočet. Pak je nutné zkusit postup opakovat a zvolit si jiné pořadí vzorků, ve kterém je do řešení začleníme.

4.5 Numerické experimenty

Předvedme si ukázkou výpočtu interpolantu pomocí metody inverzních diferencí.

Příklad 4.5.1. *Mějme zadána vstupní data*

$$\begin{array}{c|c|c|c|c|c} x_i & -1 & -1/2 & 0 & 1/2 & 1 \\ \hline y_i & -1/2 & 0 & 0 & 1/2 & 1/2 \end{array},$$

která odpovídají předpisu funkce $\frac{x}{2} + |x| - x^2$. Sestavíme tabulku inverzních diferencí (viz tabulku 4.1):

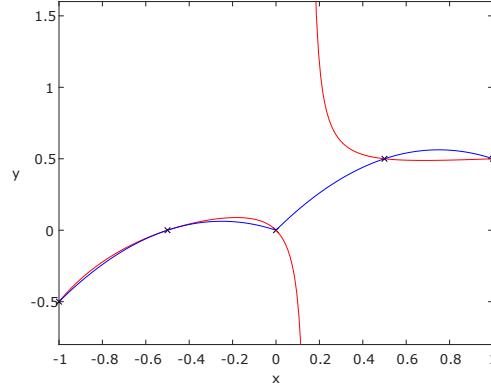
| i | x | y | $\phi(x_0, x_i)$ | $\phi(x_0, x_1, x_i)$ | $\phi(x_0, x_1, x_2, x_i)$ | $\phi(x_0, x_1, x_2, x_3, x_i)$ |
|-----|--------|--------|------------------|-----------------------|----------------------------|---------------------------------|
| 0 | -1 | $-1/2$ | | | | |
| 1 | $-1/2$ | 0 | 1 | | | |
| 2 | 0 | 0 | 2 | $1/2$ | | |
| 3 | $1/2$ | $1/2$ | $3/2$ | 2 | $1/3$ | |
| 4 | 1 | $1/2$ | 2 | $3/2$ | 1 | $3/4$ |

Použitím vyznačených koeficientů sestavíme řešení ve tvaru

$$R(x) = -\frac{1}{2} + \frac{x+1}{1 + \frac{1}{\frac{1}{2} + \frac{1}{\frac{1}{3} + \frac{x - \frac{1}{2}}{\frac{3}{4}}}}},$$

což po upravení do tvaru (4.1) vydá předpis

$$R(x) = \frac{3x(2x+1)}{2(4x^2+6x-1)}.$$



Obrázek 4.5.1: Funkce R , 5 vzorků

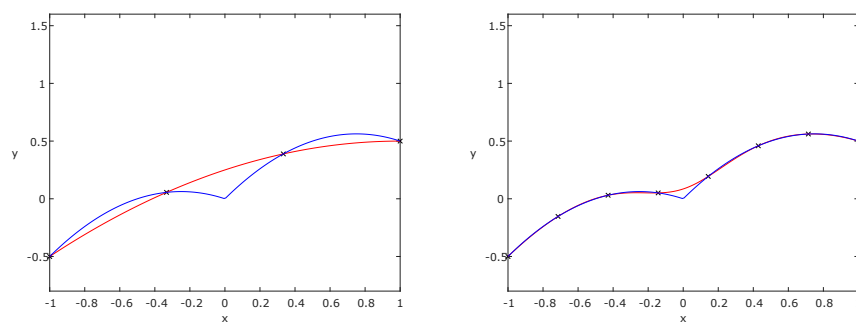
Na obrázku 4.5.1 je interpolovaná funkce označena modře. Červeně je znázorněn interpolant R procházející vstupními daty, která jsou označeny křížky. Kořeny jmenovatele dostáváme jako $\frac{-3 \pm \sqrt{13}}{4}$, z nichž jeden leží v intervalu $\langle -1, 1 \rangle$, na kterém interpolujeme. Tento bod nespojitosti v řešení nechceme. Můžeme se jej pokusit zbavit snížením či navýšením počtu vzorků. Například pro 4 vzorky v ekvidistantních uzlech ($x_0 = -1, x_3 = 1$) získáváme funkci

$$R_1(x) = \frac{-x^2 + 2x + 1}{4},$$

která je již na „interpolačním intervalu“ spojitá. Zvýšením počtu uzlů na šest či sedm se bodů nespojistosti nezbavíme. Pro 8 vzorků ($x_0 = -1, x_7 = 1$) dostáváme funkci

$$R_2(x) = \frac{-441x^4 + 392x^3 + 426x^2 + 88x + 15}{16(49x^2 + 11)},$$

která konečně splňuje požadavek na spojitost v intervalu $\langle -1, 1 \rangle$. Funkce R_1 a R_2 mají na obrázku 4.5.2 v grafech červenou barvu.



(a) Funkce R_1 , 4 vzorky

(b) Funkce R_2 , 8 vzorků

Obrázek 4.5.2: Interpolanty pro změněný počet vstupních dat

△

Kapitola 5

Barycentrická racionální interpolace

U barycentrického tvaru Lagrangeovy interpolace (3.3) si lze všimnout, že stejně jako racionální tvar (4.1) je zapsán ve tvaru zlomku. Není to náhodou a ukážeme si, že barycentrický tvar Lagrangeovy interpolace je jistým „speciálním případem“ racionálního tvaru interpolace. V textu níže čerpáme především z článků J.-P. Berruta, R. Baltenspergera a H. Mittelmana [5], J.-P. Berruta [6] a M. S. Floatera a K. Hormanna [7].

5.1 Barycentrický tvar racionální interpolace

Věta 5.1.1. *Mějme váhy $u_j \in \mathbb{R}$ a vstupní data (x_j, y_j) , $j = 0, \dots, n$, kde x_j jsou navzájem různá. Pak platí následující:*

a) *Pokud $u_j \neq 0$, pak racionální funkce $R \in \mathbb{R}_{n,n}$ definovaná pomocí*

$$R(x) = \frac{\sum_{j=0}^n \frac{u_j}{x - x_j} y_j}{\sum_{j=0}^n \frac{u_j}{x - x_j}} \quad (5.1)$$

limitně splňuje interpolační podmínky:

$$\lim_{x \rightarrow x_k} R(x) = y_k, \quad k = 0, \dots, n. \quad (5.2)$$

b) *Každý racionální interpolant $R^* \in \mathbb{R}_{n,n}$ vstupních dat lze zapsat ve tvaru (5.1) vhodnou volbou koeficientů u_j , $j = 0, \dots, n$.*

Důkaz.

- a) Po rozšíření zlomku členem $x - x_k$ dosadíme do limity a výsledek napočteme přímo:

$$\begin{aligned} \lim_{x \rightarrow x_k} R(x) &= \lim_{x \rightarrow x_k} \frac{\left(\sum_{\substack{j=0 \\ j \neq k}}^n \frac{u_j y_j}{x - x_j} \right) + \frac{u_k y_k}{x - x_k}}{\left(\sum_{\substack{j=0 \\ j \neq k}}^n \frac{u_j}{x - x_j} \right) + \frac{u_k}{x - x_k}} \\ &= \lim_{x \rightarrow x_k} \frac{\left(\sum_{\substack{j=0 \\ j \neq k}}^n \frac{u_j y_j}{x - x_j} \right) (x - x_k) + u_k y_k}{\left(\sum_{\substack{j=0 \\ j \neq k}}^n \frac{u_j}{x - x_j} \right) (x - x_k) + u_k} = \frac{u_k y_k}{u_k} = y_k. \end{aligned}$$

- b) Uvažujme racionální interpolant $R^* \in \mathbb{R}_{n,n}$ vstupních dat, tzn.

$$R^*(x_j) = \frac{P^*(x_j)}{Q^*(x_j)} = y_j, \quad P^*, Q^* \in \mathbb{P}_n, \quad j = 0, \dots, n. \quad (5.3)$$

Zapíšeme-li čitatele a jmenovatele v barycentrickém tvaru (3.2), dostaneme

$$\begin{aligned} P^*(x) &= \ell(x) \sum_{j=0}^n \frac{w_j}{x - x_j} P^*(x_j), \\ Q^*(x) &= \ell(x) \sum_{j=0}^n \frac{w_j}{x - x_j} Q^*(x_j), \end{aligned}$$

a uvědomíme-li si, že $P^*(x_j) = y_j Q^*(x_j)$ dle (5.3), můžeme celý výraz R^* zapsat jako

$$R^*(x) = \frac{P^*(x)}{Q^*(x)} = \frac{\sum_{j=0}^n \frac{w_j Q^*(x_j)}{x - x_j} y_j}{\sum_{j=0}^n \frac{w_j Q^*(x_j)}{x - x_j}}.$$

Odtud vyjádříme váhy $u_j = w_j Q^*(x_j)$.

□

Vidíme, že pokud $Q^* \equiv 1$, tak se jedná o barycentrickou Lagrangeovu interpolaci.

5.2 Odstranění bodů nespojitosti pomocí Berrutových vah

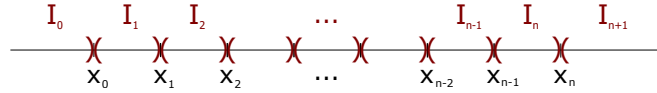
V příkladu 4.1.3 jsme si předvedli, že mohou existovat body nespojitosti na intervalu interpolace. Interpolujeme-li však pomocí vzorce (5.1), správná volba koeficientů u_j může zaručit, že jmenovatel bude nenulový všude na $\mathbb{R} \setminus \{x_0, \dots, x_n\}$.

Věta 5.2.1. *Mějme uzly x_0, x_1, \dots, x_n , jež splňují $x_0 < x_1 < \dots < x_n$, a necht' $\ell(x) = (x - x_0)(x - x_1) \cdots (x - x_n)$. Pak*

$$\forall x \in \mathbb{R} : \quad \ell(x) \sum_{k=0}^n \frac{(-1)^k}{x - x_k} \neq 0. \quad (5.4)$$

Důkaz. (viz str. 4 v [6]) Je zřejmé, že každý člen sumy v (5.4) představuje předpis pro rovnosou hyperbolu s vertikální asymptotou v bodě x_k . Intervaly mezi těmito asymptotami si označíme následovně (viz také obrázek 5.2.1):

$$I_0 = (-\infty, x_0), \quad I_k = (x_{k-1}, x_k), \quad I_{n+1} = (x_n, \infty), \quad k = 1, 2, \dots, n.$$



Obrázek 5.2.1: Intervaly I_0, \dots, I_{n+1}

Je zřejmé, že uvnitř těchto intervalů funkce ℓ nemění znaménko, tedy neprochází nulou. Nyní potřebujeme ukázat, že suma ve výroku (5.4) také nemění na těchto intervalech znaménko, aby bylo tvrzení věty splněno.

Definujeme si funkce

$$l_k(x) = \begin{cases} \frac{(-1)^k}{x - x_k}, & x < x_k, \\ 0, & x > x_k, \end{cases} \quad \text{a} \quad r_k(x) = \begin{cases} 0, & x < x_k, \\ \frac{(-1)^k}{x - x_k}, & x > x_k. \end{cases}$$

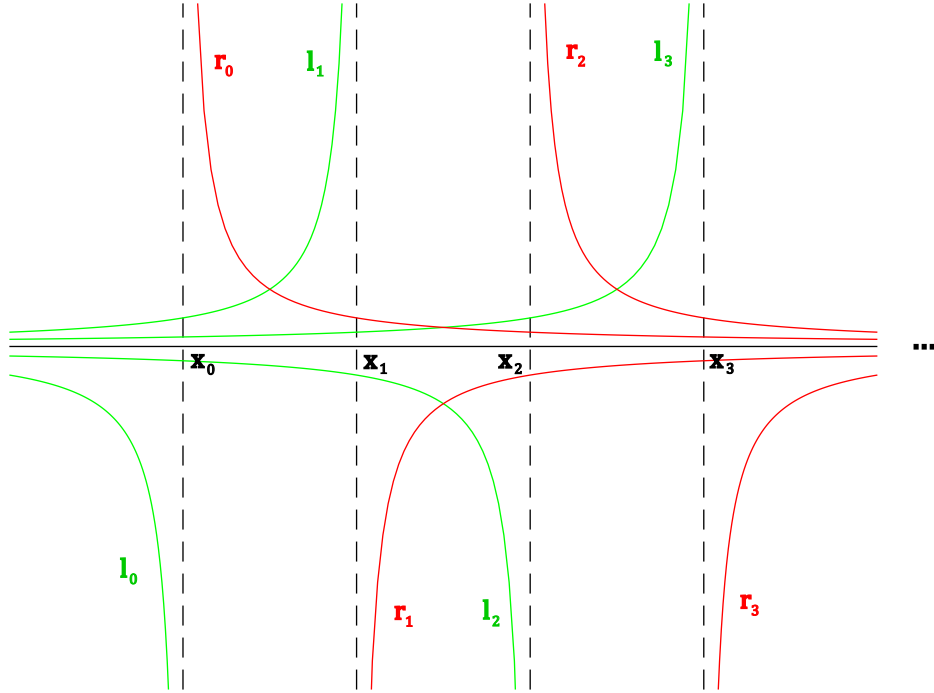
Každé l_k obsahuje „levou větev“ rovnosé hyperboly s asymptotou v $x = x_k$. Analogicky, všechna r_k jsou „pravými větvemi“ těchto hyperbol, viz obrázek 5.2.2 (nulové části funkcí nejsou v obrázku zakresleny). Pomocí funkcí

$$l(x) = \sum_{k=0}^n l_k(x) \quad \text{a} \quad r(x) = \sum_{k=0}^n r_k(x) \quad \text{vyjádříme sumu z (5.4) jako}$$

$$\sum_{k=0}^n \frac{(-1)^k}{x - x_k} = \sum_{k=0}^n (l_k(x) + r_k(x)) = l(x) + r(x).$$

Pro fixní $i \in \{0, \dots, n+1\}$ a fixní $x \in I_i$ dostáváme hodnoty funkcí l a r v x ve tvaru

$$l(x) = \sum_{k=i}^n l_k(x) \quad \text{a} \quad r(x) = \sum_{k=0}^{i-1} r_k(x).$$



Obrázek 5.2.2: Větve hyperbol $\frac{(-1)^k}{x - x_k}$

Členy sumy $l(x)$ tvoří konečnou, alternující a v absolutní hodnotě klesající posloupnost. Znaménko sumy je tedy určeno znaménkem prvního členu:

$$\operatorname{sgn} l(x) = \operatorname{sgn} l_i(x) = (-1)^{i-1}.$$

Členy sumy $r(x)$ tvoří konečnou, alternující, ale v absolutní hodnotě rostoucí posloupnost. Znaménko sumy je pak určeno znaménkem posledního členu:

$$\operatorname{sgn} r(x) = \operatorname{sgn} r_{i-1}(x) = (-1)^{i-1}.$$

Vidíme, že platí $\operatorname{sgn} l(x) = \operatorname{sgn} r(x)$ pro všechna $x \in I_i$, $i = 0, 1, \dots, n+1$, z čehož vyplývá neměnnost znaménka $\sum_{k=0}^n \frac{(-1)^k}{x - x_k}$ na jednotlivých intervalech, což jsme potřebovali ukázat. \square

Z vět 5.2.1 a 5.1.1 tedy ihned plyne věta 5.2.2.

Věta 5.2.2. *Racionální funkce*

$$R(x) = \frac{\ell(x) \sum_{k=0}^n \frac{(-1)^k}{x - x_k} y_k}{\ell(x) \sum_{k=0}^n \frac{(-1)^k}{x - x_k}} \quad (5.5)$$

interpoluje vzorky dat (x_i, y_i) , $i = 0, 1, \dots, n$, ve smyslu (5.2) a je spojitá pro $x \in \mathbb{R} \setminus \{x_0, \dots, x_n\}$.

Interpoláčnı́ funkci (5.5) pak stačí v bodech x_i dodefinovat jako $R(x_i) = y_i$, $i = 0, 1, \dots, n$, a získáme spojitost na celém \mathbb{R} . Nakonec krácením výrazu $\ell(x)$ v předpisu (5.5) získáme barycentrický tvar (3.3)

$$R(x) = \frac{\sum_{k=0}^n \frac{(-1)^k}{x - x_k} y_k}{\sum_{k=0}^n \frac{(-1)^k}{x - x_k}}.$$

Váhy $(-1)^k$ byly navrženy J.-P. Berrutem [6]. Nejsou ale jediné, které můžeme pro interpolaci použít a získat nenulovost jmenovatele na celém \mathbb{R} . Jednou z motivací přestoupit k jiným vahám může být snížení chyby interpolace. Dá se ukázat, že tohoto můžeme dosáhnout pomocí následujícího systému vah (viz str. 4 v [7]).

5.3 Floaterovy-Hormannovy váhy

V této sekci čerpáme z prací M. S. Floatera a K. Hormanna [7] a J. Venclovského [8].

Věta 5.3.1. *Mějme uzly x_0, x_1, \dots, x_n , jež splňují $x_0 < x_1 < \dots < x_n$, a nechť $d \in \{0, 1, \dots, n\}$. Uvažujme mnohočlen $p_i \in \mathbb{P}_d$ takový, že pro každé $i = 0, 1, \dots, n - d$ interpoluje $d + 1$ bodů $(x_i, y_i), \dots, (x_{i+d}, y_{i+d})$. Definujme dále λ_i jako*

$$\lambda_i(x) = \frac{(-1)^i}{(x - x_i) \cdots (x - x_{i+d})}.$$

Potom pro dané d dostáváme racionální interpolant dat $(x_0, y_0), \dots, (x_n, y_n)$ ve tvaru

$$R(x) = \frac{\sum_{i=0}^{n-d} \lambda_i(x) p_i(x)}{\sum_{i=0}^{n-d} \lambda_i(x)}. \quad (5.6)$$

Důkaz. V první řadě ukažme, že (5.6) je v racionálním tvaru, přesněji patří do $\mathbb{R}_{n,n-d}$. Rozšířme zlomek mnohočlenem $(-1)^{n-d}(x-x_0)\cdots(x-x_n)$. Dostáváme

$$R(x) = \frac{\sum_{i=0}^{n-d} \mu_i(x) p_i(x)}{\sum_{i=0}^{n-d} \mu_i(x)}, \quad (5.7)$$

kde

$$\begin{aligned} \mu_i(x) &= (-1)^{n-d} \prod_{j=0}^n (x-x_j) \lambda_i(x) \\ &= (-1)^{n-d} (-1)^i \prod_{j=0}^{i-1} (x-x_j) \prod_{k=i+d+1}^n (x-x_k) \\ &= (-1)^{n-d-i} \prod_{j=0}^{i-1} (x-x_j) \prod_{k=i+d+1}^n (x-x_k) \\ &= \prod_{j=0}^{i-1} (x-x_j) \prod_{k=i+d+1}^n (-1)(x-x_k) \\ &= \prod_{j=0}^{i-1} (x-x_j) \prod_{k=i+d+1}^n (x_k-x). \end{aligned}$$

Podotkněme, že prázdný součin je roven 1. Lze snadno vidět, že každé $\mu_i \in \mathbb{P}_{n-d}$, $i = 0, \dots, n-d$. Jelikož každé $p_i \in \mathbb{P}_d$, platí $R \in \mathbb{R}_{n,n-d}$. Nyní jen potřebujeme ověřit platnost interpolačních podmínek (5.2). Zavedme si pro tento účel následující indexové množiny J_k :

$$J_k = \{a \in A : k-d \leq a \leq k\}, \quad A = \{0, \dots, n-d\}. \quad (5.8)$$

Můžeme si rozmyslet, že pro každé $k \in \{0, \dots, n\}$ platí:

- $\forall i \in J_k : p_i(x_k) = y_k$,
- $\forall i \in J_k : \mu_i(x_k) > 0$,
- $\forall i \in A \setminus J_k : \mu_i(x_k) = 0$.

Tyto vlastnosti nám umožní zjednodušit indexy sum v zápise (5.7) v případě, kdy budeme (limitně) vyjadřovat hodnoty $R(x_k)$:

$$\lim_{x \rightarrow x_k} R(x) = \lim_{x \rightarrow x_k} \frac{\sum_{i \in J_k} \mu_i(x) p_i(x)}{\sum_{i \in J_k} \mu_i(x)} = y_k \lim_{x \rightarrow x_k} \frac{\sum_{i \in J_k} \mu_i(x)}{\sum_{i \in J_k} \mu_i(x)} = y_k.$$

□

Věta 5.3.2. *Racionální interpolant ve tvaru (5.6) je spojitá funkce na $\mathbb{R} \setminus \{x_0, \dots, x_n\}$.*

Důkaz. Viz stranu 3 v [7]. □

Věta 5.3.3. *Racionální interpolant ve tvaru (5.6) lze přepsat do barycentrického tvaru (5.1) s vahami*

$$u_k = \sum_{i \in J_k} (-1)^i \prod_{\substack{j=i \\ j \neq k}}^{i+d} \frac{1}{x_k - x_j}, \quad k = 0, \dots, n,$$

kde J_k je definováno pomocí (5.8).

Důkaz. Zapišme polynom p_i pomocí Lagrangeovy interpolace (2.3):

$$p_i(x) = \sum_{k=i}^{i+d} \left(\prod_{\substack{j=i \\ j \neq k}}^{i+d} \frac{x - x_j}{x_k - x_j} \right) y_k.$$

Vložíme tento tvar do čitatele (5.6) a upravíme:

$$\begin{aligned} \sum_{i=0}^{n-d} \lambda_i(x) p_i(x) &= \sum_{i=0}^{n-d} (-1)^i \frac{1}{\prod_{j=i}^{i+d} (x - x_j)} \sum_{k=i}^{i+d} \left(\prod_{\substack{j=i \\ j \neq k}}^{i+d} \frac{x - x_j}{x_k - x_j} \right) y_k \\ &= \sum_{i=0}^{n-d} (-1)^i \sum_{k=i}^{i+d} \frac{1}{x - x_k} \left(\prod_{\substack{j=i \\ j \neq k}}^{i+d} \frac{1}{x_k - x_j} \right) y_k \\ &= \sum_{k=0}^n \frac{y_k}{x - x_k} \sum_{i \in J_k} (-1)^i \left(\prod_{\substack{j=i \\ j \neq k}}^{i+d} \frac{1}{x_k - x_j} \right) \end{aligned}$$

Jmenovatele ve vzorci (5.6) lze zapsat stejně, jen místo hodnot y_k budou jedničky. Dosadíme-li zpět do vzorce (5.6), získáme barycentrickou racionální interpolaci

$$R(x) = \frac{\sum_{k=0}^n \frac{u_k}{x - x_k} y_k}{\sum_{k=0}^n \frac{u_k}{x - x_k}}, \quad u_k = \sum_{i \in J_k} (-1)^i \prod_{\substack{j=i \\ j \neq k}}^{i+d} \frac{1}{x_k - x_j}.$$

□

Lze si také rozmyslet [7], že Floaterovy-Hormannovy váhy alternují ve znaménku:

$$u_k = (-1)^{k-d} \sum_{i \in J_k} \prod_{\substack{j=i \\ j \neq k}}^{i+d} \frac{1}{|x_k - x_j|}. \quad (5.9)$$

Tuto vlastnost uplatníme pro vyjádření Floaterových-Hormannových vah pro ekvidistantní uzly. Všimněme si také, že pokud $d = 0$, získáváme Berrtovy váhy $(-1)^k$.

5.3.1 Ekvidistantní uzly

Věta 5.3.4. *Floaterovy-Hormannovy váhy w_k při použití ekvidistantní distribuce $n + 1$ uzlů lze zjednodušeně vyjádřit pomocí*

$$w_k = (-1)^k \sum_{i \in J_k} \binom{d}{k-i}$$

pro $k \in \{0, 1, \dots, n\}$, kde J_k je definováno pomocí (5.8).

Důkaz. Budeme postupovat podobně jako při odvození vah ekvidistantních bodů barycentrické Lagrangeovy interpolace (3.4). Každé dva ekvidistantní uzly x_k a x_j můžeme nahradit součty $a + kh$ a $a + jh$, kde h je vzdálenost mezi dvěma libovolnými sousedními uzly. Pak

$$|x_k - x_j| = |a + kh - a - jh| = |k - j| \cdot h.$$

Dosaďme nyní upravený rozdíl $|x_k - x_j|$ do vzorce pro Floaterovy-Hormannovy váhy (5.9):

$$w_k = (-1)^{k-d} \sum_{i \in J_k} \prod_{\substack{j=i \\ j \neq k}}^{i+d} \frac{1}{|k - j| \cdot h} = \frac{(-1)^{k-d}}{h^d} \sum_{i \in J_k} \prod_{\substack{j=i \\ j \neq k}}^{i+d} \frac{1}{|k - j|}. \quad (5.10)$$

Nyní se zaměříme na poslední součin v (5.10). Pro přehlednost budeme pracovat s jeho převrácenou hodnotou. Můžeme psát:

$$\begin{aligned} \left(\prod_{\substack{j=i \\ j \neq k}}^{i+d} \frac{1}{|k - j|} \right)^{-1} &= \prod_{j=i}^{k-1} |k - j| \cdot \prod_{j=k+i}^{i+d} |k - j| \\ &= (k-i)! \cdot (i+d-k)! \\ &= (k-i)! \cdot (d - (k-i))! \cdot \frac{d!}{d!} = \frac{d!}{\binom{d}{k-i}}. \end{aligned}$$

Dosažením zpět do (5.10) dostáváme

$$w_k = \frac{(-1)^{k-d}}{h^d} \sum_{i \in J_k} \frac{\binom{d}{k-i}}{d!} = \frac{(-1)^{k-d}}{h^d d!} \sum_{i \in J_k} \binom{d}{k-i}.$$

V barycentrickém racionálním tvaru se převrácené hodnoty h^d , $d!$ a $(-1)^{-d}$ zkrátí a váhy nakonec můžeme zapsat ve zjednodušeném tvaru jako

$$w_k = (-1)^k \sum_{i \in J_k} \binom{d}{k-i}, \quad k = 0, \dots, n.$$

□

Při redukci složitosti počítání kombinačních čísel lze opět využít principu Pascalova trojúhelníku jako u ekvidistantních vah pro barycentrickou Lagrangeovu interpolaci (viz kapitolu 3.3.1, str. 19). Pro ukázkou uveďme výčet absolutních hodnot vah ekvidistantních uzlů pro několik prvních d do tabulky 5.1.

| d | absolutní hodnoty vah w_k | | | | | | | | | | |
|-----|-----------------------------|---|----|----|----|-----|----|----|----|---|---|
| 0 | 1 | 1 | 1 | 1 | 1 | ... | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 2 | 2 | 2 | 2 | ... | 2 | 2 | 2 | 2 | 1 |
| 2 | 1 | 3 | 4 | 4 | 4 | ... | 4 | 4 | 4 | 3 | 1 |
| 3 | 1 | 4 | 7 | 8 | 8 | ... | 8 | 8 | 7 | 4 | 1 |
| 4 | 1 | 5 | 11 | 15 | 16 | ... | 16 | 15 | 11 | 5 | 1 |

Tabulka 5.1: Absolutní hodnoty vah ekvidistantních uzlů

5.4 Numerické experimenty

V této sekci budeme interpolovat funkce pomocí Floaterových-Hormannových vah. Interpolace bude prováděna na ekvidistantních uzlech rozprostřených do intervalu $\langle -1, 1 \rangle$. Interpolant bude vyčíslován v 50000 ekvidistantních bodech.

Příklad 5.4.1. *Uvažujme škálovanou Rungeovu funkci*

$$h_1(x) = \frac{1}{1 + 25x^2}$$

(viz obrázek 3.4.1(a)), kterou navzorkujeme, a následně budeme vzorky interpolovat. Jednu interpolaci provedeme pomocí Floaterových-Hormannových vah s parametrem $d = 3$, druhou podle Berrutových vah ($d = 0$) a pro třetí jsme použili barycentrickou Lagrangeovu metodu v Čebyševových uzlech 2. druhu. Výsledky vidíme v tabulce 5.2. U této funkce můžeme pozorovat, že pro zvyšující se počet uzlů chyba konverguje [7] pro každou metodu v různých rychlostech. Floaterovy-Hormannovy váhy jsou výhodnější než Berrutovy, barycentrický Lagrangeův přístup v Čebyševových uzlech je ale řádově lepší. Experimentálně ovšem dokážeme nalézt nejlepší hodnotu d , která chybu interpolace minimalizuje pro případ použití Floaterových-Hormannových vah. Konkrétní hodnoty jsme uvedli v tabulce 5.3. V závislosti na počtu uzlů bychom tedy měli para-

| počet uzlů | F.-H., $d = 3$ | Berrut, $d = 0$ | bar. Lag. |
|------------|-----------------------|----------------------|-----------------------|
| 10 | $6,91 \cdot 10^{-2}$ | $3,61 \cdot 10^{-2}$ | $1,32 \cdot 10^{-1}$ |
| 20 | $2,83 \cdot 10^{-3}$ | $4,56 \cdot 10^{-3}$ | $1,77 \cdot 10^{-2}$ |
| 40 | $4,31 \cdot 10^{-6}$ | $1,46 \cdot 10^{-3}$ | $3,40 \cdot 10^{-4}$ |
| 80 | $5,12 \cdot 10^{-8}$ | $7,47 \cdot 10^{-4}$ | $1,20 \cdot 10^{-7}$ |
| 160 | $3,01 \cdot 10^{-9}$ | $3,78 \cdot 10^{-4}$ | $1,57 \cdot 10^{-14}$ |
| 320 | $1,82 \cdot 10^{-10}$ | $1,90 \cdot 10^{-4}$ | $3,11 \cdot 10^{-15}$ |
| 640 | $1,12 \cdot 10^{-11}$ | $9,54 \cdot 10^{-5}$ | $4,11 \cdot 10^{-15}$ |

Tabulka 5.2: Srovnání chyb interpolací

| počet uzlů | nejlepší d | max. abs. chyba interpolace |
|------------|--------------|-----------------------------|
| 10 | 0 | $3,61 \cdot 10^{-2}$ |
| 20 | 1 | $1,54 \cdot 10^{-3}$ |
| 40 | 3 | $4,31 \cdot 10^{-6}$ |
| 80 | 7 | $2,04 \cdot 10^{-10}$ |
| 160 | 10 | $2,44 \cdot 10^{-15}$ |
| 320 | 9 | $3,00 \cdot 10^{-15}$ |
| 640 | 10 | $4,44 \cdot 10^{-15}$ |

Tabulka 5.3: Chyby pro nejlepší d

metr d měnit. Výhodou barycentrického racionálního přístupu s Floaterovými-Hormannovými vahami oproti barycentrické Lagrangeově metodě je ten, že nepotřebujeme Čebyševovo rozložení bodů, abychom snížili chybu interpolace, stačí vybrat nejlepší hodnotu d pro ekvidistantní uzly. Pokud ale nevíme, jakou funkci interpolujeme, nemáme na základě čeho hodnotu d hledat. Pro Rungeovu funkci je příhodné pomalu zvyšovat d s narůstajícím n [7]. Poslední dva řádky tabulky 5.3 tuto radu vyvracejí, to je ale způsobeno tím, že chyba je na hranici počítačové přesnosti.

△

Příklad 5.4.2. Nyní proved'me ještě experiment pro funkci

$$g(x) = \frac{x}{2} + |x| - x^2$$

(viz obrázek 4.5.1). Z tabulky 5.4 vidíme, že v tomto případě chyba interpolace konverguje velice pomalu. Pomalost konvergence způsobuje nehladkost funkce g , v bodě 0 totiž není diferencovatelná. Nejlepší d také vykazuje pomalé snižování chyby (viz tabulka 5.5).

| počet uzlů | F.-H., $d = 3$ | Berrut, $d = 0$ | bar. Lag. |
|------------|----------------------|----------------------|----------------------|
| 10 | $3,81 \cdot 10^{-2}$ | $6,97 \cdot 10^{-2}$ | $5,92 \cdot 10^{-2}$ |
| 20 | $1,90 \cdot 10^{-2}$ | $3,44 \cdot 10^{-2}$ | $2,98 \cdot 10^{-2}$ |
| 40 | $9,5 \cdot 10^{-3}$ | $1,73 \cdot 10^{-2}$ | $1,49 \cdot 10^{-2}$ |
| 80 | $4,75 \cdot 10^{-3}$ | $8,63 \cdot 10^{-3}$ | $7,46 \cdot 10^{-3}$ |
| 160 | $2,38 \cdot 10^{-3}$ | $4,31 \cdot 10^{-3}$ | $3,73 \cdot 10^{-3}$ |
| 320 | $1,19 \cdot 10^{-3}$ | $2,16 \cdot 10^{-3}$ | $1,87 \cdot 10^{-3}$ |
| 640 | $5,94 \cdot 10^{-4}$ | $1,08 \cdot 10^{-3}$ | $9,33 \cdot 10^{-4}$ |

Tabulka 5.4: Srovnání chyb interpolací

| počet uzlů | nejlepší d | max. abs. chyba interpolace |
|------------|--------------|-----------------------------|
| 10 | 2 | $3,79 \cdot 10^{-2}$ |
| 20 | 1 | $1,89 \cdot 10^{-2}$ |
| 40 | 1 | $9,49 \cdot 10^{-3}$ |
| 80 | 1 | $4,75 \cdot 10^{-3}$ |
| 160 | 1 | $2,38 \cdot 10^{-3}$ |
| 320 | 1 | $1,19 \cdot 10^{-3}$ |
| 640 | 1 | $5,94 \cdot 10^{-4}$ |

Tabulka 5.5: Chyby pro nejlepší d

△

Kapitola 6

Závěr

Cílem této práce bylo seznámení se s principem a vlastnostmi barycentrické Lagrangeovy interpolace s následným zobecněním na barycentrickou racionální interpolaci. Nastudovali jsme si, jaké výhody barycentrický tvar přináší. Kromě numerické stability jsme se dozvěděli, že není potřeba vzorkovat data v Čebyševových uzlech pro zajištění konvergence chyby. Zároveň by byla určitě zajímavá studie chování barycentrické racionální interpolace s Floaterovými-Hormannovými vahami v závislosti na hodnotě d . Vytvořili jsme také kódy v MATLABu, pomocí nichž jsme realizovali pokusy a podle nich analyzovali jednotlivé metody interpolací.

Literatura

- [1] W. Shen: *An Introduction To Numerical Computation*. World Scientific, 2019.
- [2] T. Staško: *Nejlepší polynomiální aproximace*. Bakalářská práce, VŠB, Ostrava, 2020. <https://dspace.vsb.cz/handle/10084/140434>
- [3] J.-P. Berrut, L. N. Trefethen: *Barycentric Lagrange Interpolation*. Society for Industrial and Applied Mathematics, 2004.
- [4] J. Stoer, R. Bulirsch: *Introduction to Numerical Analysis, Second Edition*. Springer-Verlag New York, Inc., 1993.
- [5] J.-P. Berrut, R. Baltensperger, H. Mittelman: *Recent Developments in Barycentric Rational Interpolation*. Birkhäuser Verlag Basel, 2005.
- [6] J.-P. Berrut: *Rational Functions for Guaranteed and Experimentally Well-Conditioned Global Interpolation*. University of California, San Diego, 1988.
- [7] M. S. Floater, K. Hormann: *Barycentric rational interpolation with no poles and high rates of approximation*. Journal Numerische Mathematik, 2007.
- [8] J. Venclovský: *Barycentrická racionální interpolace*. Bakalářská práce, VUT, Brno, 2012.
<https://dspace.vutbr.cz/handle/11012/13986?locale-attribute=cs>

Internetové zdroje citovány k datu 30. 4. 2022.

Příloha A: MATLAB kódy

Popis vstupních argumentů níže uvedených funkcí:

x, y - vektory vstupních dat

xx - vektor hodnot na ose x , ve kterých budeme vyčíslovat interpolant

w - vektor vah

a, b - meze intervalu pro generaci uzlů vybraného rozdělení

$n+1$ - počet uzlů použitých pro interpolaci

Pro vyčíslení kombinačních čísel používáme zabudovanou MATLABovskou funkci *nchoosek*.

Lagrangeova interpolace

```

1 function I = LagrangeInterpolation(x,y,xx)
2
3 n = length(x);
4 I = zeros(size(xx));
5
6 for i = 1 : n
7     Li = ones(size(xx));
8
9     for j = 1 : n
10         if j ~= i
11             Li = Li .* (xx-x(j))/(x(i)-x(j));
12         end
13     end
14
15     I = I + Li*y(i);
16 end

```

Newtonova interpolace

```

1 function I = NewtonInterpolation(x,y,xx)
2
3 n = length(x);
4 M = zeros(n);
5 I = zeros(size(xx));
6
7 M(:,1) = y;
8
9 for i = 2 : n

```



```

10     for j = i : n
11         M(j,i) = (M(j,i-1)-M(j-1,i-1))/(x(j)-x(j-i+1));
12     end
13 end
14
15 I = I + M(n,n);
16
17 for i = n-1 : -1 : 1
18     I = I .* (xx-x(i)) + M(i,i);
19 end

```

Ekvidistantní uzly

```

1 function p = EquidistantPoints(a,b,n)
2
3 p = linspace(a,b,n+1);

```

Čebyševovy uzly

```

1 function p = ChebyshevPoints(a,b,n)
2
3 i = 0 : n;
4 p = (a+b)/2 + (a-b)/2*cos((2*i+1)*pi/(2*n+2));

```

```

1 function p = ChebyshevPoints2(a,b,n)
2
3 i = 0 : n;
4 p = (a+b)/2 + (a-b)/2*cos(i*pi/n);

```

Barycentrické váhy pro Lagrangeovu interpolaci

```

1 function w = BarycentricWeightsEquidistant(n)
2
3 w = zeros(1,n+1);
4
5 for i = 0 : n
6     w(i+1) = (-1)^i*nchoosek(n,i);
7 end

```

```

1 function w = BarycentricWeightsChebyshev(n)
2
3 v = 0 : n;
4 w = (-1).^v.*sin((2*v+1)*pi/(2*n+2));

```



```

1 function w = BarycentricWeightsChebyshev2(n)
2
3 w = ones(1,n+1);
4 w(1) = 1/2;
5 w(end) = 1/2;
6 w(2:2:end) = w(2:2:end) .* (-1);

```

Metoda inverzních diferencí

```

1 function I = InverseDifferences(x,y,xx)
2
3 n = length(x);
4 M = zeros(n);
5 I = zeros(size(xx));
6
7 M(:,1) = y;
8
9 for i = 2 : n
10     for j = i : n
11         M(j,i) = (x(i-1) - x(j))/(M(i-1,i-1) - M(j,i-1));
12     end
13 end
14
15 I = I + M(n,n);
16
17 for i = n-1 : -1 : 1
18     I = (xx - x(i))./I + M(i,i);
19 end

```

Floaterovy-Hormannovy váhy

```

1 function w = FloaterHormannWeights(n,d)
2
3 w = (-1).^(0:n);
4 w_abs = zeros(1,n+1);
5
6 for k = 0 : n
7     for i = max(0,k-d) : min(k,n-d)
8         w_abs(k+1) = w_abs(k+1) + nchoosek(d,k-i);
9     end
10 end
11
12 w = w .* w_abs;

```


Barycentrická Lagrangeova/racionální interpolace

Základ funkce *BarycentricInterpolation* byl převzat ze strany 510 v článku [3].
Práce s proměnnými `exact_y` a `exact_yy` řeší dodefinování interpolantu
v interpolačních uzlech.

```
1 function I = BarycentricInterpolation(x,y,xx,w)
2
3 n = length(x);
4 m = size(xx);
5 numerator = zeros(m);
6 denominator = zeros(m);
7 exact_yy = zeros(m);
8 exact_y = zeros(1,n);
9
10 for i = 1 : n
11     xdiff = xx - x(i);
12     temp = w(i) ./ xdiff;
13     numerator = numerator + temp*y(i);
14     denominator = denominator + temp;
15     exact_yy(xdiff == 0) = 1;
16     exact_y(i) = nnz(xdiff == 0);
17 end
18
19 I = numerator ./ denominator;
20 I(exact_yy == 1) = y(exact_y == 1);
```