

# Numerické metody

Mirko Navara

<http://cmp.felk.cvut.cz/~navara/>

katedra kybernetiky FEL ČVUT

Karlovo náměstí, budova G, místnost 104a

<https://moodle.felk.cvut.cz/courses/B4B01NUM>

20. prosince 2023

## Obsah

<b>1</b>	<b>APROXIMACE FUNKCÍ</b>	<b>11</b>
1.1	Typické úlohy . . . . .	11
1.1.1	Aproximace funkcí v ekonomii . . . . .	11
1.1.2	Aproximace funkcí v teorii pravděpodobnosti a matematické analýze . . . . .	11
1.1.3	Aproximace funkcí v elektrotechnice . . . . .	11
1.1.4	Základní úloha aproximace . . . . .	12
1.2	Interpolace . . . . .	13
1.2.1	Prostá interpolace . . . . .	13
1.3	Interpolace polynomem . . . . .	14
1.3.1	Lagrangeova konstrukce interpolačního polynomu . . . . .	15
1.3.2	Newtonova konstrukce interpolačního polynomu . . . . .	17
1.3.3	Nevillův algoritmus . . . . .	18
1.3.4	Chyba aproximace interpolačním polynomem . . . . .	19
1.3.5	Čebyševovy polynomy . . . . .	22
1.3.6	Příklad použití interpolačního polynomu na reálných datech . . . . .	24
1.3.7	Hermitův interpolační polynom . . . . .	26
1.3.8	Aproximace Taylorovou řadou . . . . .	27
1.4	Interpolace spliny . . . . .	29
1.4.1	Kubický spline . . . . .	29
1.4.2	Příklad použití splinu na reálných datech . . . . .	32
1.5	Metoda nejmenších čtverců . . . . .	34
1.5.1	Řešení aproximace podle kritéria nejmenších čtverců . . . . .	35
1.5.2	Ortogonalizace . . . . .	37
1.5.3	Aproximace goniometrickým polynomem . . . . .	39
1.5.4	Čebyševova aproximace polynomem . . . . .	40
<b>2</b>	<b>NUMERICKÁ DERIVACE A RICHARDSONOVA EXTRAPOLACE</b>	<b>43</b>
2.1	Formulace problému . . . . .	43
2.2	Chyba metody u numerické derivace . . . . .	44
2.2.1	Řád metod numerické derivace . . . . .	44
2.3	Odhady chyb metody u numerické derivace . . . . .	45
2.4	Doporučená délka kroku . . . . .	46
2.5	Obecný princip Richardsonovy extrapolace . . . . .	47
2.6	Využití Richardsonovy extrapolace v numerické derivaci . . . . .	48
2.7	Odhad derivace z reálných dat . . . . .	49

<b>3</b>	<b>NUMERICKÁ INTEGRACE</b>	<b>60</b>
3.1	Newtonovy-Cotesovy vzorce . . . . .	61
3.1.1	Metoda levých obdélníků . . . . .	62
3.1.2	Obdélníková metoda . . . . .	62
3.1.3	Lichoběžníková metoda . . . . .	63
3.1.4	Simpsonova metoda . . . . .	63
3.1.5	Obecné Newtonovy-Cotesovy vzorce . . . . .	64
3.2	Odhad chyby numerické integrace . . . . .	64
3.3	Řád metod integrace . . . . .	65
3.4	Numerická integrace reálných dat . . . . .	66
3.5	Gaussova metoda integrace . . . . .	67
3.6	Richardsonova extrapolace při integraci . . . . .	68
3.6.1	Rombergova metoda . . . . .	70
3.7	Praktické stanovení počtu intervalů . . . . .	71
3.8	Řešení obtížnějších úloh úpravou zadání . . . . .	72
3.8.1	Integrace přes nekonečný interval . . . . .	72
3.8.2	Omezení intervalu . . . . .	73
3.8.3	Pomalu konvergentní integrály . . . . .	73
<b>4</b>	<b>NUMERICKÉ ŘEŠENÍ NELINEÁRNÍCH ROVNIC</b>	<b>77</b>
4.1	Formulace problému . . . . .	77
4.2	Metoda půlení intervalu neboli bisekce . . . . .	77
4.3	Metoda regula falsi . . . . .	77
4.4	Univerzální odhad chyby . . . . .	78
4.5	Metoda sečen . . . . .	79
4.6	Newtonova metoda (metoda tečen) . . . . .	80
4.6.1	Odhad chyby Newtonovy metody . . . . .	81
4.6.2	Konvergence Newtonovy metody . . . . .	82
4.6.3	Náhrada derivace numerickým odhadem . . . . .	83
4.7	Konvergence a její rychlost (řád metody) . . . . .	84
4.7.1	Řád Newtonovy metody . . . . .	86
4.7.2	Řád metody regula falsi . . . . .	86
4.8	Kombinace startovacích a zpřesňujících metod . . . . .	87
4.9	Metoda prosté iterace (MPI) . . . . .	88
4.9.1	Kontraktivní funkce . . . . .	89
4.9.2	Věta o pevném bodě . . . . .	89
4.9.3	Optimalizace MPI . . . . .	90
4.9.4	Řád metody prosté iterace . . . . .	90
4.9.5	Kritéria pro výběr metody řešení rovnic . . . . .	92
4.10	Podobné úlohy . . . . .	92
4.10.1	Hledání násobných kořenů . . . . .	92
4.10.2	Řešení algebraických rovnic neboli hledání kořenů polynomů . . . . .	93
4.10.3	Řešení rovnic v komplexním oboru . . . . .	93
4.10.4	Řešení soustav rovnic . . . . .	93
<b>5</b>	<b>NUMERICKÉ ŘEŠENÍ SOUSTAV LINEÁRNÍCH ROVNIC</b>	<b>95</b>
5.1	Formulace úlohy a její obtíž . . . . .	95
5.1.1	Druhy problémů . . . . .	95
5.1.2	Špatná podmíněnost . . . . .	95
5.1.3	Zdroje chyb . . . . .	96
5.2	Přímé metody . . . . .	96
5.2.1	Gaussova eliminace (GEM) . . . . .	96
5.2.2	Výběr hlavního prvku . . . . .	96
5.2.3	Gaussova-Jordanova redukce . . . . .	97
5.2.4	LU-rozklad . . . . .	97
5.2.5	Výpočet inverzní matice . . . . .	98
5.2.6	Výpočet determinantu . . . . .	98

5.2.7	Zpřesnění výsledků pomocí rezidua . . . . .	99
5.3	Iterační metody . . . . .	99
5.3.1	Normy vektorů a matic . . . . .	99
5.3.2	Vlastní čísla a spektrální poloměr . . . . .	104
5.3.3	Výpočet vlastních čísel . . . . .	104
5.3.4	Maticové iterační metody . . . . .	105
5.3.5	Jacobiova iterační metoda (JIM) . . . . .	105
5.3.6	Gaussova-Seidelova iterační metoda (GSM) . . . . .	106
5.3.7	Superrelaxační metoda (SOR – Successive OverRelaxation method) . . . . .	107
5.4	Jaký postup volit? . . . . .	108

## Proč se tím zabývat 1

### Podmínky zápočtu

- Adekvátní účast na cvičení v počítačové laboratoři (mnohé lze dělat i jinde, Maple si můžete instalovat)
- Zápočtové úlohy  $5 \times 6$  bodů  
Postačuje 18 bodů, zbytek tvoří bonus ke zkoušce  
K tomu vám mají napomoci přednášky a ...

## Literatura

Navara, M., Němeček, A.: *Numerické metody*. Skriptum ČVUT, Praha, dotisk 2005.

Press, W.H., Teukolsky, S.A., Vetterling, W.T., Flannery, B.P.: *Numerical Recipes (The Art of Scientific Computing)*. 3rd edition, Cambridge University Press, Cambridge, 2007. <http://www.nrbook.com/a/bookcpdf.php>

Knuth, D.E.: *Fundamental Algorithms*. Vol. 1 of *The Art of Computer Programming*, Addison-Wesley, Reading, MA, 3rd ed. 1997.

## Co se máte naučit

Použít probírané metody, a to **kvalifikovaně**, včetně zhodnocení výsledků, zodpovědnosti za ně a v případě potřeby volby jiného postupu.

Máte k dispozici počítač, ale nemáte mu slepě věřit a máte mít předem rámcovou představu, co má vyjít.

## Proč se tím zabývat 2

**Úloha:** Pacienta budeme 24 hodin sledovat (např. glykémii, krevní tlak, tep nebo **teplotu**). Lékař chce znát

- průměrnou hodnotu,
- maximum,
- minimum.

**Řešení 1** (snadné): Změříme mnohokrát (např. v pravidelných intervalech), z posloupnosti měření vypočteme aritmetický průměr, maximum a minimum.

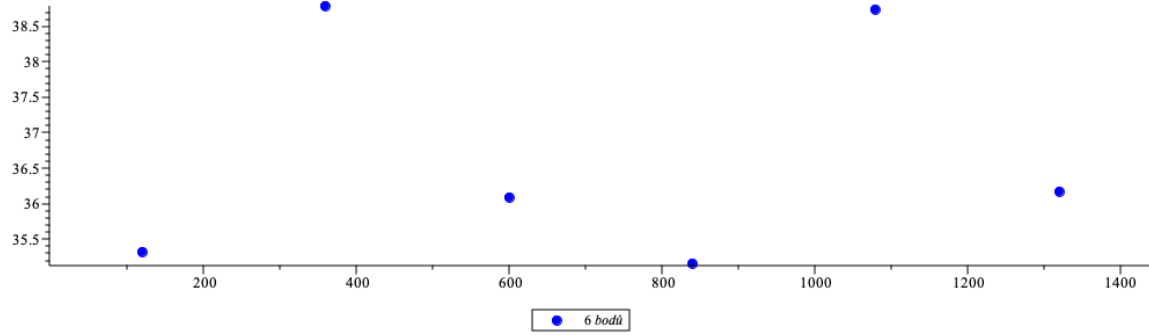
To nemá chybu.

Nebo ano?

Na konci kursu byste měli znát řadu námitek a vylepšení tohoto postupu.

## Měření teploty

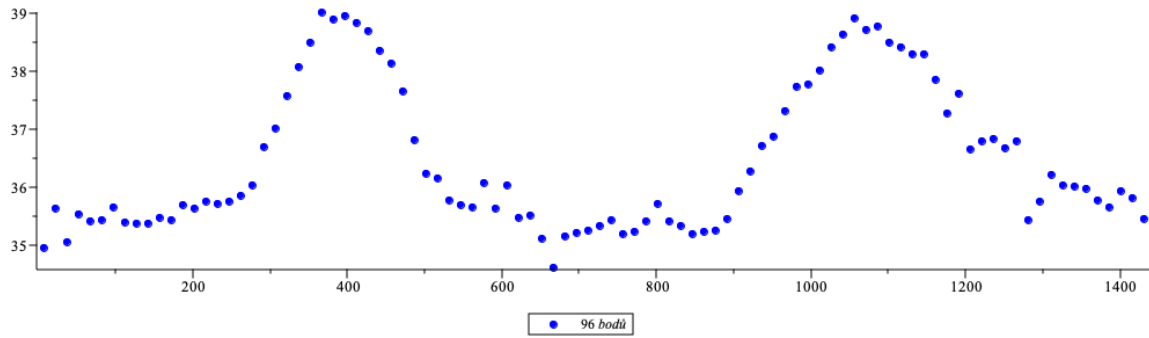
interval měření	počet údajů	průměr	min	max
ráno, poledne, večer	3	37.64	35.29	38.83
6 hodin	4	35.92	35.59	36.69
4 hodiny	6	36.71	35.16	38.79





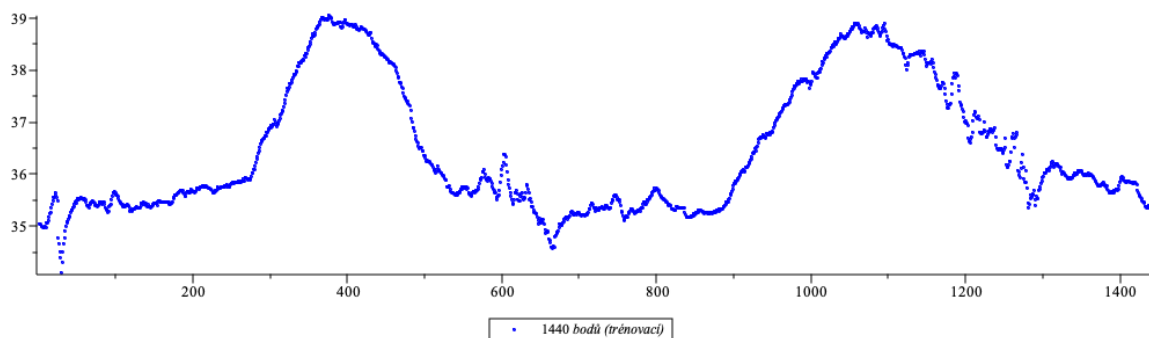
## Měření teploty

interval měření	počet údajů	průměr	min	max
ráno, poledne, večer	3	37.64	35.29	38.83
6 hodin	4	35.92	35.59	36.69
4 hodiny	6	36.71	35.16	38.79
1 hodina	24	36.41	34.11	38.93
15 minut	96	36.46	34.62	39.02



## Měření teploty

interval měření	počet údajů	průměr	min	max
ráno, poledne, večer	3	37.64	35.29	38.83
6 hodin	4	35.92	35.59	36.69
4 hodiny	6	36.71	35.16	38.79
1 hodina	24	36.41	34.11	38.93
15 minut	96	36.46	34.62	39.02
1 minuta	1440	36.45	34.11	39.05



# 1 APROXIMACE FUNKCÍ

## 1.1 Typické úlohy

### 1.1.1 Aproximace funkcí v ekonomii

**Úloha:** Odhadněte rychlost růstu průmyslové výroby v posledním období a nejlépe i v blízké budoucnosti.

**Úloha:** Ze známých hodnot burzovních indexů do dnešního dne odhadněte jejich zítřejší hodnoty.

### 1.1.2 Aproximace funkcí v teorii pravděpodobnosti a matematické analýze

**Úloha:** Náhodná veličina s normovaným normálním rozdělením je popsána distribuční funkcí

$$\Phi(u) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^u \exp\left(-\frac{t^2}{2}\right) dt.$$

**(Čistý) matematik:** To je transcendentní funkce.

**Numerický matematik:** Numerická integrace dá přibližný výsledek s požadovanou přesností.

(Ve skutečnosti i exponenciální funkce je počítána jen numericky, procesor sám umí jen 4 základní početní úkony.)

Pro rychlejší opakovaný výpočet si připravíme tabulku Gaussova integrálu.

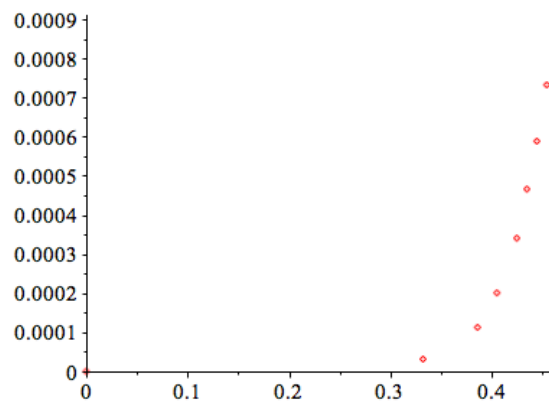
### 1.1.3 Aproximace funkcí v elektrotechnice

**Úloha:** Ze známého napětí baterie (v mobilu, v počítači) odhadněte zbývající kapacitu.

Vycházíme z konečně mnoha hodnot, ale aproximaci chceme použít na celém intervalu.

Zde navíc chceme, aby byla monotónní.

**Úloha:** Máme nakreslit V-A charakteristiku diody na základě naměřených dat:



**Úloha:** Digitální obrázek zvětšíme, popř. otočíme. Změní se počet pixelů, popř. i jejich orientace.

## Motivační úlohy na aproximaci

- **Úloha 1:** Závislost směnného kursu na čase na základě údajů z burzy.
- **Úloha 2:** Rychlý odhad Gaussova integrálu (distribuční funkce normálního rozdělení) s využitím tabulkových hodnot.
- **Úloha 3:** V-A charakteristika diody na základě naměřených hodnot.
- **Úloha 4:** Teploty naměřené na meteorologické stanici.

Data jsou vždy diskrétní (konečná).

Rozlišujeme:

1. aproximaci spojitě funkce pomocí konečných dat (1. zápočtová úloha),
2. nalezení funkce s malým počtem parametrů, která přibližně odpovídá datům (2. zápočtová úloha).

#### 1.1.4 Základní úloha aproximace

**Dáno:**

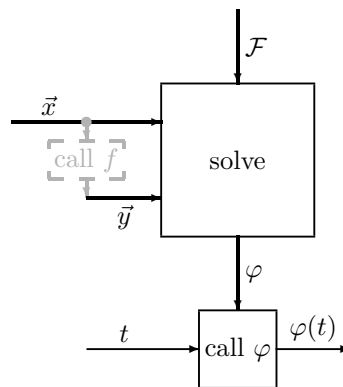
vektor  $\vec{x} = (x_0, \dots, x_{n-1}) \in \mathbb{R}^n$  určující  $n$  (různých) **uzlových bodů**,

vektor  $\vec{y} = (y_0, \dots, y_{n-1}) \in \mathbb{R}^n$  požadovaných hodnot v uzlových bodech,

množina funkcí  $\mathcal{F}$ , které smíme použít k aproximaci (definovaných alespoň v bodech  $x_0, \dots, x_{n-1}$ ).

**Úkol:** vybrat funkci  $\varphi \in \mathcal{F}$  tak, aby vektor  $\varphi(\vec{x}) = (\varphi(x_0), \dots, \varphi(x_{n-1})) \in \mathbb{R}^n$  byl „co nejbližší“ danému vektoru  $\vec{y}$ .

#### Blokové schéma obecné aproximace



**Předpoklad:**  $\mathcal{F}$  je lineární obal  $k$  známých, tzv. **aproximačních funkcí**  $\varphi_0, \dots, \varphi_{k-1}$ ,  $k \leq n$ :

$$\mathcal{F} = \text{Lin}\{\varphi_0, \dots, \varphi_{k-1}\} = \left\{ \sum_{j < k} c_j \varphi_j : c_j \in \mathbb{R} \right\}.$$

Zbývá určit vektor reálných koeficientů  $\vec{c} = (c_0, \dots, c_{k-1})$  lineární kombinace

$$\varphi = \sum_{j < k} c_j \varphi_j.$$

#### Vektorová formulace aproximační úlohy

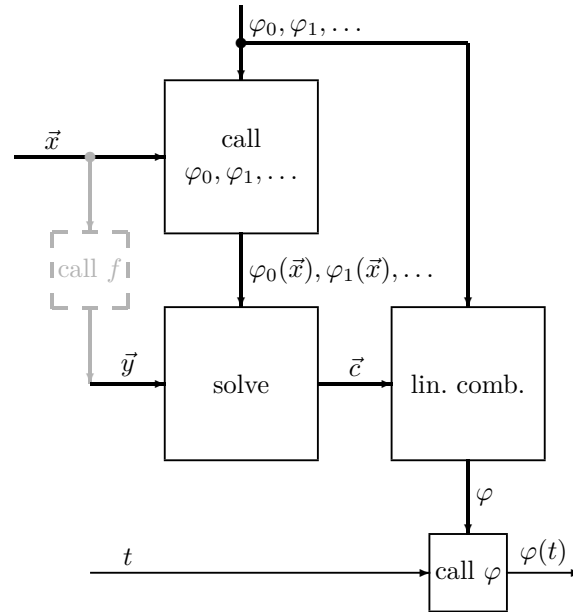
Aproximační funkce jsou reprezentovány vektory  $\varphi_j(\vec{x}) = (\varphi_j(x_0), \dots, \varphi_j(x_{n-1}))$ ,  $j = 0, \dots, k-1$ . (Žádné jiné jejich hodnoty do aproximační úlohy nevstupují.)

Pro výslednou aproximaci platí

$$\varphi(\vec{x}) = \sum_{j < k} c_j \varphi_j(\vec{x}),$$

neboli k  $\vec{y}$  hledáme „co nejbližší“ vektor  $\varphi(\vec{x})$  z lineárního obalu **známých** vektorů  $\varphi_0(\vec{x}), \dots, \varphi_{k-1}(\vec{x})$ .

## Blokové schema obecné aproximace lineární kombinací



### 1.2 Interpolace

Speciální případ aproximace, kdy požadujeme **přesnou shodu** v uzlových bodech.

#### Dáno:

vektor  $\vec{x} = (x_0, \dots, x_{n-1}) \in \mathbb{R}^n$  určující  $n$  (různých) uzlových bodů,  
vektor  $\vec{y} = (y_0, \dots, y_{n-1}) \in \mathbb{R}^n$  požadovaných hodnot v uzlových bodech,  
množina funkcí  $\mathcal{F}$ , které smíme použít k aproximaci (definovaných alespoň v bodech  $x_0, \dots, x_{n-1}$ ).

**Hledáme:**  $\varphi \in \mathcal{F}$  splňující  $\varphi(\vec{x}) = \vec{y}$ , tj.

$$\varphi(x_i) = y_i, \quad i = 0, \dots, n-1.$$

**Předpoklad:**  $\mathcal{F} = \text{Lin}\{\varphi_0, \dots, \varphi_{k-1}\} = \left\{ \sum_{j < k} c_j \varphi_j : c_j \in \mathbb{R} \right\}$ .

Ve vektorové formulaci:  $\varphi(\vec{x}) = \sum_{j < k} c_j \varphi_j(\vec{x}) = \vec{y}$ .

#### 1.2.1 Prostá interpolace

Dosazením dostáváme

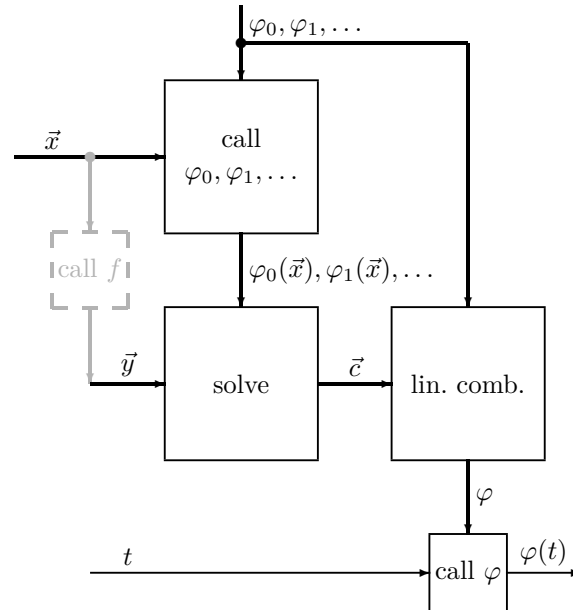
$$\sum_{j < k} c_j \varphi_j(x_i) = y_i, \quad i = 0, \dots, n-1,$$

což je soustava  $k$  lineárních rovnic pro neznámé  $c_0, \dots, c_{k-1}$ .

Pro jednoznačnost řešení potřebujeme  $k = n$  a navíc, aby aritmetické **vektory**  $\varphi_j(\vec{x}) = (\varphi_j(x_0), \dots, \varphi_j(x_{n-1}))$ ,  $j = 0, \dots, n-1$ , byly lineárně nezávislé. (To lze pokazit, i když **funkce**  $\varphi_0, \dots, \varphi_{n-1}$  jsou lineárně nezávislé. Záleží i na volbě uzlových bodů.)

Složitost výpočtu  $\propto n^3$ , u speciálních úloh dosáhneme menší.

## Blokové schéma obecné prosté interpolace



### 1.3 Interpolace polynomem

Interpolujeme polynomem stupně  $< n$ , neboli  $\leq n - 1$ .

Má právě  $n$  koeficientů, což potřebujeme pro existenci a jednoznačnost řešení.

Za bázi prostoru těchto polynomů můžeme volit  $\varphi_j(t) = t^j$ ,  $j = 0, \dots, n - 1$ .

Jiná báze může být výhodnější.

#### Výhody interpolace polynomem

- Jediné, co počítač umí rychle, jsou 4 základní aritmetické operace, tedy kromě polynomů už jen racionální funkce.
- Polynomy lze snadno integrovat, derivovat...
- Řešitelnost:

**Věta 1.1** Interpolační úlohu s  $n$  **různými** uzlovými body řeší právě jeden polynom stupně  $< n$ .

- Univerzálnost:

**Věta 1.2 (Weierstrassova)** Nechť  $f$  je spojitá funkce na uzavřeném intervalu  $I$  a nechť je dáno číslo  $\varepsilon > 0$ . Pak existuje polynom  $\varphi$  takový, že

$$\forall t \in I : |f(t) - \varphi(t)| \leq \varepsilon.$$

#### Nevýhody interpolace polynomem

- Weierstrassova věta neříká nic o potřebném stupni polynomu, takže výsledek nemusí být použitelný.
- Velmi málo skutečných závislostí je polynomiálních.

Lze řešit prostou interpolací, ale se složitostí  $\propto n^3$ ; jde to lépe.

## „Vodorovná linearita“ – interpolace lineárním polynomem

Dvěma body  $(x_i, y_i), (x_j, y_j)$  prokládáme lineární funkci  $\varphi$ . Její hodnota v bodě  $t$  je váženým průměrem  $y_i, y_j$ , kde váhy jsou nepřímo úměrné vzdálenostem  $t - x_i, x_j - t$ :

$$\begin{aligned}x_i &\mapsto \varphi(x_i) = y_i, \\x_j &\mapsto \varphi(x_j) = y_j, \\t &= \frac{\frac{x_i}{t-x_i} + \frac{x_j}{x_j-t}}{\frac{1}{t-x_i} + \frac{1}{x_j-t}} \mapsto \varphi(t) = \frac{\frac{y_i}{t-x_i} + \frac{y_j}{x_j-t}}{\frac{1}{t-x_i} + \frac{1}{x_j-t}}, \\t &= \frac{(x_j - t)x_i + (t - x_i)x_j}{x_j - x_i} \mapsto \varphi(t) = \frac{(x_j - t)y_i + (t - x_i)y_j}{x_j - x_i}.\end{aligned}\tag{1}$$

Odvození vyžadovalo  $t \notin \{x_i, x_j\}$ , ale výsledek platí vždy.

### 1.3.1 Lagrangeova konstrukce interpolačního polynomu

1. Pro  $j = 0, \dots, n-1$  vyřešíme speciální případ, kdy  $j$ -tá složka vektoru  $\vec{y}$  je jednotková, ostatní nulové; výsledkem je polynom  $\varrho_j$ ,

$$\varrho_j(x_i) = \delta_{ij} = \begin{cases} 1 & \text{pro } i = j, \\ 0 & \text{pro } i \neq j. \end{cases}$$

( $\delta_{ij}$  je tzv. **Kroneckerovo delta**)

Polynom  $\varrho_j$  stupně  $\leq n-1$  má  $n-1$  kořenů  $x_0, \dots, x_{j-1}, x_{j+1}, \dots, x_{n-1}$ ;

$$\begin{aligned}\varrho_j(t) &= a_j(t-x_0)\dots(t-x_{j-1})(t-x_{j+1})\dots(t-x_{n-1}) \\ &= a_j \prod_{i < n, i \neq j} (t-x_i),\end{aligned}$$

kde  $a_j$  určíme ze vztahu  $\varrho_j(x_j) = 1$ :

$$a_j = \frac{1}{\prod_{i < n, i \neq j} (x_j - x_i)}, \quad \varrho_j(t) = \frac{\prod_{i < n, i \neq j} (t - x_i)}{\prod_{i < n, i \neq j} (x_j - x_i)}.$$

2. Obecné řešení úlohy interpolace polynomem je lineární kombinace

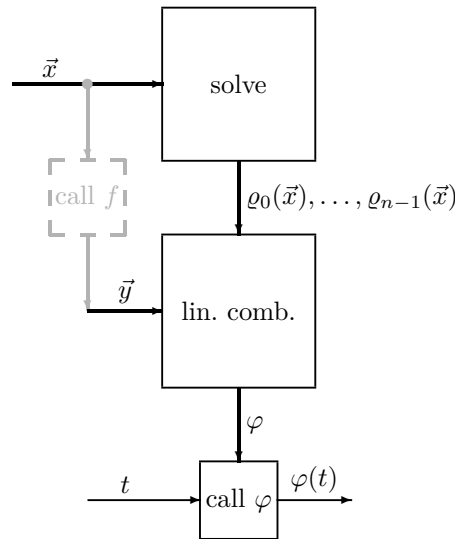
$$\begin{aligned}\varphi &= \sum_{j < n} y_j \varrho_j, \\ \varphi(t) &= \sum_{j < n} y_j \varrho_j(t) = \sum_{j < n} y_j \frac{\prod_{i < n, i \neq j} (t - x_i)}{\prod_{i < n, i \neq j} (x_j - x_i)}.\end{aligned}$$

Kontrola:

$$\varphi(x_i) = \sum_{j < n} y_j \varrho_j(x_i) = \sum_{j < n} y_j \delta_{ij} = y_i.$$

Myšlenka Lagrangeovy konstrukce interpolačního polynomu je použitelná i pro obecnější úlohy.

## Blokové schema Lagrangeovy konstrukce interpolačního polynomu



### „Svislá linearita“ aproximační úlohy (v nezávislé proměnné)

Většinou předpokládáme linearitu výstupu (princip superpozice), což zajistí nezávislost řešení na zvoleném lineárním měřítku.

Pak jakékoli řešení aproximační úlohy musí **lineárně** záviset na složkách aritmetického vektoru  $\vec{y} = (y_0, \dots, y_{n-1})$ . Vstupní vektor  $\vec{y} = (y_0, \dots, y_{n-1})$  je lineární kombinací vektorů **standardní báze**,

$$\vec{y} = y_0 \underbrace{(1, 0, \dots, 0)}_{\vec{e}_0} + y_1 \underbrace{(0, 1, 0, \dots, 0)}_{\vec{e}_1} + \dots + y_{n-1} \underbrace{(0, \dots, 0, 1)}_{\vec{e}_{n-1}} = \sum_{j < n} y_j \vec{e}_j.$$

Výsledná aproximace bude lineární kombinací (se stejnými koeficienty  $y_0, \dots, y_{n-1}$ ) funkcí  $\varrho_j$ , které řeší aproximační úlohu se vstupy  $\vec{e}_j$ ,

$$\begin{aligned} \vec{e}_0 &= (1, 0, 0, \dots, 0) && \mapsto \varrho_0, \\ \vec{e}_1 &= (0, 1, 0, \dots, 0) && \mapsto \varrho_1, \\ & && \dots \\ \vec{e}_{n-1} &= (0, 0, \dots, 0, 1) && \mapsto \varrho_{n-1}, \\ \vec{y} &= \sum_{j < n} y_j \vec{e}_j && \mapsto \sum_{j < n} y_j \varrho_j = \varphi. \end{aligned}$$

Funkce  $\varrho_0, \varrho_1, \dots, \varrho_{n-1}$  nám poskytují plnou informaci o řešení úlohy pro libovolné  $\vec{y} = (y_0, \dots, y_{n-1})$ .

**Poznámka:** Pro interpolaci jsou  $\vec{e}_0, \dots, \vec{e}_{n-1}$  vektory hodnot funkcí  $\varrho_0, \dots, \varrho_{n-1}$ ,

$$\vec{e}_j = (\varrho_j(x_0), \dots, \varrho_j(x_{n-1})).$$

Pro obecnou aproximaci to tak být nemusí, ale uvedené důsledky linearity zůstávají v platnosti.

### Nedostatky Lagrangeovy konstrukce interpolačního polynomu

Složitost  $\propto n^2$ , a to i pro výpočet jedné hodnoty, pokud výsledek neupravíme.

Při aproximaci vývoje na burze dostáváme průběžně nová data.

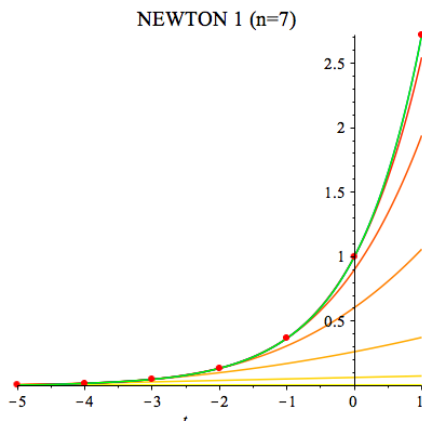
Musíme kvůli tomu počítat vše znova?

(Některé mezivýsledky lze použít, pokud jsme si je nezapomněli zaznamenat.)

Lze však vyjít z předchozího výsledku a ten jen opravit o určitý polynom:



### 1.3.2 Newtonova konstrukce interpolačního polynomu



Jedná se stále o stejný polynom.

Konstantní polynom  $d_0 = y_0$  má správnou hodnotu v uzlovém bodě  $x_0$ .

Hledaný polynom  $\varphi$  dostaneme přičtením vhodného polynomu  $\omega_0$  nulového v  $x_0$ :  $t \mapsto (t - x_0)\omega_0(t)$ , kde  $\omega_0$  je polynom stupně  $\leq n - 2$ :

$$\varphi(t) = d_0 + (t - x_0)\omega_0(t), \quad d_0 = y_0, \quad \omega_0(x_i) = \frac{y_i - d_0}{x_i - x_0}, \quad i > 0,$$

$$\omega_0(t) = d_1 + (t - x_1)\omega_1(t), \quad d_1 = \omega_0(x_1), \quad \omega_1(x_i) = \frac{\omega_0(x_i) - d_1}{x_i - x_1}, \quad i > 1,$$

$$\omega_1(t) = d_2 + (t - x_2)\omega_2(t), \quad d_2 = \omega_1(x_2), \quad \omega_2(x_i) = \frac{\omega_1(x_i) - d_2}{x_i - x_2}, \quad i > 2,$$

$$\omega_2(t) = d_3 + (t - x_3)\omega_3(t), \quad d_3 = \omega_2(x_3), \quad \omega_3(x_i) = \frac{\omega_2(x_i) - d_3}{x_i - x_3}, \quad i > 3,$$

...

$$\omega_{n-3}(t) = d_{n-2} + (t - x_{n-2})\omega_{n-2}(t), \quad d_{n-2} = \omega_{n-3}(x_{n-2}), \quad \omega_{n-2}(x_i) = \frac{\omega_{n-3}(x_i) - d_{n-2}}{x_i - x_{n-2}}, \quad i > n - 2,$$

$$\omega_{n-2}(t) = d_{n-1} = \omega_{n-2}(x_{n-1}).$$

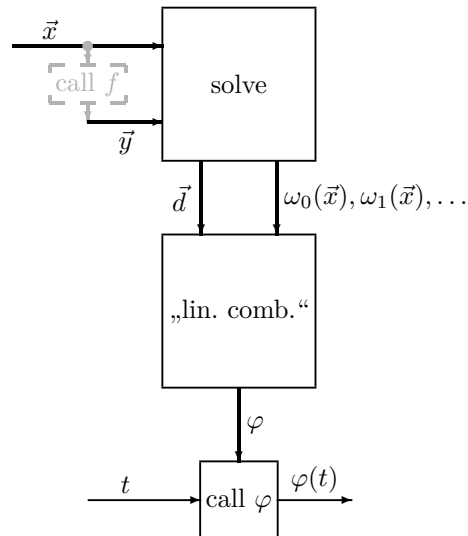
Zpětným dosazením dostaneme

$$\begin{aligned} \varphi(t) = & d_0 + (t - x_0) \cdot [d_1 + (t - x_1) \cdot [d_2 + \\ & + \dots (t - x_{n-3}) \cdot [d_{n-2} + (t - x_{n-2}) \cdot d_{n-1}] \dots]]. \end{aligned}$$

$$\begin{aligned} \varphi(t) = & d_0 + (t - x_0)d_1 + (t - x_0)(t - x_1)d_2 + \dots + d_{n-1} \prod_{i < n-1} (t - x_i) \\ = & \sum_{j < n} d_j \prod_{i < j} (t - x_i). \end{aligned}$$

Složitost výpočtu koeficientů  $\propto n^2$ , výpočtu jedné funkční hodnoty *podle prvního, neroznásobeného vzorce*  $\propto n$ .

## Blokové schema Newtonovy konstrukce interpolačního polynomu



### 1.3.3 Nevillův algoritmus

Nezájímají nás koeficienty, pouze hodnota interpolačního polynomu pro jediný argument  $t$ .

Jedním bodem  $(x_i, y_i)$  proložíme konstantní polynom  $y_i$ . Dva body  $(x_i, y_i)$ ,  $(x_{i+1}, y_{i+1})$  lineárně interpolujeme, hodnota v  $t$  je dle (1):

$$y_{i:i+1} = \frac{(x_{i+1} - t)y_i + (t - x_i)y_{i+1}}{x_{i+1} - x_i},$$

což je správně pro  $t \in \{x_i, x_{i+1}\}$ .

Rekurentní postup: Obecněji pro  $j < k$  označme  $y_{j:k}$  hodnotu v bodě  $t$  interpolačního polynomu proloženého body  $(x_j, y_j), \dots, (x_k, y_k)$ . Závisí na počáteční volbě  $t$ ; pro  $t \in \{x_j, \dots, x_k\}$  by byl  $y_{j:k}$  správný výsledek.

Speciálně:

$y_{i+1:i+m}$  by vyšlo správně pro  $t \in \{x_{i+1}, \dots, x_{i+m-1}\}$  a pro  $t = x_{i+m}$ ,

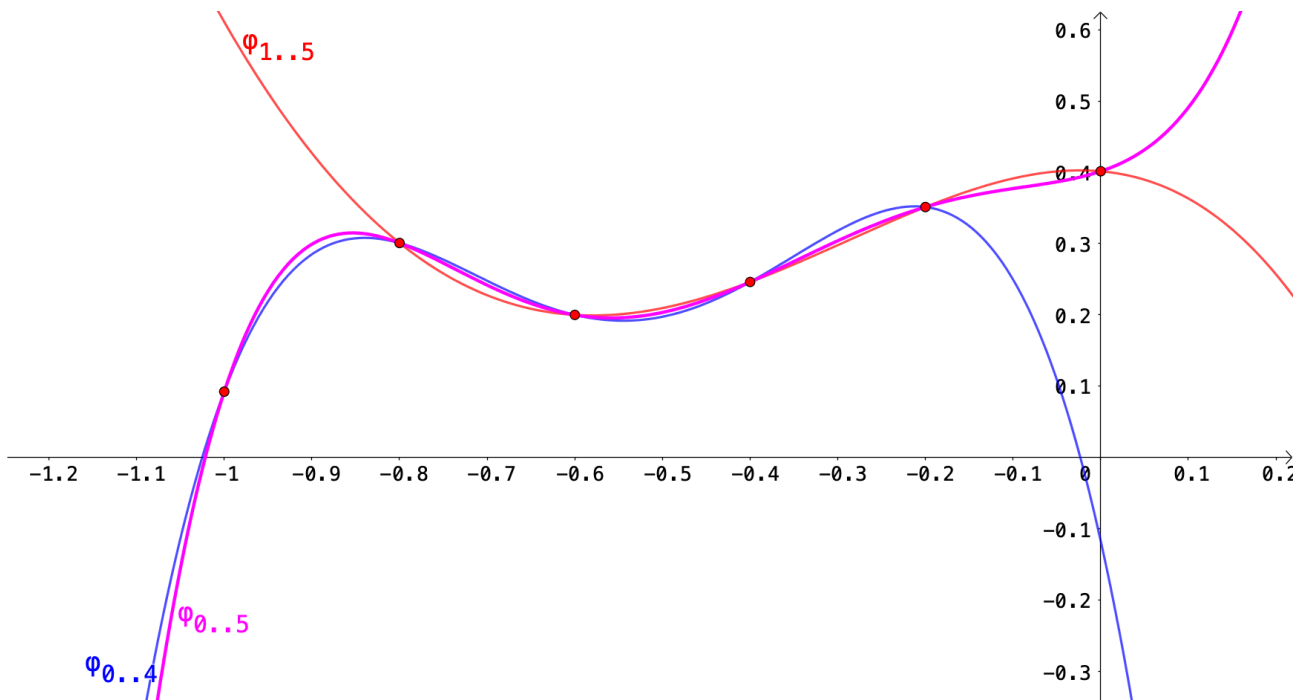
$y_{i:i+m-1}$  by vyšlo správně pro  $t \in \{x_{i+1}, \dots, x_{i+m-1}\}$  a pro  $t = x_i$ ,

$y_{i:i+m}$  by vyšlo správně pro  $t \in \{x_{i+1}, \dots, x_{i+m-1}\}$  a pro  $t = x_i, t = x_{i+m}$ ;

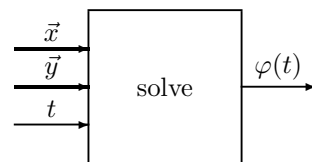
dostaneme ho lineární interpolací mezi  $x_i$  a  $x_{i+m}$  dle (1):

$$y_{i:i+m} = \frac{(x_{i+m} - t)y_{i:i+m-1} + (t - x_i)y_{i+1:i+m}}{x_{i+m} - x_i}.$$

Počítáme pro všechna  $i = 0, \dots, n-1-m$  (vnitřní cyklus);  $m = 1, \dots, n-1$  (vnější cyklus);  $y_{0:n-1}$  je výsledek. Složitost  $\propto n^2$ .



### Blokové schéma Nevillova algoritmu



Numerické chyby Nevillova algoritmu lze dále omezit, budeme-li místo hodnot interpolačních polynomů počítat s jejich rozdíly,

$$C_{i,m} = y_{i:i+m} - y_{i:i+m-1},$$

$$D_{i,m} = y_{i:i+m} - y_{i+1:i+m},$$

které lze počítat podle rekurentních vzorců

$$C_{i,m} = \frac{(t - x_i)(C_{i+1,m-1} - D_{i,m-1})}{x_{i+m} - x_i},$$

$$D_{i,m} = \frac{(t - x_{i+m})(C_{i+1,m-1} - D_{i,m-1})}{x_{i+m} - x_i}$$

a výsledek např. jako

$$y_{0:n-1} = y_0 + \sum_{m=1}^{n-1} C_{0,m}.$$

### 1.3.4 Chyba aproximace interpolačním polynomem

**Upravená úloha:** Aproximujeme funkci  $f$ ;

$\vec{x}$  je vektor uzlových bodů,

$\vec{y} = f(\vec{x})$  je vektor požadovaných hodnot v uzlových bodech.

Proložíme interpolační polynom  $\varphi$ .

Zajímá nás hodnota  $\varphi(u)$  v libovolném bodě  $u$  a její chyba  $f(u) - \varphi(u)$ .

## Odvození chyby aproximace interpolačním polynomem

Chyba metody  $f - \varphi$  je v uzlových bodech nulová, zkoumáme její hodnotu  $f(u) - \varphi(u)$  v bodě  $u \notin \{x_0, \dots, x_{n-1}\}$ . V duchu Newtonovy konstrukce najdeme interpolační polynom  $\psi_u$  stupně  $n$ , který správně interpoluje i v bodě  $u$ : k  $\varphi$  přičteme „vhodný násobek“ polynomu  $W$  stupně  $n$  s uzlovými body  $x_0, \dots, x_{n-1}$ ,

$$\psi_u(t) = \varphi(t) + P_u \underbrace{\prod_{i < n} (t - x_i)}_{W(t)}.$$

(Polynom  $\psi_u$  závisí na volbě bodu  $u$ .)

„Vhodný násobek“  $P_u$  splňuje  $\psi_u(u) = f(u)$ , takže provedená oprava v bodě  $u$  je právě chyba interpolace polynomem  $\varphi$ :

$$f(u) - \varphi(u) = \psi_u(u) - \varphi(u) = P_u W(u),$$

$$P_u = \frac{f(u) - \varphi(u)}{W(u)}.$$

Funkce  $f, \psi_u$  jsou si „blízké“, máme

$\geq n + 1$  kořenů  $u, x_0, \dots, x_{n-1}$  funkce  $f - \psi_u$ , mezi každými dvěma (aspoň jeden) kořen derivace,

$\geq n$  kořenů funkce  $(f - \psi_u)'$ ,

$\geq n - 1$  kořenů funkce  $(f - \psi_u)''$ ,

...

$\geq 1$  kořen funkce  $(f - \psi_u)^{(n)} = f^{(n)} - \psi_u^{(n)}$ , označme jej  $\xi_u$  (závisí na  $u$ ).

(Potřebovali jsme spojité derivace do řádu  $n$ .) Budeme potřebovat  $n$ -tou derivaci polynomu

$$\psi_u(t) = \varphi(t) + P_u \prod_{i < n} (t - x_i) = \underbrace{\varphi(t)}_{\text{polynom stupně } < n} + P_u (t^n + \underbrace{\dots}_{\text{polynom stupně } < n}),$$

$$\psi_u^{(n)}(t) = P_u (t^n)^{(n)} = P_u n!.$$

Využijeme

$$0 = f^{(n)}(\xi_u) - \psi_u^{(n)}(\xi_u) = f^{(n)}(\xi_u) - \frac{P_u}{W(u)} n!,$$

$$f(u) - \varphi(u) = \frac{f^{(n)}(\xi_u)}{n!} W(u),$$

$$|f(u) - \varphi(u)| = \frac{|f^{(n)}(\xi_u)|}{n!} |W(u)|,$$

kde  $|f^{(n)}(\xi_u)|$  nahradíme horním odhadem na uzavřeném intervalu  $I \supseteq \{u, x_0, \dots, x_{n-1}\}$ :  $M_n \geq \max_{t \in I} |f^{(n)}(t)|$ ,

$$|f(u) - \varphi(u)| \leq \frac{M_n}{n!} |W(u)|.$$

Pomocí horního odhadu  $\bar{w} \geq \max_{t \in I} |W(t)|$  dostaneme odhad chyby nezávislý na  $u$ :

$$|f(u) - \varphi(u)| \leq \frac{M_n}{n!} \bar{w}.$$

**Předpoklad:**  $f$  má spojité derivace do řádu  $n$  na uzavřeném intervalu  $I \supseteq \{u, x_0, \dots, x_{n-1}\}$ .

**Co můžeme ovlivnit**

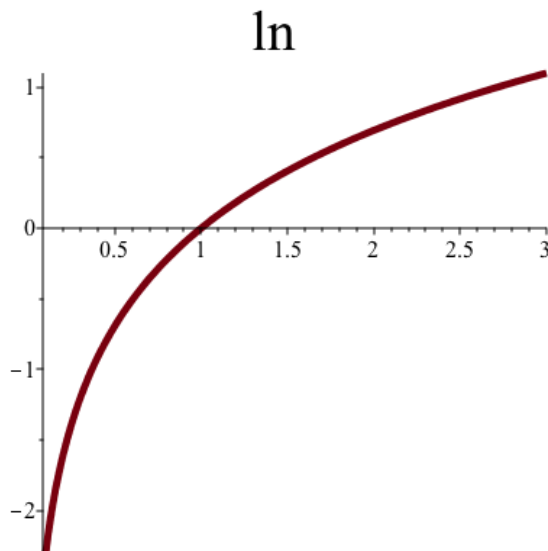
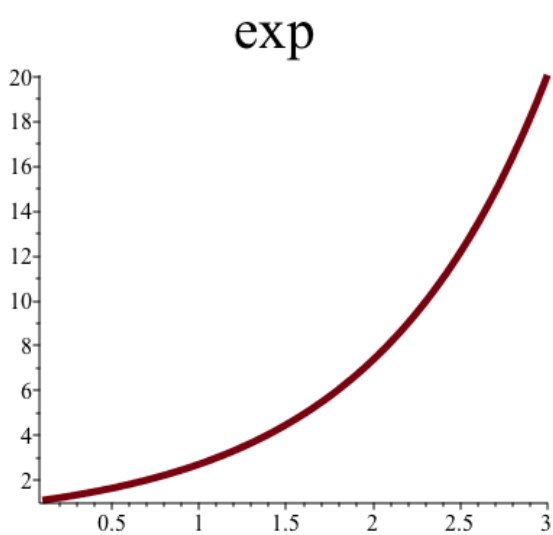
1. Počet uzlových bodů  $\implies$  koeficient  $\frac{1}{n!} \odot$ , ale i  $M_n \odot$ .

2. Interval  $I$ , ale musí obsahovat  $\{u, x_0, \dots, x_{n-1}\}$ , to lze jen pokazit, když  $u$  bude „daleko“  $\odot$ ; rozlišujeme:

- **extrapolaci**, kdy aproximujeme mimo interval  $\langle \min_i x_i, \max_i x_i \rangle$ ,
- **interpolaci** (v užším smyslu), kdy aproximujeme na intervalu  $\langle \min_i x_i, \max_i x_i \rangle$ .

3. Rozložení uzlových bodů  $\implies \bar{w} \ominus$ .

Příklad:



Ad 1. Počet uzlových bodů

	$f$	exp	ln
	$ f^{(n)}(u) $	$\exp(u)$	$\frac{(n-1)!}{u^n}$
	argmax	2	1/2
$n$	$\frac{M_n}{n!}$	$\frac{\exp(2)}{n!}$	$\frac{2^n}{n}$
2	$\frac{M_2}{2!}$	3.7	2
4	$\frac{M_4}{4!}$	0.31	4
8	$\frac{M_8}{8!}$	$1.8 \cdot 10^{-4}$	32
16	$\frac{M_{16}}{16!}$	$3.5 \cdot 10^{-13}$	4096
		$\ominus$	$\ominus$

$I = \langle 1/2, 2 \rangle$

Ad 2. Extrapolace

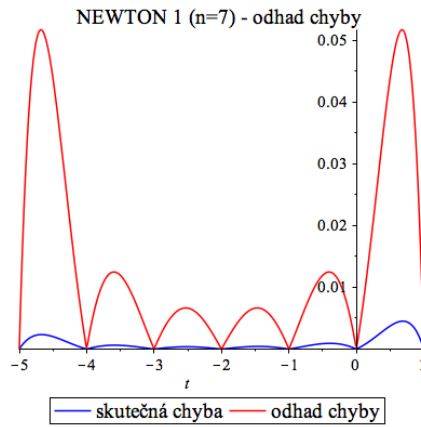
	$f$	exp	ln
	$ f^{(n)}(u) $	$\exp(u)$	$\frac{(n-1)!}{u^n}$
	argmax	3	1/10
$n$	$\frac{M_n}{n!}$	$\frac{\exp(3)}{n!}$	$\frac{10^n}{n}$
2	$\frac{M_2}{2!}$	10	50
4	$\frac{M_4}{4!}$	0.84	2500
8	$\frac{M_8}{8!}$	$5 \cdot 10^{-4}$	$1.25 \cdot 10^7$
16	$\frac{M_{16}}{16!}$	$10^{-12}$	$6.25 \cdot 10^{14}$
		☺	☹☹

$I = (1/10, 3)$

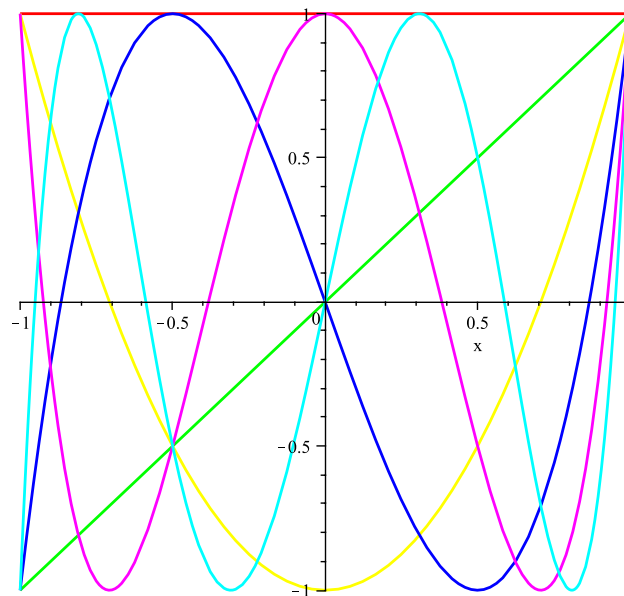
Zbývá:

Ad 3. Rozložení uzlových bodů

### Chyba aproximace interpolačním polynomem



### 1.3.5 Čebyševovy polynomy



$$\gamma_j(t) = \cos(j \arccos t), \quad j = 0, 1, 2, \dots$$

Počítají se z rekurentního vztahu

$$\begin{aligned} \gamma_0(t) &= 1 = \cos(0 \arccos t), \\ \gamma_1(t) &= t = \cos(1 \arccos t), \\ \gamma_j(t) &= 2t\gamma_{j-1}(t) - \gamma_{j-2}(t), \quad j \geq 2, \end{aligned}$$

$$\begin{aligned} \cos \alpha + \cos \beta &= 2 \cos \frac{\alpha - \beta}{2} \cos \frac{\alpha + \beta}{2}, \quad \alpha := j\theta, \quad \beta := (j-2)\theta, \\ \cos j\theta + \cos(j-2)\theta &= 2 \cos \theta \cos(j-1)\theta, \\ \cos j\theta &= 2 \underbrace{\cos \theta}_t \cos(j-1)\theta - \cos(j-2)\theta, \quad \theta := \arccos t, \\ \underbrace{\cos(j \arccos t)}_{\gamma_j(t)} &= 2t \underbrace{(\cos(j-1) \arccos t)}_{\gamma_{j-1}(t)} - \underbrace{\cos((j-2) \arccos t)}_{\gamma_{j-2}(t)}. \end{aligned}$$

Obor hodnot funkcí  $\gamma_j$ ,  $j \geq 1$ , na intervalu  $\langle -1, 1 \rangle$  je  $\langle -1, 1 \rangle$ .

Kořeny  $z_0, \dots, z_{j-1}$  jsou řešení rovnice

$$\begin{aligned} \cos(j \arccos z_k) &= 0, \\ j \arccos z_k &= \frac{\pi}{2} + k\pi, \\ \arccos z_k &= \frac{1}{j} \left( \frac{\pi}{2} + k\pi \right) \in \langle 0, \pi \rangle, \\ z_k &= \cos \left( \frac{\pi}{j} \left( k + \frac{1}{2} \right) \right), \quad k = 0, 1, \dots, j-1. \end{aligned}$$

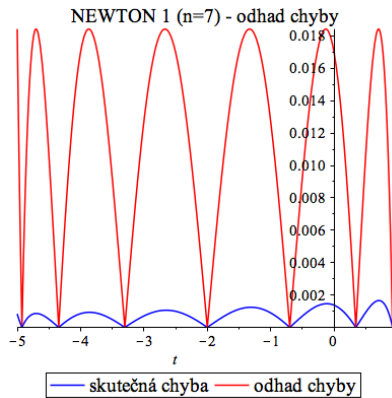
Pro  $j = n$  dostáváme doporučené **kosinové** rozdělení uzlových bodů na intervalu  $\langle -1, 1 \rangle$ ; obecně na  $\langle a, b \rangle$

$$x_k = \frac{b+a}{2} + \frac{b-a}{2} z_k = \frac{b+a}{2} + \frac{b-a}{2} \cos \frac{\pi(k + \frac{1}{2})}{n}, \quad k = 0, \dots, n-1.$$

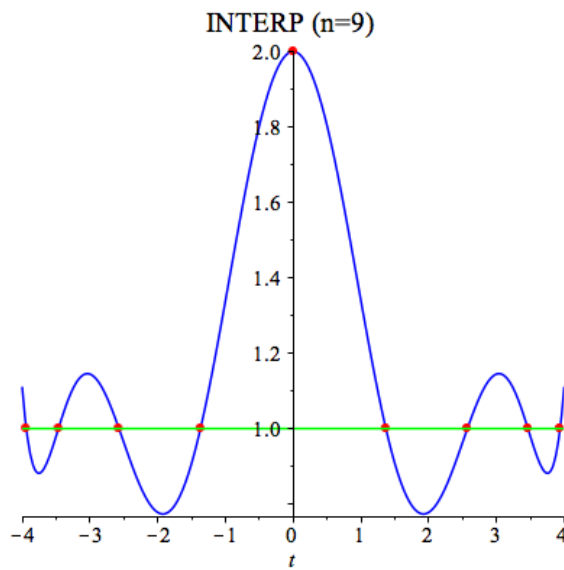
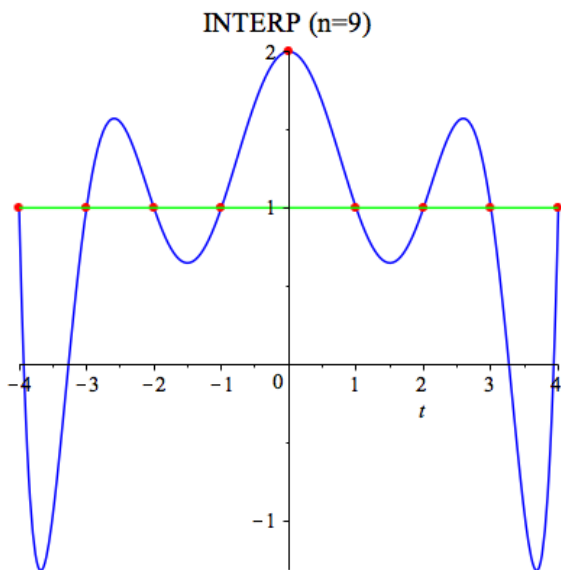
Pokud požadujeme, aby byly uzlové body v krajních bodech intervalu, můžeme vzorec modifikovat:

$$x_k = \frac{b+a}{2} + \frac{b-a}{2} \cos \frac{\pi k}{n-1}, \quad k = 0, \dots, n-1.$$

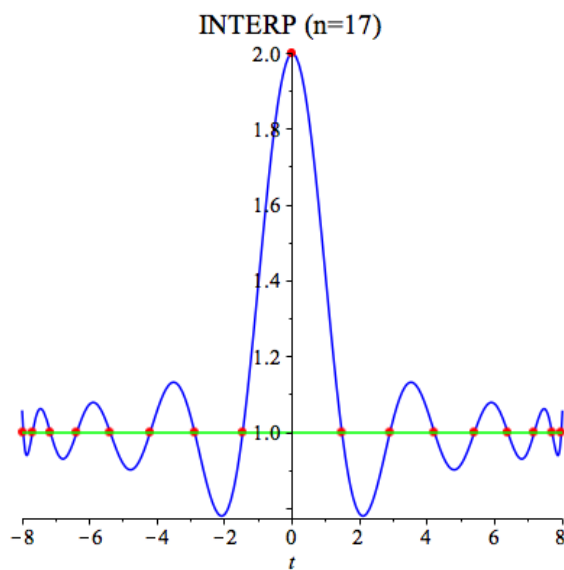
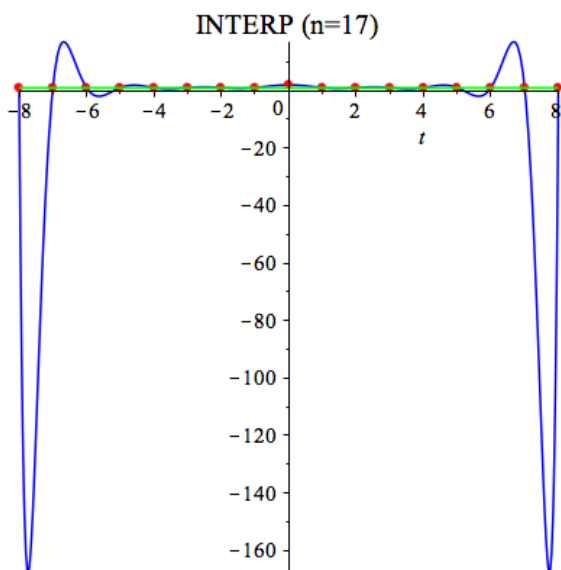
### Chyba aproximace při kosinovém rozdělení uzlových bodů



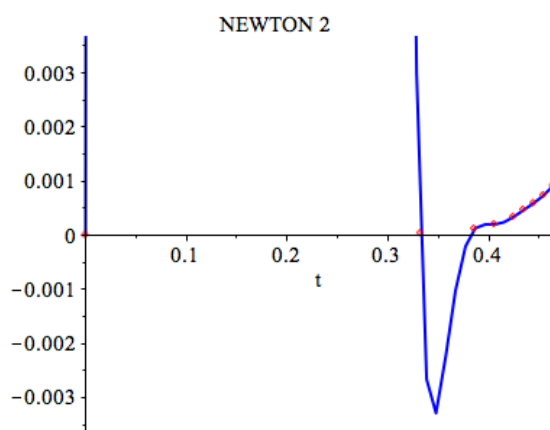
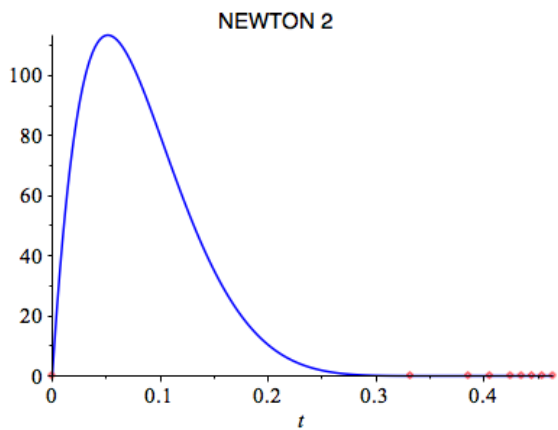
Vliv lokálních změn na interpolační polynom



Vliv lokálních změn na interpolační polynom

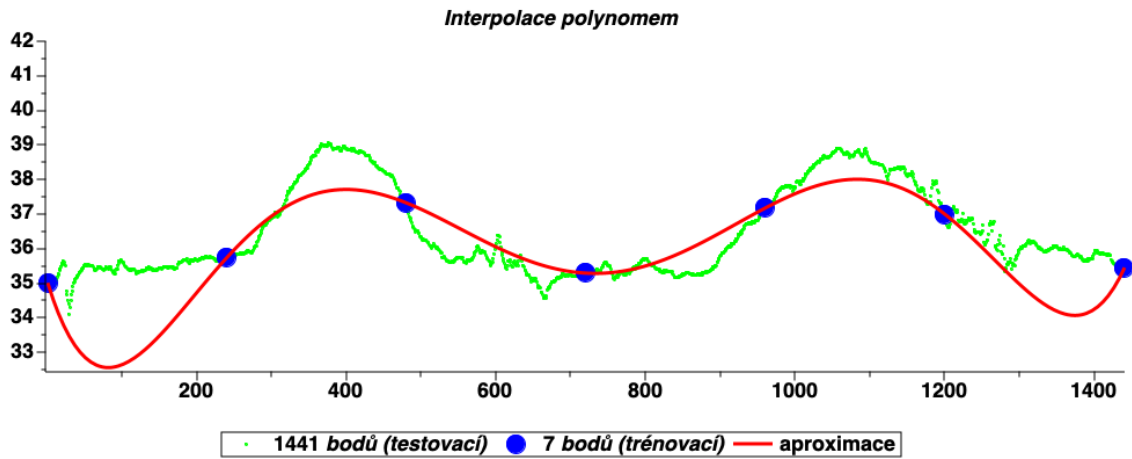


1.3.6 Příklad použití interpolačního polynomu na reálných datech

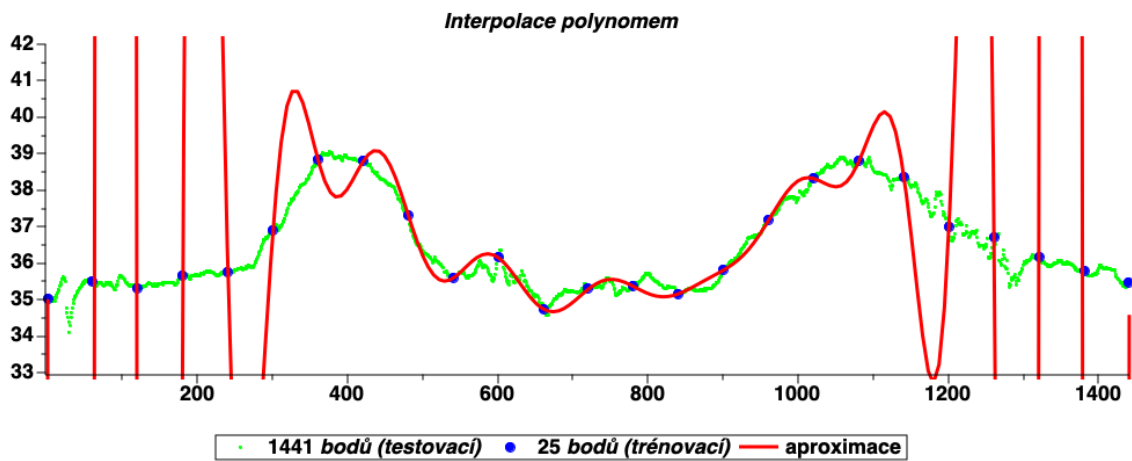




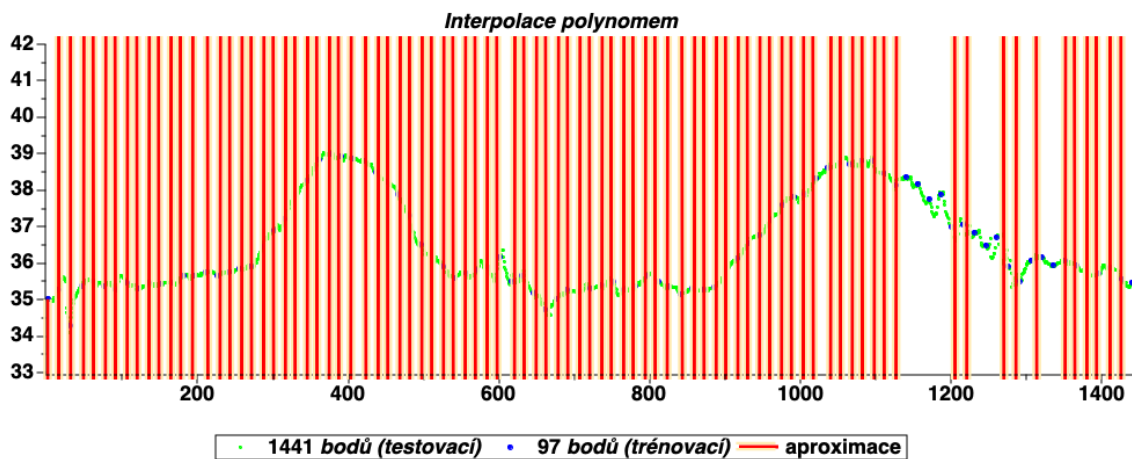
Vyjdeme z motivační úlohy (teplota pacienta) a vybrané hodnoty proložíme interpolačním polynomem.



$$\varphi(1440/\sqrt{2}) \doteq 37.74, \quad \varphi(1439.5) \doteq 35.38$$



$$\varphi(1440/\sqrt{2}) \doteq 38.33, \quad \varphi(1439.5) \doteq -4.71 \cdot 10^3 \quad \ominus \ominus$$



$$\varphi(1440/\sqrt{2}) \doteq 2.94 \cdot 10^{25}, \quad \varphi(1439.5) \doteq -2.34 \cdot 10^{49} \quad \ominus \ominus$$

Při použití všech 1440 bodů

$$\varphi(1440/\sqrt{2}) \doteq 1.48 \cdot 10^{1944}, \quad \varphi(1439.5) \doteq 7.23 \cdot 10^{2317} \quad \ominus \ominus \ominus$$

⇒ hrubá síla nic neřeší, vyhněme se použití polynomů velkého stupně.

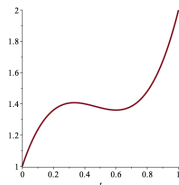
### 1.3.7 Hermitův interpolační polynom

**Příklad: Dáno:**  $x_0 = 0, \quad x_1 = 1; \quad y_{0,0} = 1, \quad y_{1,0} = 2, \quad y_{0,1} = 3, \quad y_{1,1} = 4.$

**Hledáme:** polynom  $\varphi(t) = c_0 + c_1t + c_2t^2 + c_3t^3$ , splňující

$$\begin{aligned} \varphi(x_0) &= y_{0,0}, & \varphi(x_1) &= y_{1,0}, & \varphi'(x_0) &= y_{0,1}, & \varphi'(x_1) &= y_{1,1}, \\ \varphi(0) &= 1, & \varphi(1) &= 2, & \varphi'(0) &= 3, & \varphi'(1) &= 4. \end{aligned}$$

$$\varphi(t) = 1 + 3t - 7t^2 + 5t^3.$$



**Jiný postup:** Stejně jako u Lagrangeovy konstrukce interpolačního polynomu můžeme sestavit nejdříve polynomy  $\eta, \varrho, \sigma, \tau$  stupně 3, splňující

$\psi$	$\psi(0)$	$\psi(1)$	$\psi'(0)$	$\psi'(1)$
$\eta$	1	0	0	0
$\varrho$	0	1	0	0
$\sigma$	0	0	1	0
$\tau$	0	0	0	1

Výsledek je jejich lineární kombinace se známými koeficienty:

$$\begin{aligned} \varphi &= y_{0,0} \eta + y_{1,0} \varrho + y_{0,1} \sigma + y_{1,1} \tau \\ &= 1\eta + 2\varrho + 3\sigma + 4\tau. \end{aligned}$$

Polynomy řešící dílčí úlohy:

$\eta(0)$	$\eta(1)$	$\eta'(0)$	$\eta'(1)$
1	0	0	0

$\eta$  má dvojnásobný kořen 1, je tedy tvaru

$$\eta(t) = (at + b)(t - 1)^2,$$

kde  $a, b$  určíme z hodnot v 0:

$$\begin{aligned} \eta(0) &= b = 1 \\ \eta'(0) &= a - 2b = 0 \end{aligned}$$

$$a = 2, \quad b = 1, \quad \eta(t) = (2t + 1)(t - 1)^2$$

$\varrho(0)$	$\varrho(1)$	$\varrho'(0)$	$\varrho'(1)$
0	1	0	0

$\varrho$  má dvojnásobný kořen 0, je tedy tvaru

$$\varrho(t) = (a^*t + b^*)t^2,$$

kde  $a^*, b^*$  určíme z hodnot v 1:

$$\begin{aligned} \varrho(1) &= a^* + b^* = 1 \\ \varrho'(1) &= 3a^* + 2b^* = 0 \end{aligned}$$

$$a^* = -2, \quad b^* = 3, \quad \varrho(t) = (-2t + 3)t^2$$

$\sigma(0)$	$\sigma(1)$	$\sigma'(0)$	$\sigma'(1)$
0	0	1	0

$\sigma$  má dvojnásobný kořen 1 a jednoduchý kořen 0, je tedy tvaru

$$\sigma(t) = ct(t-1)^2,$$

kde  $c$  určíme z derivace v 0:

$$\sigma'(0) = c = 1$$

$$\sigma(t) = t(t-1)^2$$

$\tau(0)$	$\tau(1)$	$\tau'(0)$	$\tau'(1)$
0	0	0	1

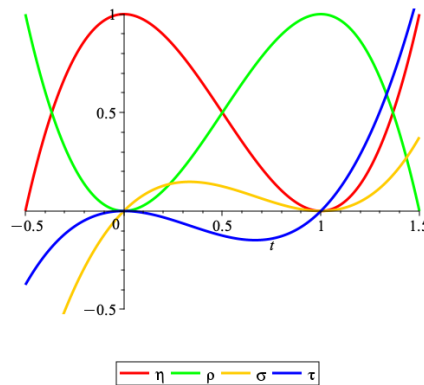
$\tau$  má dvojnásobný kořen 0 a jednoduchý kořen 1, je tedy tvaru

$$\tau(t) = c^* t^2 (t-1),$$

kde  $c^*$  určíme z derivace v 1:

$$\tau'(1) = c^* = 1$$

$$\tau(t) = t^2(t-1)$$



$\varphi$  dostaneme jako lineární kombinaci  $\eta, \varrho, \sigma, \tau$ ,

$$\varphi = y_{0,0} \eta + y_{1,0} \varrho + y_{0,1} \sigma + y_{1,1} \tau$$

### 1.3.8 Aproximace Taylorovou řadou

(přesněji Taylorovým polynomem)

Speciální případ Hermitova interpolačního polynomu s jediným uzlovým bodem, v němž je zadáno prvních  $n$  derivací (včetně nulté).

**Úloha: Dáno:**

Uzlový bod  $x_0$ ,

$n$  hodnot  $y_{0,0}, y_{0,1}, \dots, y_{0,n-1} \in \mathbb{R}$ .

**Hledáme:** polynom  $\varphi$  stupně  $< n$  takový, že

$$\varphi^{(j)}(x_0) = y_{0,j}, \quad j = 0, \dots, n-1.$$

Řešení je ve tvaru

$$\varphi = \sum_{j < n} y_{0,j} \varrho_j, \quad \varrho_j^{(i)}(x_0) = \delta_{ij}, \quad \varrho_j(t) = \frac{1}{j!} (t - x_0)^j,$$

$$\varphi(t) = \sum_{j < n} \frac{y_{0,j}}{j!} (t - x_0)^j .$$

Pokud jsou  $y_{0,0}, y_{0,1}, \dots, y_{0,n-1}$  hodnoty derivací nějaké funkce  $f$  v bodě  $x_0$ , tj.  $y_{0,j} = f^{(j)}(x_0)$ , pak  $\varphi$  je konečná Taylorova řada se středem  $x_0$ :

$$\varphi(t) = \sum_{j < n} \frac{f^{(j)}(x_0)}{j!} (t - x_0)^j .$$

Pokud  $f$  má na intervalu  $I(u, x_0)$  spojitou derivaci řádu  $n$ , pak

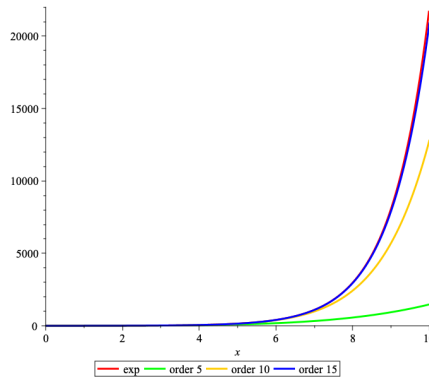
$$f(u) - \varphi(u) = \frac{f^{(n)}(\xi_u)}{n!} (u - x_0)^n ,$$

kde  $\xi_u \in I(u, x_0)$ ,

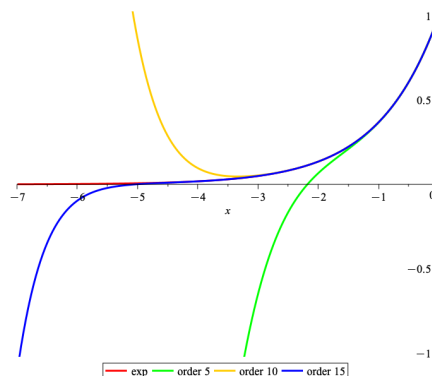
$$|f(u) - \varphi(u)| \leq \frac{M_n}{n!} |(u - x_0)^n| .$$

Jediný rozdíl od chyby interpolačního polynomu je, že polynom  $W(u)$  s kořeny  $x_0, \dots, x_{n-1}$  je nahrazen polynomem  $(u - x_0)^n$  (stejného stupně) s  $n$ -násobným kořenem  $x_0$ . Aproximace Taylorovou řadou je velmi přesná v okolí  $x_0$ , na úkor chyby ve vzdálenějších bodech.

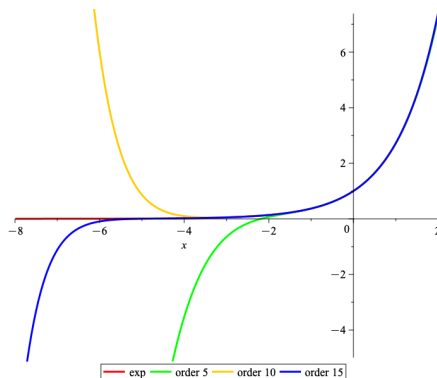
### Aproximace Taylorovým polynomem



Aproximace exponenciály (červeně) Taylorovými polynomy stupňů 5, 10, 15 pro kladné argumenty.



Aproximace exponenciály (červeně) Taylorovými polynomy stupňů 5, 10, 15 pro záporné argumenty.



Aproximace exponenciály (červeně) Taylorovými polynomy stupňů 5, 10, 15 pro kladné i záporné argumenty.

## 1.4 Interpolace splíny

Nevýhoda interpolace polynomem: malá změna vstupní hodnoty v jednom uzlovém bodě může zásadně ovlivnit výsledné hodnoty v místech značně vzdálených.

**Spline** je funkce po částech polynomiální. (Je dána různými polynomy nízkého stupně na jednotlivých intervalech.)

Nejjednodušším případem je náhrada po částech lineární funkcí, „lomenou čarou“, **lineární spline**.

### 1.4.1 Kubický spline

**Úloha: Dáno:**

vektor  $\vec{x} = (x_0, \dots, x_{n-1}) \in \mathbb{R}^n$  určující  $n$  uzlových bodů **vzestupně uspořádaných**,

vektor  $\vec{y} = (y_0, \dots, y_{n-1}) \in \mathbb{R}^n$  požadovaných hodnot v uzlových bodech,

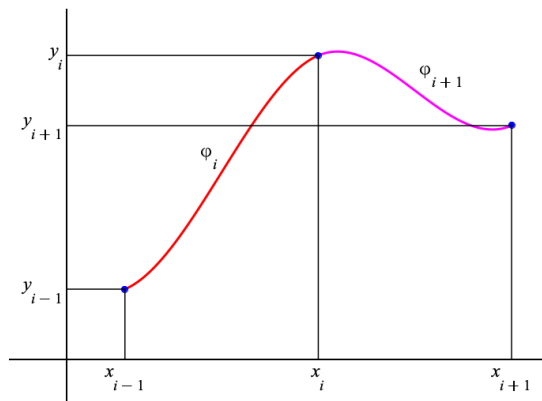
**Hledáme:** funkci  $\varphi$ , definovanou na intervalu  $\langle x_0, x_{n-1} \rangle$ , splňující:

- $\varphi(x_i) = y_i, \quad i = 0, \dots, n-1$ ,
- $\varphi$  se na intervalu  $\langle x_{i-1}, x_i \rangle$  shoduje s nějakým polynomem  $\varphi_i$  stupně nejvýše 3,  $i = 1, \dots, n-1$ ,
- $\varphi$  má na intervalu  $\langle x_0, x_{n-1} \rangle$  spojitou první a druhou derivaci.

(*Toto zadání bude nutné ještě upřesnit.*) Spojitost derivací stačí zajistit v bodech  $x_1, \dots, x_{n-2}$ :

$$\begin{aligned} \varphi'_i(x_i) &= \varphi'_{i+1}(x_i), & i = 1, \dots, n-2, \\ \varphi''_i(x_i) &= \varphi''_{i+1}(x_i), & i = 1, \dots, n-2. \end{aligned}$$

Předpokládejme, že známe hodnoty  $c_i = \varphi'(x_i) = \varphi'_i(x_i) = \varphi'_{i+1}(x_i)$ ,  $i = 0, \dots, n-1$ . Tím bude zajištěna spojitost  $\varphi'$ , zbývá zajistit spojitost  $\varphi''$ .



Polynomy  $\varphi_i$ ,  $i = 1, \dots, n-1$ , lze najít stejně jako Hermitův interpolační polynom v předchozí úloze, pouze uzlové body jsou  $x_{i-1}, x_i$  místo  $0, 1$ .

Obecný případ dostaneme lineární transformací  $u \in \langle 0, 1 \rangle$  na  $t \in \langle x_{i-1}, x_i \rangle$ :

$$t = x_{i-1} + (x_i - x_{i-1})u, \quad u = \frac{t - x_{i-1}}{x_i - x_{i-1}}.$$

Na druhých derivacích se to projeví např.

$$\begin{aligned} \eta_i(t) &= \eta(u), \\ \frac{d\eta_i}{dt} &= \frac{d\eta}{du} \cdot \frac{du}{dt}, & (\text{nekorektní Leibnizova notace}) \\ \eta_i'(t) &= \frac{\eta'(u)}{x_i - x_{i-1}}, \\ \eta_i''(t) &= \frac{\eta''(u)}{(x_i - x_{i-1})^2}. \end{aligned}$$

Na intervalu  $\langle x_{i-1}, x_i \rangle$ ,  $i = 1, \dots, n-1$ , dostáváme

$$\varphi_i(t) = y_{i-1} \eta_i(t) + y_i \varrho_i(t) + \overbrace{\varphi'(x_{i-1})}^{c_{i-1}} \sigma_i(t) + \overbrace{\varphi'(x_i)}^{c_i} \tau_i(t),$$

kde  $\eta_i, \varrho_i, \sigma_i, \tau_i$  jsou polynomy stupně nejvýše 3 (určené stejně jako polynomy  $\eta, \varrho, \sigma, \tau$  v úloze na Hermitův interpolační polynom).

Můžeme vypočítat jejich druhé derivace:

$$\begin{aligned} \eta''(0) &= -6, & \varrho''(0) &= 6, & \sigma''(0) &= -4, & \tau''(0) &= -2, \\ \eta_{i+1}''(x_i) &= \frac{-6}{(x_{i+1} - x_i)^2}, & \varrho_{i+1}''(x_i) &= \frac{6}{(x_{i+1} - x_i)^2}, & \sigma_{i+1}''(x_i) &= \frac{-4}{(x_{i+1} - x_i)^2}, & \tau_{i+1}''(x_i) &= \frac{-2}{(x_{i+1} - x_i)^2}, \\ \eta''(1) &= 6, & \varrho''(1) &= -6, & \sigma''(1) &= 2, & \tau''(1) &= 4, \\ \eta_i''(x_i) &= \frac{6}{(x_i - x_{i-1})^2}, & \varrho_i''(x_i) &= \frac{-6}{(x_i - x_{i-1})^2}, & \sigma_i''(x_i) &= \frac{2}{(x_i - x_{i-1})^2}, & \tau_i''(x_i) &= \frac{4}{(x_i - x_{i-1})^2}. \end{aligned}$$

Zbývá určit  $c_i = \varphi'(x_i)$ ; pro  $i = 1, \dots, n-2$  máme

$$\begin{aligned} & y_{i-1} \eta_i''(x_i) + y_i \varrho_i''(x_i) + \overbrace{\varphi'(x_{i-1})}^{c_{i-1}} \sigma_i''(x_i) + \overbrace{\varphi'(x_i)}^{c_i} \tau_i''(x_i) = \\ &= y_i \eta_{i+1}''(x_i) + y_{i+1} \varrho_{i+1}''(x_i) + \overbrace{\varphi'(x_i)}^{c_i} \sigma_{i+1}''(x_i) + \overbrace{\varphi'(x_{i+1})}^{c_{i+1}} \tau_{i+1}''(x_i), & (2) \\ & \frac{6y_{i-1} - 6y_i + 2\overbrace{\varphi'(x_{i-1})}^{c_{i-1}} + 4\overbrace{\varphi'(x_i)}^{c_i}}{(x_i - x_{i-1})^2} = \frac{-6y_i + 6y_{i+1} - 4\overbrace{\varphi'(x_i)}^{c_i} - 2\overbrace{\varphi'(x_{i+1})}^{c_{i+1}}}{(x_{i+1} - x_i)^2}, \\ & \frac{2}{(x_i - x_{i-1})^2} \overbrace{\varphi'(x_{i-1})}^{c_{i-1}} + \frac{4}{(x_i - x_{i-1})^2} \overbrace{\varphi'(x_i)}^{c_i} + \\ & + \frac{4}{(x_{i+1} - x_i)^2} \overbrace{\varphi'(x_i)}^{c_i} + \frac{2}{(x_{i+1} - x_i)^2} \overbrace{\varphi'(x_{i+1})}^{c_{i+1}} = \frac{-6y_{i-1} + 6y_i}{(x_i - x_{i-1})^2} + \frac{-6y_i + 6y_{i+1}}{(x_{i+1} - x_i)^2}, \end{aligned}$$

to je  $n-2$  lineárních rovnic o  $n$  neznámých.

Zbývají 2 volitelné parametry. Obvykle se volí  $\varphi''(x_0) = \varphi''(x_{n-1}) = 0$ , tzv. **přirozený spline** (angl. *natural spline*). To znamená, že v (2) pro  $i = 0$ , resp.  $n-1$ , nahradíme (nedefinovanou) levou, resp. pravou, stranu

nulou.

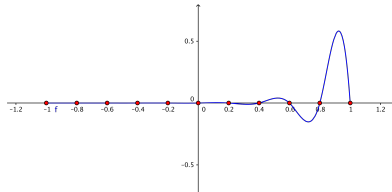
Další možnosti, které v současnosti podporuje Maple:

$$\varphi'(x_0) = \varphi'(x_{n-1}), \quad \varphi''(x_0) = \varphi''(x_{n-1}), \quad (\text{endpoints}='periodic'),$$

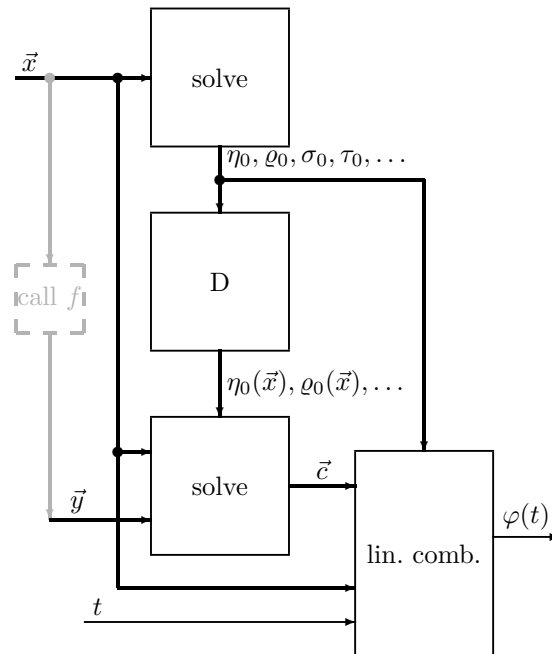
spojitost 3. derivace v  $x_1, x_{n-2}$  (`endpoints='notaknot'`),

nebo explicitní zadání první derivace v krajních bodech, popř. jakýchkoli podobných podmínek v požadovaném počtu.

Tyto volby se projeví pouze na okrajích intervalu.



### Blokové schéma interpolace spliny



Výpočet má dvě části:

1. Výpočet koeficientů  $c_i = \varphi'(x_i)$ ,  $i = 0, \dots, n - 1$  (složitost nejvýše  $\propto n^3$ ).
2. Výpočet funkční hodnoty (složitost  $\propto \ln n$ ).

Matice soustavy je **třídiagonální** a koeficienty na diagonále jsou největší z řádku.

Jsou-li navíc uzlové body *ekvidistantní*, je matice soustavy *symetrická*.

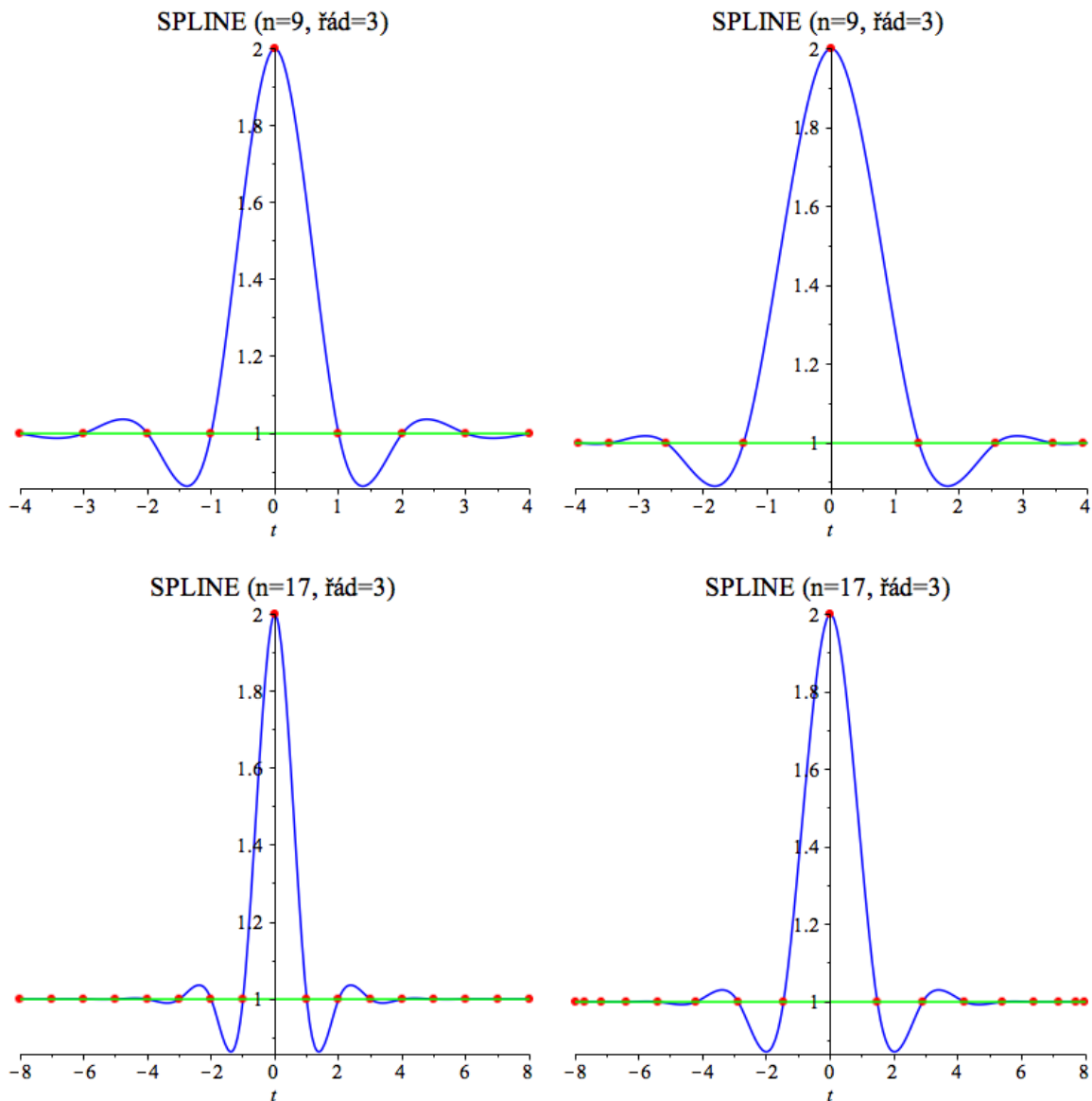
(Obojí lze využít pro efektivnější řešení.)

**Spliny nelze extrapolovat!** ☹

*Přesto to většina implementací dovoluje.*

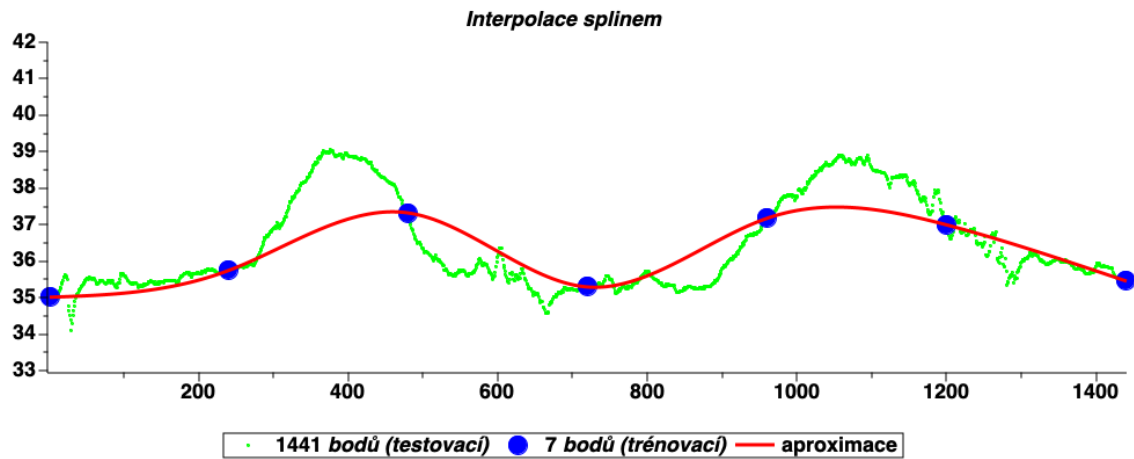
**Poznámka:** Volba rozdělení má v případě interpolace splinem výrazně menší vliv.

### Vliv lokálních změn na spline



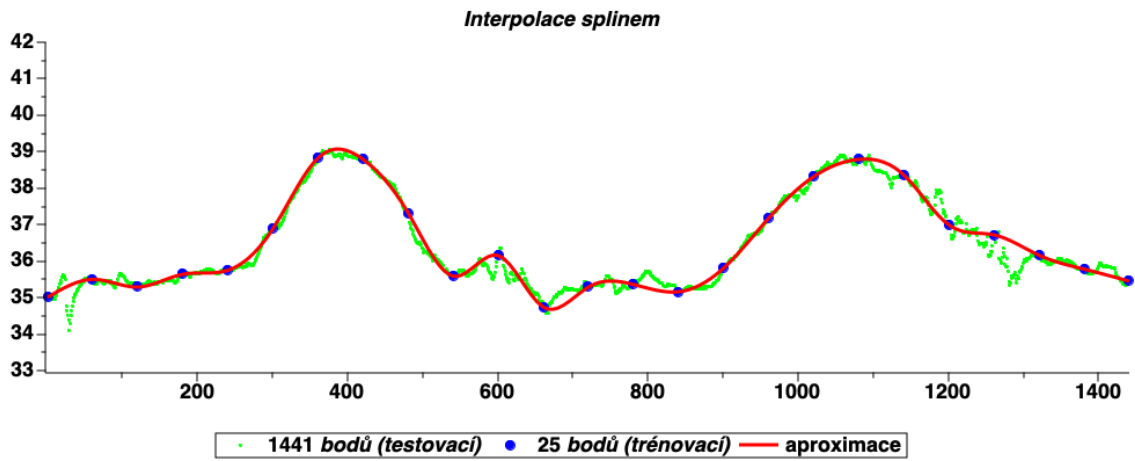
### 1.4.2 Příklad použití splinu na reálných datech

Vyjdeme z motivační úlohy (teplota pacienta) a vybrané hodnoty proložíme splinem.

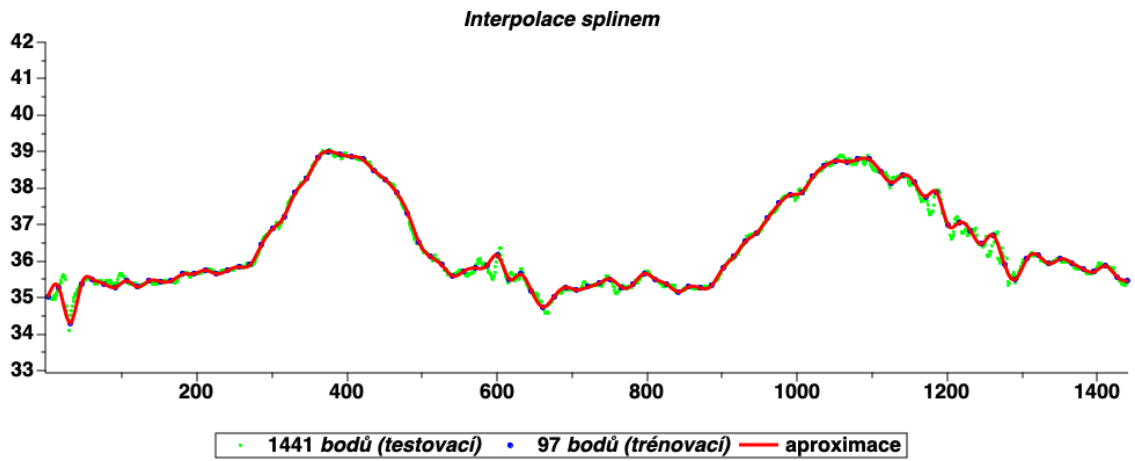




$$\varphi(1440/\sqrt{2}) \doteq 37.44, \quad \varphi(1439.5) \doteq 35.46$$



$$\varphi(1440/\sqrt{2}) \doteq 38.28, \quad \varphi(1439.5) \doteq 35.46$$

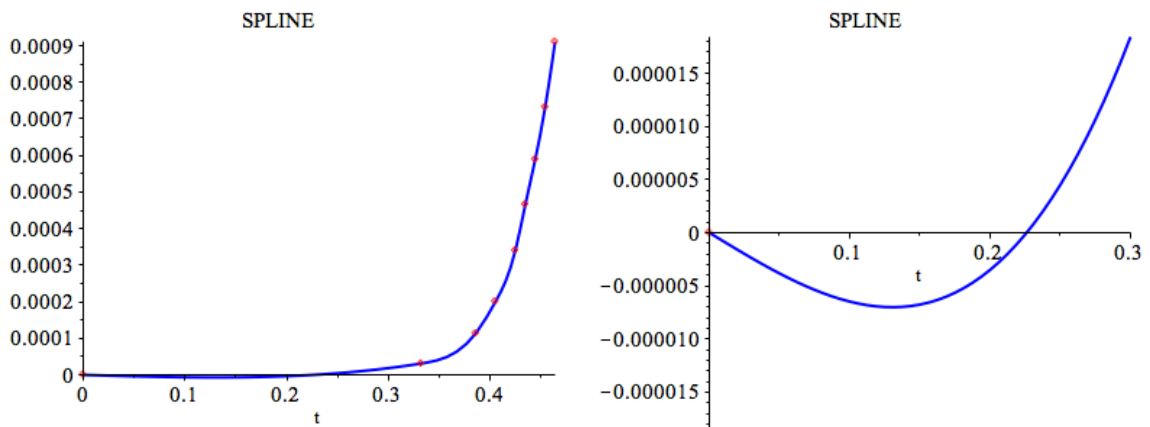


$$\varphi(1440/\sqrt{2}) \doteq 38.24, \quad \varphi(1439.5) \doteq 35.45$$

Při použití všech 1441 bodů

$$\varphi(1440/\sqrt{2}) \doteq 38.25, \quad \varphi(1439.5) \doteq 35.42$$

V-A charakteristika diody



## 1.5 Metoda nejmenších čtverců

**Úloha: Dáno:**

vektor  $\vec{x} = (x_0, \dots, x_{n-1}) \in \mathbb{R}^n$  určující  $n$  uzlových bodů (**ne nutně různých**),

vektor  $\vec{y} = (y_0, \dots, y_{n-1}) \in \mathbb{R}^n$  **žádoucích** hodnot v uzlových bodech,

$k$  funkcí  $\varphi_0, \dots, \varphi_{k-1}$ ,  $k \leq n$ , definovaných alespoň ve všech uzlových bodech.

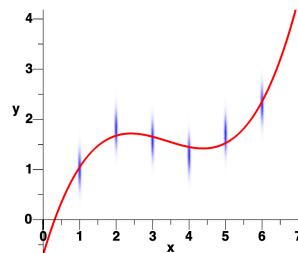
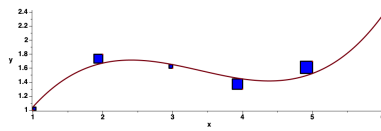
**Hledáme:** koeficienty  $c_0, \dots, c_{k-1} \in \mathbb{R}$  lineární kombinace funkcí  $\varphi_j$

$$\varphi = \sum_{j < k} c_j \varphi_j$$

takové, abychom minimalizovali výraz

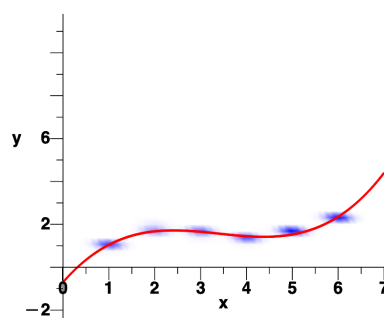
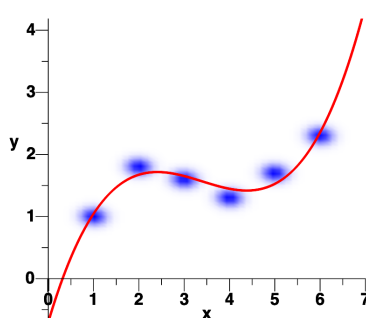
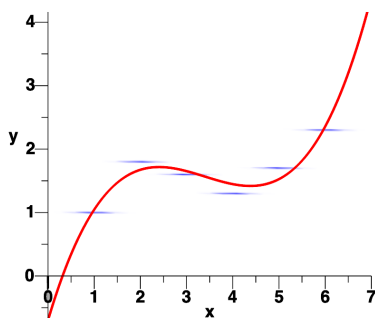
$$H_2 = \sum_{i < n} (\varphi(x_i) - y_i)^2 = \sum_{i < n} \left( \sum_{j < k} c_j \varphi_j(x_i) - y_i \right)^2,$$

tj. kvadrát svislé vzdálenosti od daných bodů.



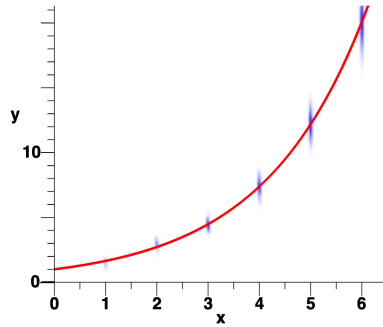
To odpovídá maximálně věrohodnému odhadu, pokud chyby ve svislém směru mají stejná nezávislá normální rozdělení.

**Modifikovaná kritéria**



Vodorovná vzdálenost má problém, šikmá ani nemusí dávat smysl.

Různé váhy:  $H_{2w} = \sum_{i < n} w_i (\varphi(x_i) - y_i)^2$



Řeší se analogicky (Maple podporuje),

$$H_1 = \sum_{i < n} |\varphi(x_i) - y_i|$$

nevede na jednoznačné řešení, neuzívá se,

$$H_0 = \max_{i < n} |\varphi(x_i) - y_i|$$

řeší se (tzv. Čebyševova aproximace), ale je obtížnější.

### 1.5.1 Řešení aproximace podle kritéria nejmenších čtverců

V  $\mathbb{R}^n$  máme skalární součin vektorů  $\vec{u} = (u_0, \dots, u_{n-1})$ ,  $\vec{v} = (v_0, \dots, v_{n-1})$ :

$$\vec{u} \cdot \vec{v} = \sum_{i < n} u_i \cdot v_i.$$

Máme aproximovat vektor  $\vec{y}$  lineární kombinací  $\varphi(\vec{x}) = \sum_{j < k} c_j \varphi_j(\vec{x})$ ,

minimalizujeme kritérium  $H_2 = (\varphi(\vec{x}) - \vec{y}) \cdot (\varphi(\vec{x}) - \vec{y}) = \|\varphi(\vec{x}) - \vec{y}\|^2$ .

**Řešení:** Kolmý průmět splňuje soustavu podmínek (pro  $m = 0, \dots, k-1$ )

$$\begin{aligned} (\varphi(\vec{x}) - \vec{y}) &\perp \varphi_m(\vec{x}), \\ (\varphi(\vec{x}) - \vec{y}) \cdot \varphi_m(\vec{x}) &= 0, \\ \varphi(\vec{x}) \cdot \varphi_m(\vec{x}) &= \vec{y} \cdot \varphi_m(\vec{x}). \end{aligned}$$

$\varphi(\vec{x})$  se vzhledem ke skalárním součinům s vektory  $\varphi_m(\vec{x})$  chová stejně jako  $\vec{y}$ .

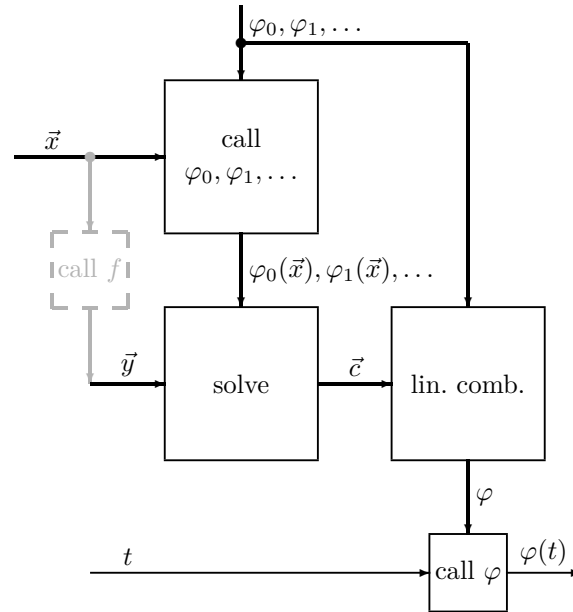
$$\begin{aligned} \left( \sum_{j < k} c_j \varphi_j(\vec{x}) \right) \cdot \varphi_m(\vec{x}) &= \vec{y} \cdot \varphi_m(\vec{x}), \\ \sum_{j < k} c_j (\varphi_j(\vec{x}) \cdot \varphi_m(\vec{x})) &= \vec{y} \cdot \varphi_m(\vec{x}), \quad m = 0, \dots, k-1; \end{aligned}$$

soustava lineárních rovnic pro neznámé  $c_0, \dots, c_{k-1}$  (**soustava normálních rovnic**).

**Speciální případ:** aproximujeme polynomem stupně  $< k$ , můžeme volit  $\varphi_j(t) = t^j$ ,

$$\varphi_j(\vec{x}) \cdot \varphi_m(\vec{x}) = \sum_{i < n} x_i^j \cdot x_i^m = \sum_{i < n} x_i^{j+m}.$$

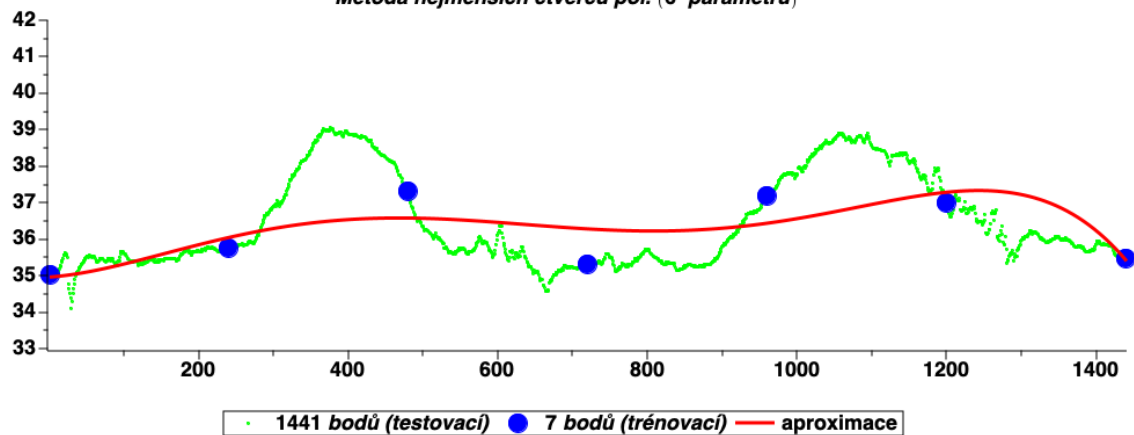
## Blokové schéma obecné aproximace metodou nejmenších čtverců



### Příklad použití metody nejmenších čtverců na reálných datech

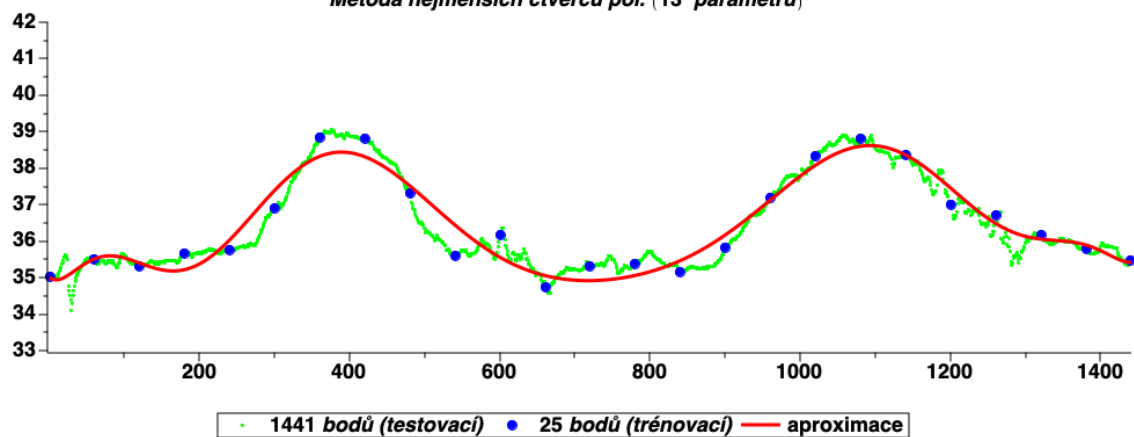
Vyjdeme z motivační úlohy (teplota pacienta) a vybrané hodnoty proložíme polynomem. (Čísla v závorkách udávají počet použitých aproximačních funkcí.)

**Metoda nejmenších čtverců pol. (6 parametrů)**



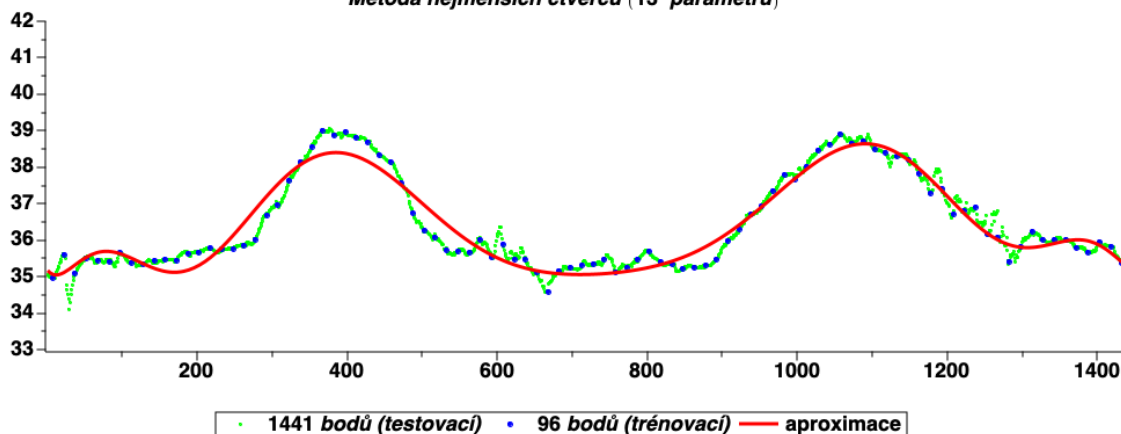
$$\varphi(1440/\sqrt{2}) \doteq 36.63, \quad \varphi(1439.5) \doteq 35.44$$

**Metoda nejmenších čtverců pol. (13 parametrů)**



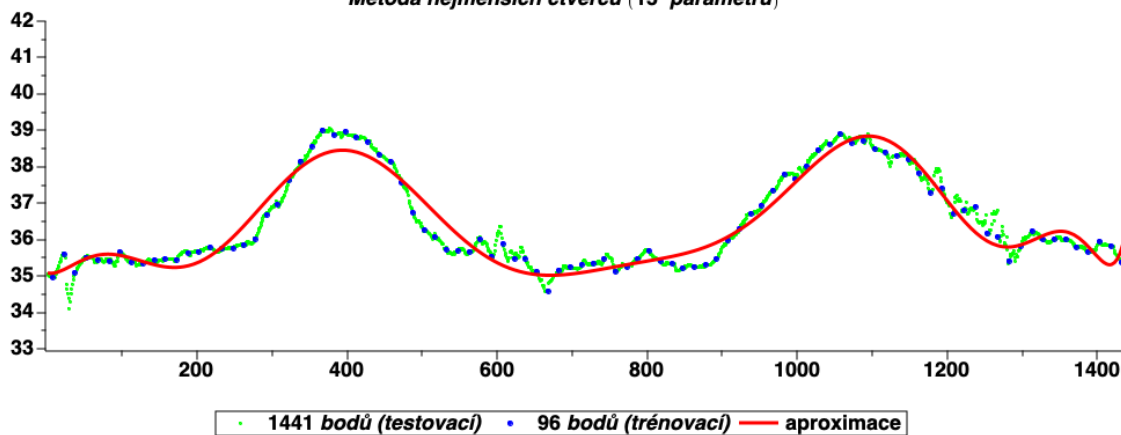
$$\varphi(1440/\sqrt{2}) \doteq 38.05, \quad \varphi(1439.5) \doteq 35.43$$

**Metoda nejmenších čtverců (13 parametrů)**

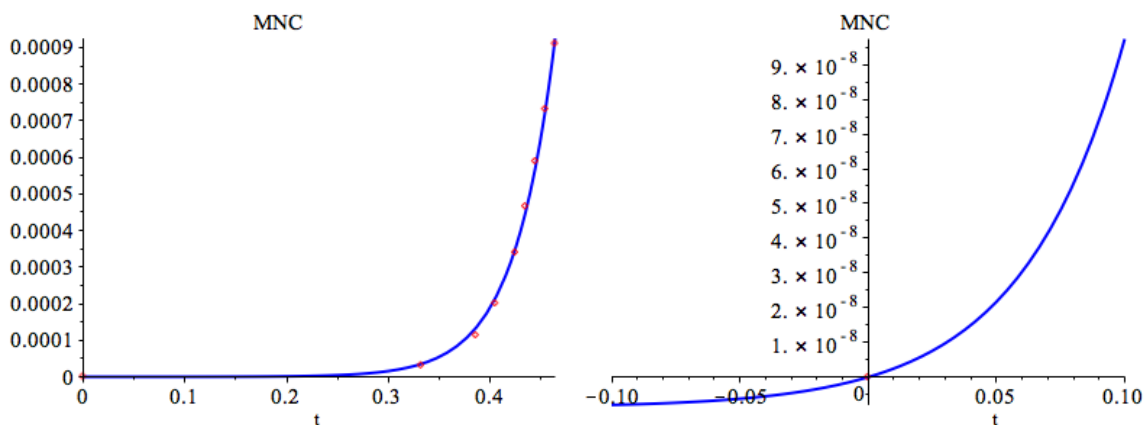


$$\varphi(1440/\sqrt{2}) \doteq 38.03, \quad \varphi(1439.5) \doteq 35.30$$

**Metoda nejmenších čtverců (15 parametrů)**



V-A charakteristika diody



Aproximace V-A charakteristiky diody lineární kombinací konstanty a dvou exponenciál s vhodnými základy.

### 1.5.2 Ortogonalizace

V podprostoru  $P = \text{Lin}\{\varphi_0(\vec{x}), \dots, \varphi_{k-1}(\vec{x})\}$  najdeme ortogonální bázi  $(\psi_0(\vec{x}), \dots, \psi_{k-1}(\vec{x}))$ ,

$$\psi_j(\vec{x}) \cdot \psi_m(\vec{x}) = 0 \text{ pro } j \neq m.$$

Ortogonalita závisí nejen na funkcích  $\varphi_0, \dots, \varphi_{k-1}$ , ale i na volbě uzlových bodů!

Hledáme řešení ve tvaru  $\varphi = \sum_{j < k} d_j \psi_j$ , kde  $d_j, j = 0, \dots, k-1$  jsou souřadnice vzhledem k nové bázi. Matice soustavy normálních rovnic je diagonální:

$$\begin{aligned} d_j (\psi_j(\vec{x}) \cdot \psi_j(\vec{x})) &= \vec{y} \cdot \psi_j(\vec{x}), & j = 0, \dots, k-1, \\ d_j = \frac{\vec{y} \cdot \psi_j(\vec{x})}{\psi_j(\vec{x}) \cdot \psi_j(\vec{x})} &= \frac{\vec{y} \cdot \psi_j(\vec{x})}{\|\psi_j(\vec{x})\|^2}, & j = 0, \dots, k-1. \end{aligned}$$

Navíc lze volit vektory  $\psi_j(\vec{x})$  jednotkové, pak vyjde jednotkový i jmenovatel.

### Gramova-Schmidtova ortogonalizace

Z báze  $(\vec{\varphi}_0, \dots, \vec{\varphi}_{k-1})$  vytvoříme ortogonální bázi  $(\psi_0(\vec{x}), \dots, \psi_{k-1}(\vec{x}))$ :

$$\begin{aligned} \psi_0(\vec{x}) &= \varphi_0(\vec{x}), \\ \psi_1(\vec{x}) &= \varphi_1(\vec{x}) + \alpha_{1,0} \psi_0(\vec{x}), \end{aligned}$$

$$\begin{aligned} \psi_1(\vec{x}) \cdot \psi_0(\vec{x}) = 0 &\Rightarrow \varphi_1(\vec{x}) \cdot \psi_0(\vec{x}) + \alpha_{1,0} \psi_0(\vec{x}) \cdot \psi_0(\vec{x}) = 0 \\ &\Rightarrow \alpha_{1,0} = \frac{-\varphi_1(\vec{x}) \cdot \psi_0(\vec{x})}{\psi_0(\vec{x}) \cdot \psi_0(\vec{x})} \end{aligned}$$

$$\psi_2(\vec{x}) = \varphi_2(\vec{x}) + \alpha_{2,0} \psi_0(\vec{x}) + \alpha_{2,1} \psi_1(\vec{x})$$

$$\begin{aligned} \psi_2(\vec{x}) \cdot \psi_0(\vec{x}) = 0 &\Rightarrow \varphi_2(\vec{x}) \cdot \psi_0(\vec{x}) + \alpha_{2,0} \psi_0(\vec{x}) \cdot \psi_0(\vec{x}) + \alpha_{2,1} \underbrace{\psi_1(\vec{x}) \cdot \psi_0(\vec{x})}_0 = 0 \\ &\Rightarrow \alpha_{2,0} = \frac{-\varphi_2(\vec{x}) \cdot \psi_0(\vec{x})}{\psi_0(\vec{x}) \cdot \psi_0(\vec{x})} \end{aligned}$$

$$\begin{aligned} \psi_2(\vec{x}) \cdot \psi_1(\vec{x}) = 0 &\Rightarrow \varphi_2(\vec{x}) \cdot \psi_1(\vec{x}) + \alpha_{2,0} \underbrace{\psi_0(\vec{x}) \cdot \psi_1(\vec{x})}_0 + \alpha_{2,1} \psi_1(\vec{x}) \cdot \psi_1(\vec{x}) = 0 \\ &\Rightarrow \alpha_{2,1} = \frac{-\varphi_2(\vec{x}) \cdot \psi_1(\vec{x})}{\psi_1(\vec{x}) \cdot \psi_1(\vec{x})} \end{aligned}$$

...

$$\psi_j(\vec{x}) = \varphi_j(\vec{x}) + \sum_{m < j} \alpha_{j,m} \psi_m(\vec{x})$$

$$\begin{aligned} \forall p, p < j: \quad \psi_j(\vec{x}) \cdot \psi_p(\vec{x}) &= 0 = \varphi_j(\vec{x}) \cdot \psi_p(\vec{x}) + \sum_{m < j} \alpha_{j,m} \psi_m(\vec{x}) \cdot \psi_p(\vec{x}) \\ &= \varphi_j(\vec{x}) \cdot \psi_p(\vec{x}) + \alpha_{j,p} \psi_p(\vec{x}) \cdot \psi_p(\vec{x}) \\ \Rightarrow \alpha_{j,p} &= \frac{-\varphi_j(\vec{x}) \cdot \psi_p(\vec{x})}{\psi_p(\vec{x}) \cdot \psi_p(\vec{x})} \end{aligned}$$

$$\begin{pmatrix} \psi_0(\vec{x}) \\ \psi_1(\vec{x}) \\ \psi_2(\vec{x}) \\ \psi_3(\vec{x}) \\ \vdots \\ \psi_{k-1}(\vec{x}) \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & \cdots & 0 \\ \alpha_{1,0} & 1 & 0 & 0 & \cdots & 0 \\ \alpha_{2,0} & \alpha_{2,1} & 1 & 0 & \cdots & 0 \\ \alpha_{3,0} & \alpha_{3,1} & \alpha_{3,2} & 1 & & 0 \\ \vdots & \vdots & \vdots & & \ddots & \vdots \\ \alpha_{k-1,0} & \alpha_{k-1,1} & \alpha_{k-1,2} & \cdots & \cdots & 1 \end{pmatrix} \begin{pmatrix} \vec{\varphi}_0 \\ \vec{\varphi}_1 \\ \vec{\varphi}_2 \\ \vec{\varphi}_3 \\ \vdots \\ \vec{\varphi}_{k-1} \end{pmatrix}$$

(Komu vadí maticový zápis pro vektory, může je nahradit řádky matice.)

### 1.5.3 Aproximace goniometrickým polynomem

(konečnou Fourierovou řadou)

Aproximace metodou nejmenších čtverců, přičemž aproximační funkce jsou

$$1, \quad \cos 2\pi \frac{t}{T}, \quad \sin 2\pi \frac{t}{T}, \quad \cos 4\pi \frac{t}{T}, \quad \sin 4\pi \frac{t}{T}, \quad \dots$$



Pro ekvidistantní uzlové body na intervalu délky  $T$ ,

$$x_i = a + i \frac{T}{n}, \quad i = 0, \dots, n - 1$$

jsou vektory  $\varphi_j(\vec{x})$  (kterých smí být nejvýše  $n$ ) ortogonální.

Pro ortogonální funkce je složitost  $\propto kn$ .

Pro  $k = n$  dostáváme  $\propto n^2$ .

**Rychlá Fourierova transformace (Fast Fourier Transform, FFT)** dále snižuje složitost na  $\propto n \ln n$ .

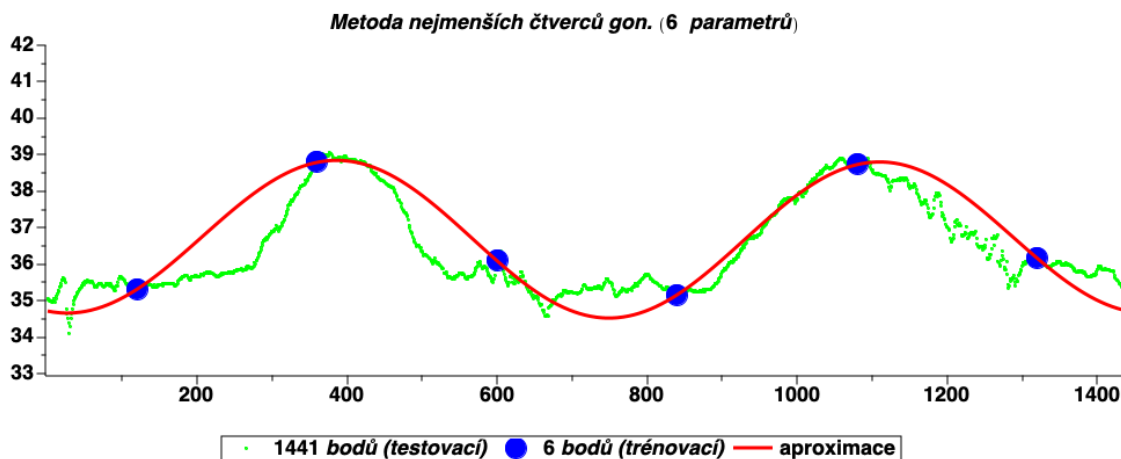
K tomu vyžaduje navíc  $n = 2^m$ , jinak efektivita klesá.

**Použití:** Aproximace periodických a „téměř periodických“ průběhů, zejména akustických, ale např. i obrazových; komprese mp3 a jpeg.

Rozklad na frekvence dovoluje další zpracování, digitální filtraci, rozpoznávání atd.

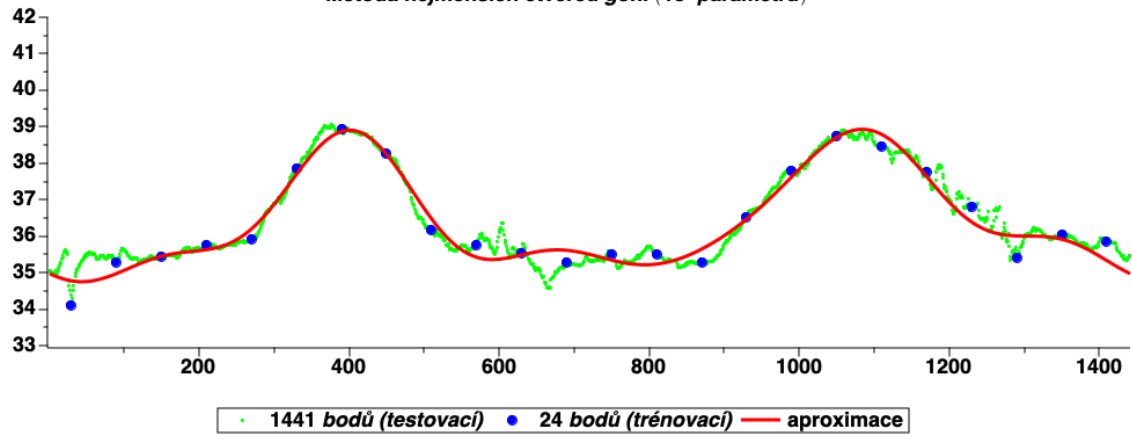
#### Příklad použití metody nejmenších čtverců na reálných datech

Vyjdeme z motivační úlohy (teplota pacienta) a vybrané hodnoty proložíme goniometrickým polynomem. (Čísla v závorkách udávají počet použitých aproximačních funkcí.)



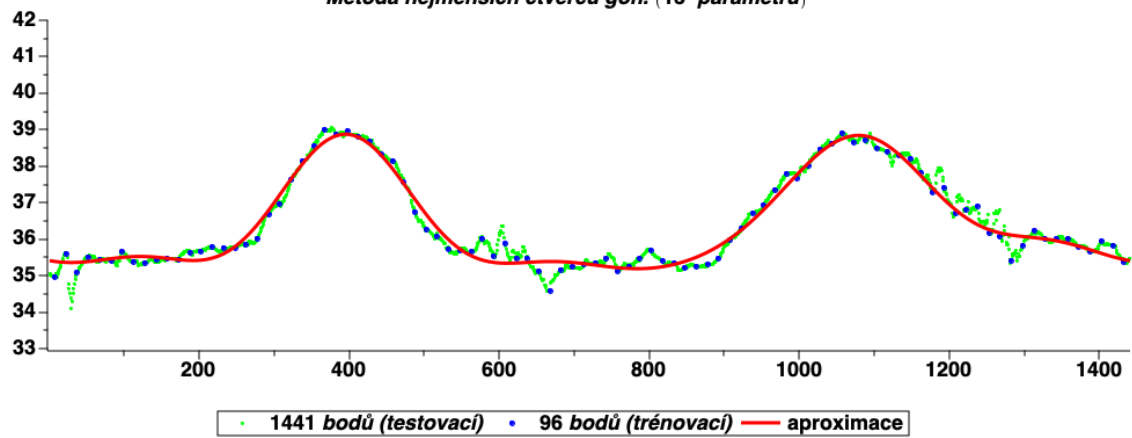
$$\varphi(1440/\sqrt{2}) \doteq 38.16, \quad \varphi(1439.5) \doteq 34.72$$

Metoda nejmenších čtverců gon. (13 parametrů)



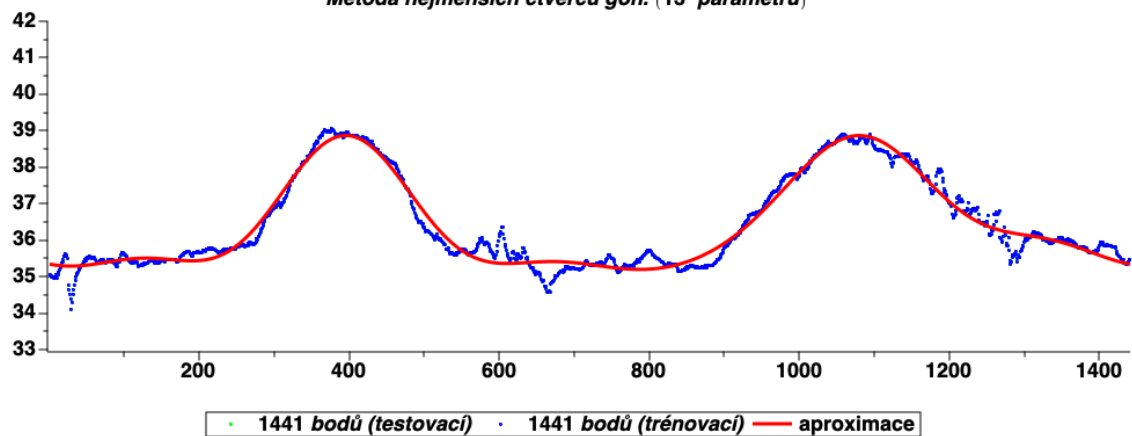
$$\varphi(1440/\sqrt{2}) \doteq 38.20, \quad \varphi(1439.5) \doteq 34.99$$

Metoda nejmenších čtverců gon. (13 parametrů)



$$\varphi(1440/\sqrt{2}) \doteq 38.20, \quad \varphi(1439.5) \doteq 35.41$$

Metoda nejmenších čtverců gon. (13 parametrů)



$$\varphi(1440/\sqrt{2}) \doteq 38.22, \quad \varphi(1439.5) \doteq 35.35$$

#### 1.5.4 Čebyševova aproximace polynomem

Úloha: Dáno:  
omezený interval  $I$ ,  
spojitá funkce  $f$  na  $I$ ,  
 $k \in \mathbb{N}$ .



**Hledáme:** polynom  $\varphi$  stupně menšího než  $k$  takový, abychom minimalizovali výraz

$$H_0 = \max_{t \in I} |\varphi(t) - f(t)|.$$

To se také dělá, ale je to mnohem pracnější. Častěji se používá modifikovaná aproximace metodou nejmenších čtverců:

### Čebyševova aproximace polynomem

Pro jednoduchost na intervalu  $I = \langle -1, 1 \rangle$ ; zobecnění na interval  $\langle a, b \rangle$  dostaneme lineární transformací

$$x = \frac{b+a}{2} + \frac{b-a}{2}z,$$

inverzní transformace je

$$z = \frac{x - \frac{b+a}{2}}{\frac{b-a}{2}}.$$

Za bázi prostoru všech polynomů stupně  $< k$  volíme **Čebyševovy polynomy**.

Pokud můžeme, volíme  $n \geq k$  uzlových bodů  $x_0, \dots, x_{n-1} \in \langle -1, 1 \rangle$  jako kořeny Čebyševova polynomu stupně  $n$ , tj. s kosinovým rozdělením:

$$x_i = \cos\left(\frac{\pi}{n}\left(i + \frac{1}{2}\right)\right), \quad i = 0, \dots, n-1.$$

Pro  $k = n$  dostaneme interpolační polynom (s doporučeným kosinovým rozdělením uzlových bodů). Řešení pro  $k < n$  se od interpolačního polynomu liší zanedbáním členů vyššího řádu. Koeficienty  $c_k, \dots, c_{n-1}$  bývají malé. (Závisí ovšem na vyšších derivacích aproximované funkce!) Chyba v uzlových bodech je proto omezena výrazem

$$|\varphi(x_i) - f(x_i)| \leq \sum_{j=k}^{n-1} |c_j|.$$

### Poznámky o Čebyševově aproximaci

- Neoptimalizujeme přesně kritérium  $H_0$ , ale výsledek se od optimálního řešení příliš neliší.
- O chybě mimo uzlové body nelze říci mnoho, přesto lze postup doporučit.
- Rekurentní vzorec lze použít nejen ke stanovení Čebyševových polynomů, ale i přímo k výpočtu jejich hodnot v daném bodě.
- Nedoporučuje se výsledek roznásobovat do standardního tvaru  $\varphi(t) = \sum_{j < k} b_j t^j$ .
- Metodu lze zobecnit i na případ, kdy hledáme aproximaci ve tvaru součinu známé funkce a neznámého polynomu.

## Dodatek: Přehled značení

Popis je zjednodušený a nemusí být přesný, podrobnosti jsou v textu. Značení použité jen lokálně zde není uvedeno.

### Značení specifické pro tuto kapitolu

$n$  ... počet uzlových bodů

$\vec{x} = (x_0, \dots, x_{n-1})$  ... vektor uzlových bodů

$\vec{y} = (y_0, \dots, y_{n-1})$  ... vektor požadovaných hodnot v uzlových bodech

$f$  ... aproximovaná funkce,  $f(x_i) = y_i$

$\varphi$  ... aproximující funkce,  $\varphi \approx f$ ,  $\varphi(x_i) \doteq y_i$

$\mathcal{F}$  ... množina povolených aproximujících funkcí,  $\varphi \in \mathcal{F}$

$(\varphi_0, \dots, \varphi_{k-1})$  ... báze prostoru povolených aproximujících funkcí,  $\mathcal{F} \in \text{Lin}\{\varphi_0, \dots, \varphi_{k-1}\}$

$k$  ... dimenze prostoru povolených aproximujících funkcí,  $k = \dim \mathcal{F}$

$\vec{c} = (c_0, \dots, c_{k-1})$  ... vektor koeficientů lineární kombinace, souřadnice aproximace  $\varphi$  vzhledem k bázi  $(\varphi_0, \dots, \varphi_{k-1})$ ,

$\varphi = \sum_{j < k} c_j \varphi_j$

$\varphi(\vec{x}), \varphi_0(\vec{x}), \dots, \varphi_{k-1}(\vec{x}), \psi_0(\vec{x})$  apod. ... vektory hodnot funkcí v uzlových bodech  $x_0, \dots, x_{n-1}$ ,  $\varphi(\vec{x}) = (\varphi(x_0), \dots, \varphi(x_{n-1}))$  apod.

$W$  ... polynom použitý v odhadech chyby,  $W(t) = \prod_{i < n} (t - x_i)$

$\bar{w}$  ... horní odhad absolutní hodnoty polynomu  $W$  na použitém intervalu

$\gamma_j$  ... Čebyševův polynom řádu  $j$ ,  $\gamma_j(t) = \cos(j \arccos t)$

$y_{i,j}$  ... požadovaná hodnota  $j$ -té derivace v  $i$ -tém uzlovém bodě,  $y_{i,j} = f^{(j)}(x_i) \doteq \varphi^{(j)}(x_i)$

$T$  ... perioda aproximované periodické funkce

### Značení používané podobně v celém předmětu

$\delta_{ij}$  ... Kroneckerovo delta,  $\delta_{ii} = 1$ ,  $\delta_{ij} = 0$  pro  $i \neq j$

$\cdot$  ... skalární součin,  $\vec{u} \cdot \vec{v} = \sum_{i < n} u_i \cdot v_i$

$\|\cdot\|$  ... norma, obvykle euklidovská,  $\|\vec{u}\| = \sqrt{\vec{u} \cdot \vec{u}}$

$I(\dots)$  ... nejmenší interval obsahující čísla (body) v závorce, např.  $I(x_0, \dots, x_{n-1}) = \langle \min_i x_i, \max_i x_i \rangle$

$M_j$  ... horní odhad absolutní hodnoty  $j$ -té derivace aproximované funkce,  $|f^{(j)}| \leq M_j$  na použitém intervalu

## 2 NUMERICKÁ DERIVACE A RICHARDSONOVA EXTRAPOLACE

### 2.1 Formulace problému

**Úloha:** Odhadnout  $f'(x)$  pomocí funkčních hodnot v konečně mnoha bodech.

**Proč to nedělat?**

1. Umíme symbolicky zderivovat jakýkoli výraz. ☹
2. Např. aproximace polynomem dle Weierstrassovy věty má libovolně velkou chybu derivace. ☹

**Proč to dělat?**

1. Symbolická derivace je programátorsky náročná. ☹
2. **Ne vždy je derivovaná funkce zadána vzorcem, máme např. jen diskrétní data.** ☹
3. Chce se to po nás.

### Základní numerické odhady derivace

Z definice

$$f'(x) = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h},$$

dostaneme odhad

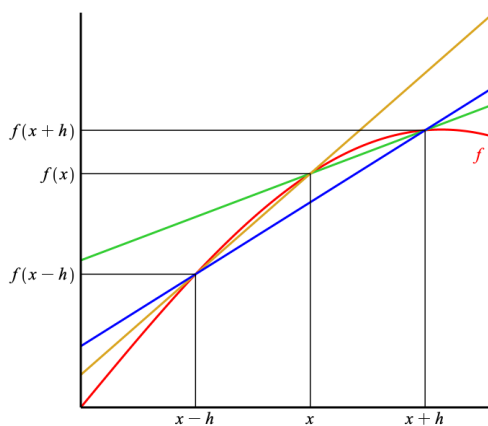
$$d_n(x, h) = \frac{f(x+h) - f(x)}{h}.$$

Směrnici tečny nahrazujeme směrnicí sečny v bodech  $(x, f(x))$  a  $(x+h, f(x+h))$ .

Symetrický odhad

$$d_s(x, h) = \frac{d_n(x, h) + d_n(x, -h)}{2} = \frac{f(x+h) - f(x-h)}{2h},$$

je směrnice sečny, vedené body  $(x-h, f(x-h))$  a  $(x+h, f(x+h))$ .



$$f, d_n(x, h) = \frac{f(x+h) - f(x)}{h}, d_n(x, -h) = \frac{f(x) - f(x-h)}{h}, d_s(x, h) = \frac{f(x+h) - f(x-h)}{2h}$$

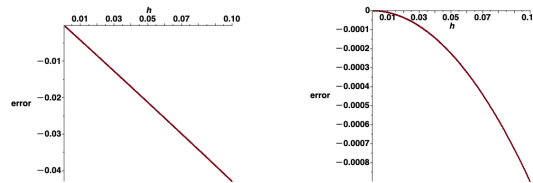
## 2.2 Chyba metody u numerické derivace

Taylorův rozvoj funkce  $f$  a odhadů derivace podle  $h$  v okolí 0:

$$\begin{aligned}
 f(x+h) &= f(x) + hf'(x) + \frac{h^2}{2}f''(x) + \frac{h^3}{6}f'''(x) + \frac{h^4}{24}f^{(4)}(x) + \frac{h^5}{120}f^{(5)}(x) + \dots, \\
 d_n(x, h) &= \frac{f(x+h) - f(x)}{h} \\
 &= f'(x) + \frac{h}{2}f''(x) + \frac{h^2}{6}f'''(x) + \frac{h^3}{24}f^{(4)}(x) + \frac{h^4}{120}f^{(5)}(x) + \dots, \\
 d_n(x, -h) &= \frac{f(x) - f(x-h)}{h} \\
 &= f'(x) - \frac{h}{2}f''(x) + \frac{h^2}{6}f'''(x) - \frac{h^3}{24}f^{(4)}(x) + \frac{h^4}{120}f^{(5)}(x) - \dots, \\
 d_s(x, h) &= \frac{f(x+h) - f(x-h)}{2h} \\
 &= f'(x) + \frac{h^2}{6}f'''(x) + \frac{h^4}{120}f^{(5)}(x) + \dots
 \end{aligned}$$

Všimněte si, že libovolný odhad derivace dělí  $f$  krokem  $h$ , takže v jejím Taylorově rozvoji se  $h^p$  násobí  $f^{(p+1)}(x)$ .

### 2.2.1 Řád metod numerické derivace



Závislost chyby nesymetrického odhadu ( $\approx$  lineární) a symetrického ( $\approx$  kvadratická).

Obecně derivaci  $f'(x)$  nahrazujeme odhadem

$$d(x, h) = f'(x) + e(h),$$

kde  $e(h)$  je chyba odhadu (její závislost na  $x$  zde nevyznačujeme).

Ta bývá pro malé  $h$  přibližně úměrná nějaké jeho mocnině,

$$|e(h)| \doteq c \cdot |h|^p.$$

Zlogaritmováním získáme lineární závislost na parametrech:

$$\begin{aligned}
 \ln |e(h)| &\doteq \ln c + p \ln |h|, \\
 p &\doteq \frac{\ln |e(h)| - \ln c}{\ln |h|} \rightarrow \frac{\ln |e(h)|}{\ln |h|}
 \end{aligned}$$

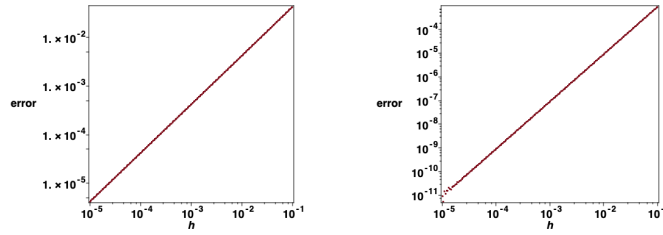
pro  $h \rightarrow 0$ , tj.  $\ln |h| \rightarrow -\infty$ .

Definujeme **řád metody** (řád odhadu  $d(x, h)$ ) jako

$$p = \lim_{h \rightarrow 0} \frac{\ln |e(h)|}{\ln |h|}$$

(pokud limita existuje).

**Poznámka 2.1** V logaritmických souřadnicích má řád metody význam směrnice asymptoty v  $(-\infty, \infty)$ .



Závislost chyby nesymetrického odhadu ( $\approx$  lineární) a symetrického ( $\approx$  kvadratická).

**Věta 2.1** *Nechť  $p$  je nejmenší číslo  $\geq 1$ , pro které koeficient u  $h^p$  v Taylorově rozvoje chyby odhadu  $d(x, h)$  derivace  $f'(x)$  může být obecně nenulový. Nechť  $f$  má v okolí bodu  $x$  spojitou derivaci řádu  $p+1$ . Pak  $p$  je řád metody.*

**Důkaz.** Z Lagrangeova tvaru zbytku Taylorova rozvoje

$$e(h) = \frac{h^p}{p!} f^{(p+1)}(\xi_h)$$

pro nějaké  $\xi_h \in I(x, x+h)$ .

$$\begin{aligned} \lim_{h \rightarrow 0} \xi_h &= x, \\ \lim_{h \rightarrow 0} f^{(p+1)}(\xi_h) &= f^{(p+1)}(x) \in (-\infty, \infty), \\ \lim_{h \rightarrow 0} \frac{\ln |e(h)|}{\ln |h|} &= \lim_{h \rightarrow 0} \frac{p \ln |h| - \ln p! + \ln |f^{(p+1)}(\xi_h)|}{\ln |h|} \\ &= p - \lim_{h \rightarrow 0} \frac{\ln p!}{\ln |h|} + \lim_{h \rightarrow 0} \frac{\ln |f^{(p+1)}(\xi_h)|}{\ln h} = p. \end{aligned}$$

Čitatel posledního zlomku konverguje k  $\ln |f^{(p+1)}(x)|$ , jmenovatel k  $-\infty$ , celý zlomek k 0.  $\square$   
 Odhad  $d_n(x, h)$  je řádu 1,  $d_s(x, h)$  řádu 2.

### 2.3 Odhady chyb metody u numerické derivace

Pro **nesymetrický** odhad:

$$d_n(x, h) = f'(x) + \frac{h}{2} f''(\xi),$$

kde  $\xi \in I(x, x+h)$ , pokud  $f$  má na intervalu  $I(x, x+h)$  spojitou druhou derivaci. Pak existuje  $M_2$  takové, že

$$\forall t \in I(x, x+h) : |f''(t)| \leq M_2.$$

$$|d_n(x, h) - f'(x)| \leq \frac{M_2}{2} |h|.$$

#### Odhady chyb metody u numerické derivace

Pro **symetrický** odhad:

$$d_s(x, h) = f'(x) + \frac{h^2}{6} f'''(\xi),$$

kde  $\xi \in I(x-h, x+h)$ , pokud  $f$  má na intervalu  $I(x-h, x+h)$  spojitou třetí derivaci. Pak existuje  $M_3$  takové, že

$$\forall t \in I(x-h, x+h) : |f'''(t)| \leq M_3.$$

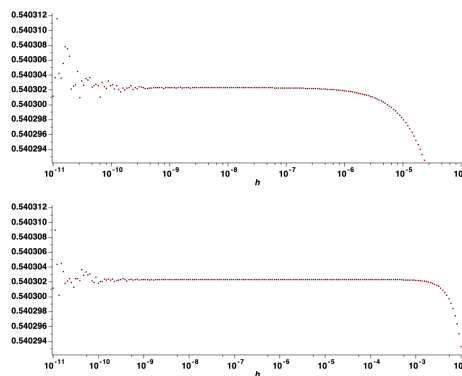
$$|d_s(x, h) - f'(x)| \leq \frac{M_3}{6} h^2.$$

## 2.4 Doporučená délka kroku

### Typické výsledky numerické derivace

Typická závislost výsledku a chyby numerické derivace na kroku  
(odhady derivace funkce sin v bodě 1, přesnější výsledek 0.5403023059):

krok	nesymetrický odhad		symetrický odhad	
	výsledek	chyba	výsledek	chyba
$10^{-2}$	0.5360859800	$-4.2 \cdot 10^{-3}$	0.5402933000	$-9 \cdot 10^{-6}$
$10^{-3}$	0.5398815000	$-4.2 \cdot 10^{-4}$	0.5403022000	$-1 \cdot 10^{-7}$
$10^{-4}$	0.5402600000	$-4.2 \cdot 10^{-5}$	0.5403020000	$-3 \cdot 10^{-7}$
$10^{-5}$	0.5403000000	$-2.3 \cdot 10^{-6}$	0.5403100000	$7.7 \cdot 10^{-6}$
$10^{-6}$	0.5403000000	$-2.3 \cdot 10^{-6}$	0.5404000000	$1 \cdot 10^{-4}$
$10^{-7}$	0.5400000000	$-3 \cdot 10^{-4}$	0.5400000000	$-3 \cdot 10^{-4}$
$10^{-8}$	0.5400000000	$-3 \cdot 10^{-4}$	0.5400000000	$-3 \cdot 10^{-4}$
$10^{-9}$	0.5000000000	$-4 \cdot 10^{-2}$	0.4000000000	-0.14
$10^{-10}$	0	-0.54	0	-0.54



Typická závislost výsledku numerické derivace na kroku  
(nesymetrický a symetrický odhad derivace funkce sin v bodě 1).

Vyjdeme z odhadu chyby metody tvaru

$$\frac{M_{p+1}}{c} h^p,$$

kde  $p$  je řád metody,

$M_{p+1}$  je odhad  $|f^{(p+1)}|$ ,

$c$  je konstanta pro danou metodu (nejčastěji  $(p+1)!$ ).

Zaokrouhlovací chybu odhadneme výrazem

$$b r M_0 \frac{1}{h},$$

kde  $M_0$  je odhad  $|f|$ ,

$r$  je relativní přesnost numerického výpočtu funkčních hodnot,

$b$  je konstanta pro danou metodu (většinou řádu jednotek, určená počtem sčítanců v čitateli použitého výrazu).

Odhad celkové chyby:

$$d_{Rn}(h) = \frac{M_{p+1}}{c} h^p + b r M_0 \frac{1}{h}$$

minimum nastane pro  $h_{dop}$ :

$$e'(h_{dop}) = 0$$

$$p \frac{M_{p+1}}{c} h_{dop}^{p-1} - \frac{b r M_0}{h_{dop}^2} = 0$$

$$h_{dop} = \sqrt[p+1]{\frac{b c r M_0}{p M_{p+1}}}$$

Pro  $d_n(x, h)$ :  $p = 1$ ,  $c = 2$ ,  $b = 2$ ,

$$h_{dop} = 2 \sqrt{\frac{r M_0}{M_2}} \quad \text{odhad chyby metody} \quad \frac{M_2}{2} h_{dop} = \sqrt{M_0 M_2 r} \sim \sqrt{r}$$

To je špatná zpráva! (Zaokrouhlovací chyba je podobná.)

Pro  $d_s(x, h)$ :  $p = 2$ ,  $c = 6$ ,  $b = 1$  (v čitateli máme dva členy, ale dělíme dvěma),

$$h_{dop} = \sqrt[3]{\frac{3r M_0}{M_3}} \quad \text{odhad chyby metody} \quad \frac{M_3}{6} h_{dop}^2 = \frac{1}{2 \cdot \sqrt[3]{3}} M_0^{2/3} M_3^{1/3} r^{2/3}$$

**Příklad:**  $r = 10^{-10}$ , funkční hodnoty i hodnoty derivací zhruba stejné (jako např. u funkce  $x \mapsto e^x$ ):

Pro odhad  $d_n(x, h)$ :  $h_{dop} = 2 \sqrt{10^{-10}} = 2 \cdot 10^{-5}$  s odhadem relativní chyby  $\frac{\sqrt{M_0 M_2}}{M_1} \sqrt{r} = \sqrt{r} = 10^{-5}$ ,

Pro odhad  $d_s(x, h)$ :  $h_{dop} = \sqrt[3]{3 \cdot 10^{-10}} \doteq 6.7 \cdot 10^{-4}$  s odhadem relativní chyby  $\frac{1}{2 \cdot 3^{1/3}} \frac{M_0^{2/3} M_3^{1/3}}{M_1} r^{2/3} = \frac{1}{2 \cdot 3^{1/3}} r^{2/3} \doteq 7.47 \cdot 10^{-8}$ .

Pro funkci  $x \mapsto e^{100x}$  (pouze změna měřítka na ose  $x$ ) je  $M_k = 100^k M_0$ ,

- pro odhad  $d_n(x, h)$ :  $h_{dop} = 2 \sqrt{10^{-14}} = 2 \cdot 10^{-7}$  s odhadem relativní chyby  $\sqrt{r} = 10^{-5}$ ,
- pro odhad  $d_s(x, h)$ :  $h_{dop} = \sqrt[3]{3 \cdot 10^{-16}} \doteq 6.7 \cdot 10^{-6}$  s odhadem relativní chyby  $\frac{1}{2 \cdot 3^{1/3}} r^{2/3} \doteq 7.47 \cdot 10^{-8}$ .

Pro odhad derivace  $\ln$  v bodě  $x = 10^{-6}$  (předchozí délky kroků nelze použít):  $M_0 \doteq 14$ ,  $M_1 \doteq 10^6$ ,  $M_2 \doteq 10^{12}$ ,  $M_3 \doteq 2 \cdot 10^{18}$ ,

- pro odhad  $d_n(x, h)$ :  $h_{dop} = \sqrt{4 \cdot 14 \cdot 10^{-22}} \doteq 8 \cdot 10^{-11}$  s odhadem relativní chyby  $\frac{\sqrt{14 \cdot 10^{12}}}{10^6} \sqrt{r} \doteq 3.74 \cdot 10^{-5}$ ,
- pro odhad  $d_s(x, h)$ :  $h_{dop} = \sqrt[3]{3 \cdot 7 \cdot 10^{-28}} \doteq 1.3 \cdot 10^{-9}$  s odhadem relativní chyby  $\frac{1}{2 \cdot 3^{1/3}} \frac{14^{2/3} \cdot (2 \cdot 10^{18})^{1/3}}{10^6} r^{2/3} \doteq 5.47 \cdot 10^{-7}$ .

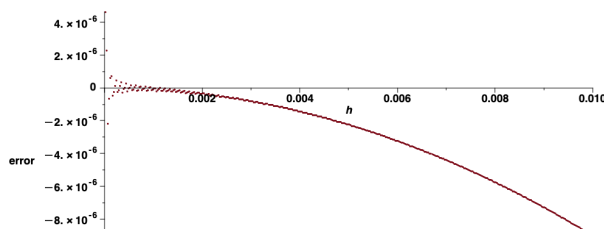
### Upřesnění:

Dosud jsme uvažovali jen chybu vyhodnocení funkce  $f$  pro přesný argument, odhadnutou výrazem  $r M_0$ . Nepřesnost  $r x$  v argumentu se projeví ve funkční hodnotě chybou přibližně  $r x M_1$ , která může být značná, bude-li velká (absolutní hodnota) derivace funkce  $f$ . Proto je žádoucí volit čísla  $h, x, x + h$  tak, aby byla v počítači zobrazena přesně, tedy nikoli např.  $h = 10^{-3}$  v binární reprezentaci.

Odvození optimální délky kroku by se mělo modifikovat podle toho, která z chyb  $r M_0$ ,  $r x M_1$  je větší. (Ještě lépe by bylo uvažovat obě chyby najednou, ale tím by se řešení zkomplikovalo, ani by nemuselo být jednoznačné.)

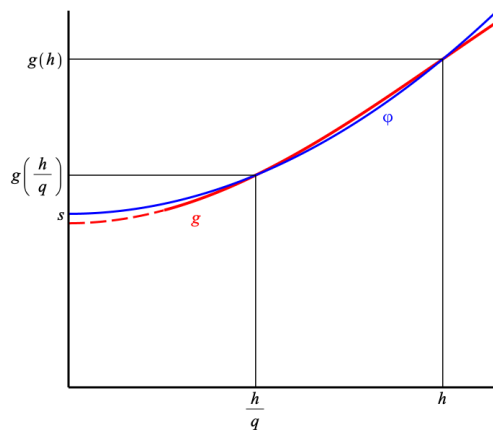
## 2.5 Obecný princip Richardsonovy extrapolace

Motivace:



Závislost chyby symetrického odhadu derivace ( $\approx$  kvadratická).

Princip:



**Úloha:** Správný výsledek nějakého výpočtu je  $g(0) = \lim_{h \rightarrow 0} g(h)$ . Předpokládáme, že  $g$  má v okolí bodu 0 Taylorův rozvoj

$$g(h) = g(0) + \frac{h^p}{p!} g^{(p)}(0) + \frac{h^r}{r!} g^{(r)}(0) + \dots,$$

kde  $p$  (řád metody) známe a  $r > p$ . Z hodnot funkce  $g$  v konečně mnoha nenulových bodech máme odhadnout  $g(0)$ .

Zanedbáme členy řádů vyšších než  $p$  a aproximujeme  $g$  polynomem  $\varphi(h) = s + ch^p$ ,  $s, c \in \mathbb{R}$ . Ke stanovení  $s, c$  zvolíme 2 uzlové body  $h, h/q$ , kde  $q \neq 1$ :

$$\begin{aligned} \varphi(h) &= s + ch^p = g(h), \\ \varphi\left(\frac{h}{q}\right) &= s + c\frac{h^p}{q^p} = g\left(\frac{h}{q}\right). \end{aligned}$$

To je regulární soustava dvou lineárních rovnic pro dvě neznámé  $s, c$ , z nichž nás zajímá pouze  $s = \varphi(0)$ :

$$\begin{aligned} (q^p - 1)s &= q^p g\left(\frac{h}{q}\right) - g(h), \\ s &= \frac{q^p g\left(\frac{h}{q}\right) - g(h)}{q^p - 1}. \end{aligned}$$

Odhad  $s$  hodnoty  $g(0)$  je zatížen pouze chybami vyšších řádů než  $p$  (zde řádu  $r$ ).

Často  $q = 2$ , pak

$$s = \frac{2^p g\left(\frac{h}{2}\right) - g(h)}{2^p - 1}.$$

## 2.6 Využití Richardsonovy extrapolace v numerické derivaci

Odhad  $d_n(x, h)$  má chybu řádu 1; z hodnot  $d_n(x, h), d_n(x, h/q)$  vypočteme odhad

$$d_{Rn}(x, h) = \frac{q d_n(x, h/q) - d_n(x, h)}{q - 1},$$

s chybou řádu 2. Pro  $q = 2$ :

$$d_{Rn}(x, h) = 2 d_n(x, h/2) - d_n(x, h) = \frac{-f(x+h) + 4f(x+h/2) - 3f(x)}{h}.$$

Ze symetrických odhadů  $d_s(x, h), d_s(x, h/q)$ :

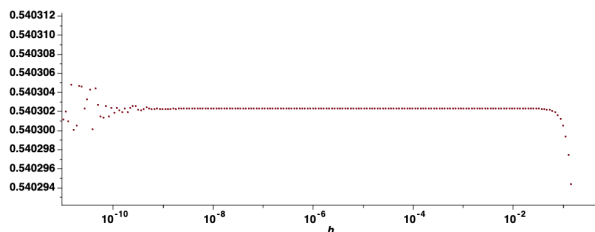
$$d_{Rs}(x, h) = \frac{q^2 d_s(x, h/q) - d_s(x, h)}{q^2 - 1},$$



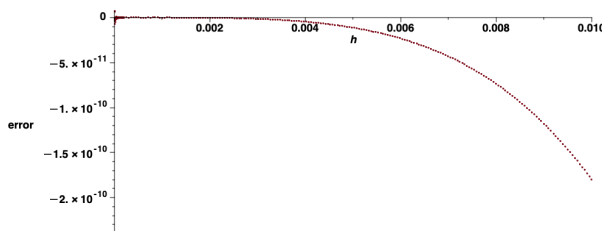
s chybou řádu 4. Pro  $q = 2$ :

$$d_{Rs}(x, h) = \frac{4d_s(x, h/2) - d_s(x, h)}{3} = \frac{-f(x+h) + 8f(x+h/2) - 8f(x-h/2) + f(x-h)}{6h}.$$

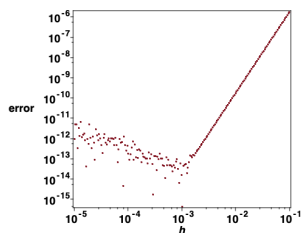
Symetrický odhad  $d_s(x, h)$  lze též dostat Richardsonovou extrapolací z odhadu  $d_{Rn}(x, h)$  s  $q = -1$ .



Výsledky symetrického vzorce zpřesněného Richardsonovou extrapolací.



Chyba (4. řádu) symetrického vzorce zpřesněného Richardsonovou extrapolací.

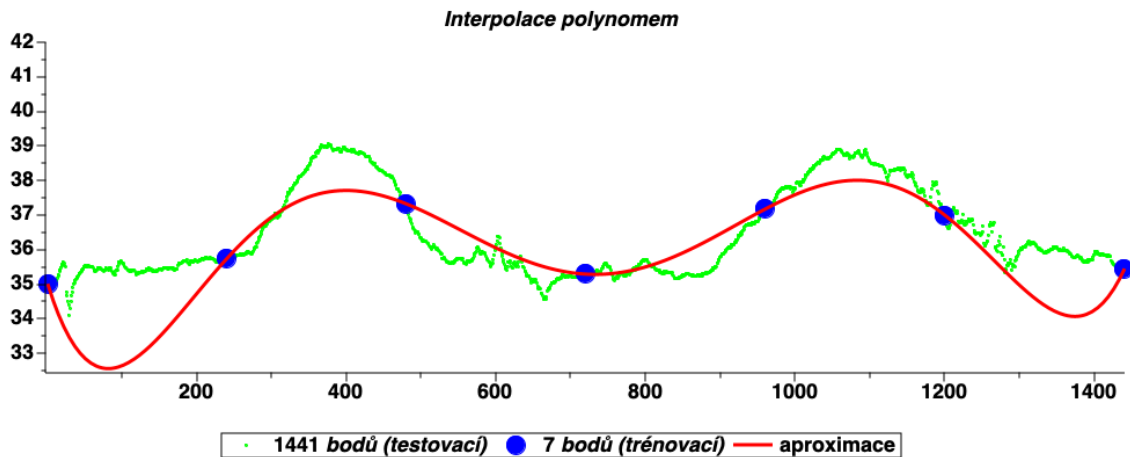


Chyba (4. řádu) symetrického vzorce zpřesněného Richardsonovou extrapolací.

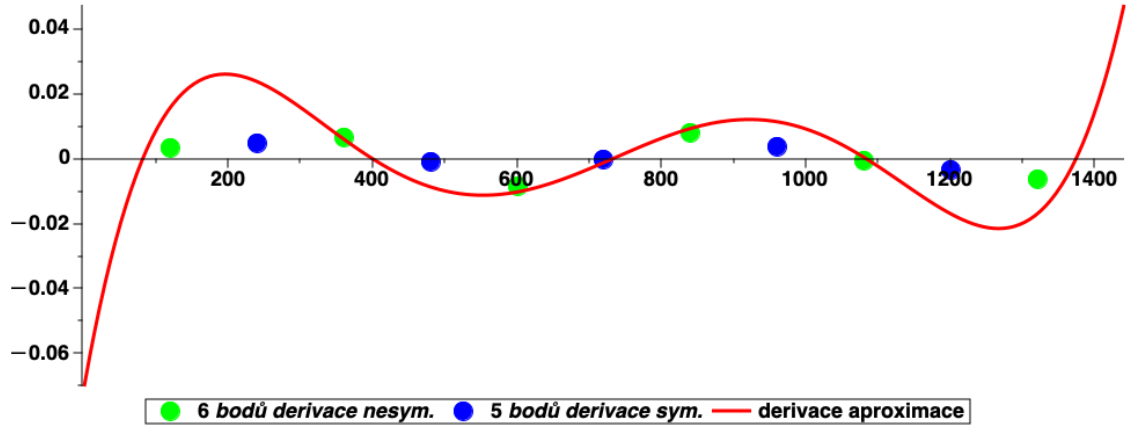
## 2.7 Odhad derivace z reálných dat

Vyjdeme z motivační úlohy (teplota pacienta).

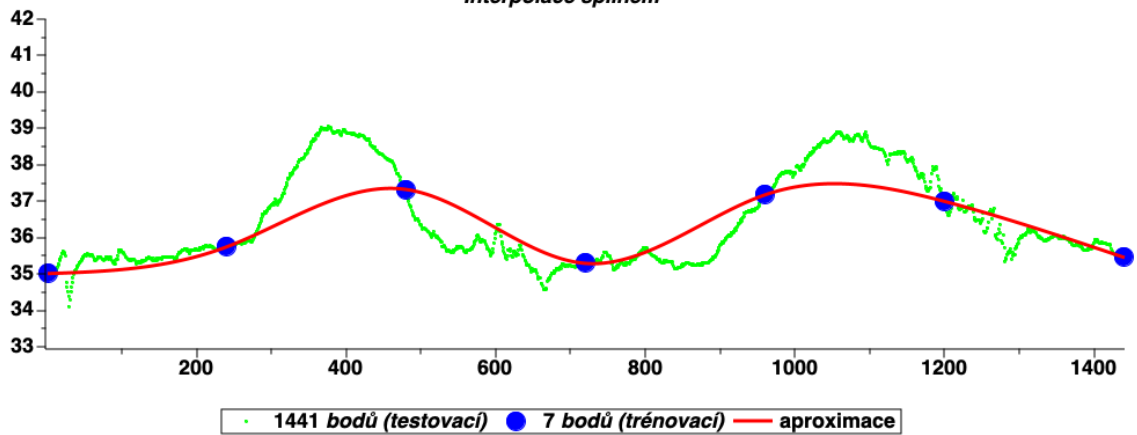
Použijeme nesymetrický i symetrický vzorec pro odhad derivace a porovnáme s derivací dříve použitých aproximací.



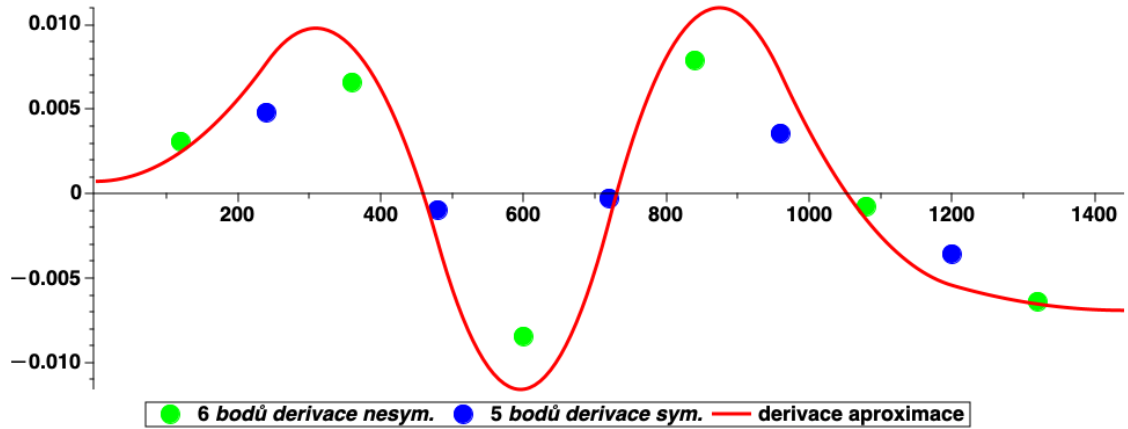
Interpolace polynomem



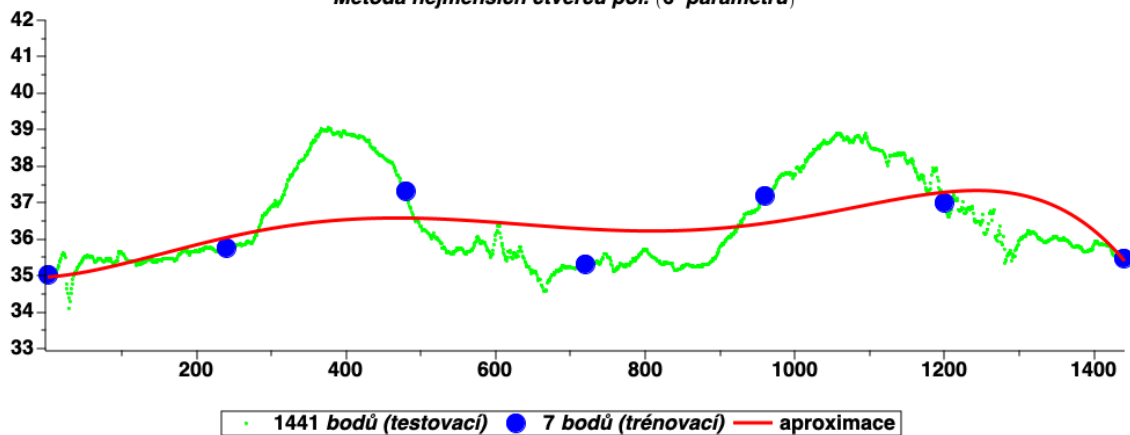
Interpolace splinem



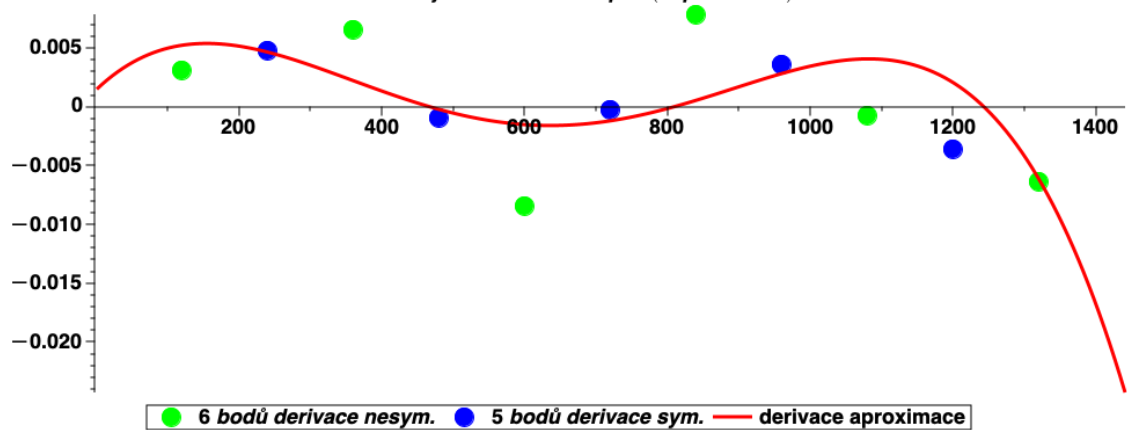
Interpolace splinem



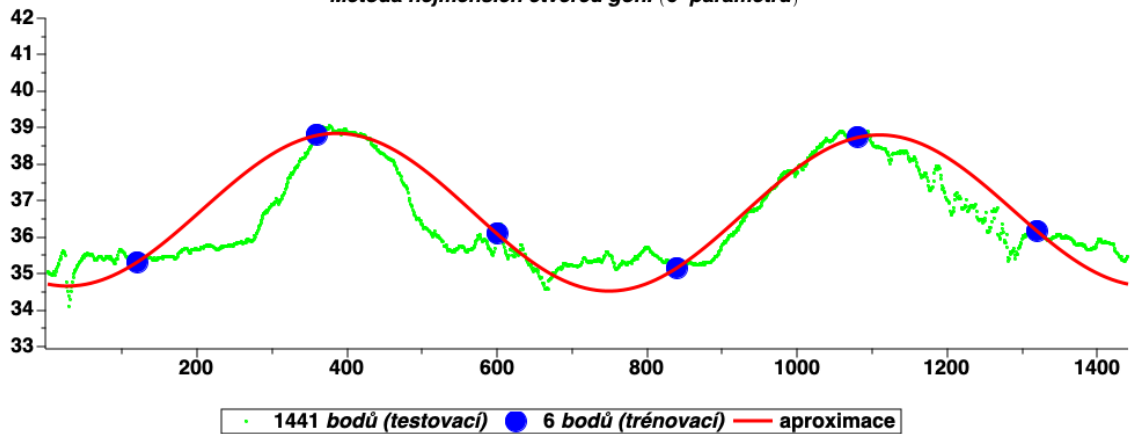
Metoda nejmenších čtverců pol. (6 parametrů)

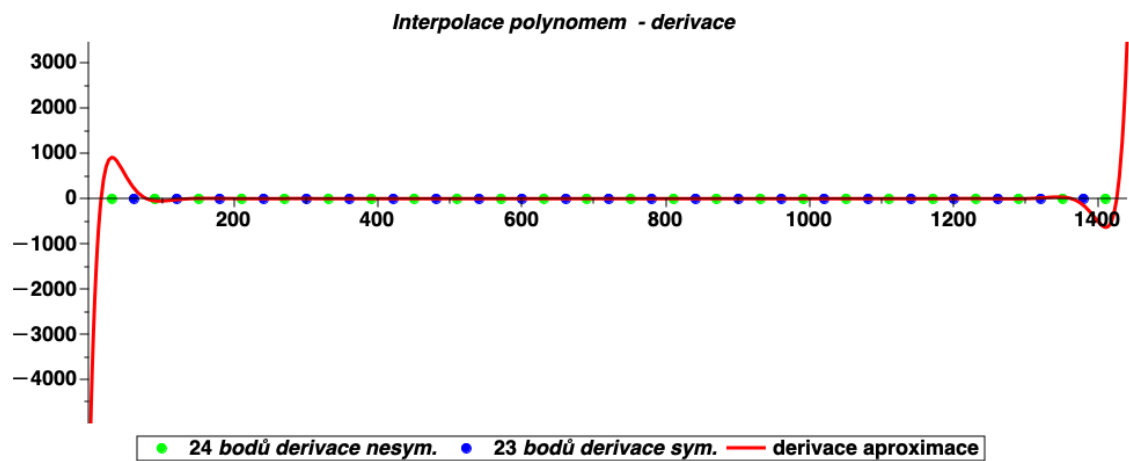
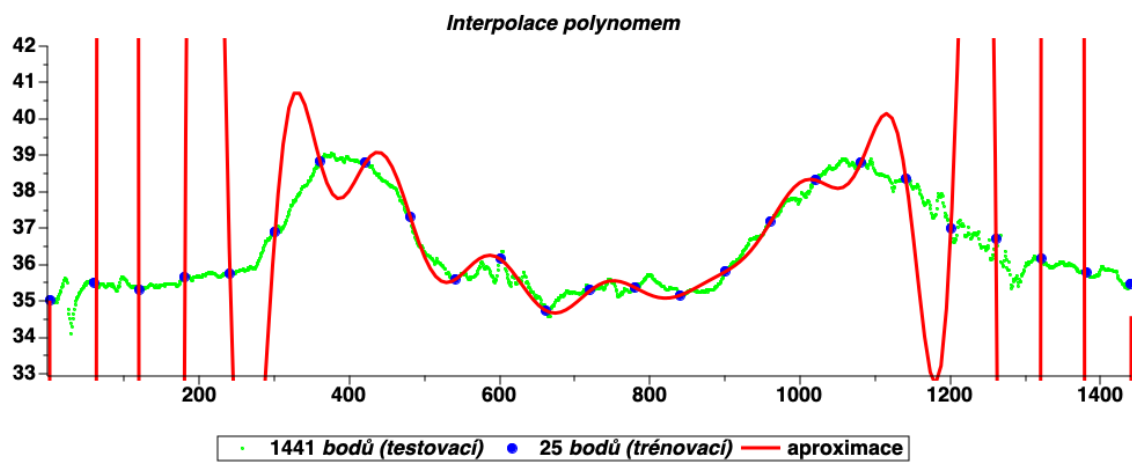
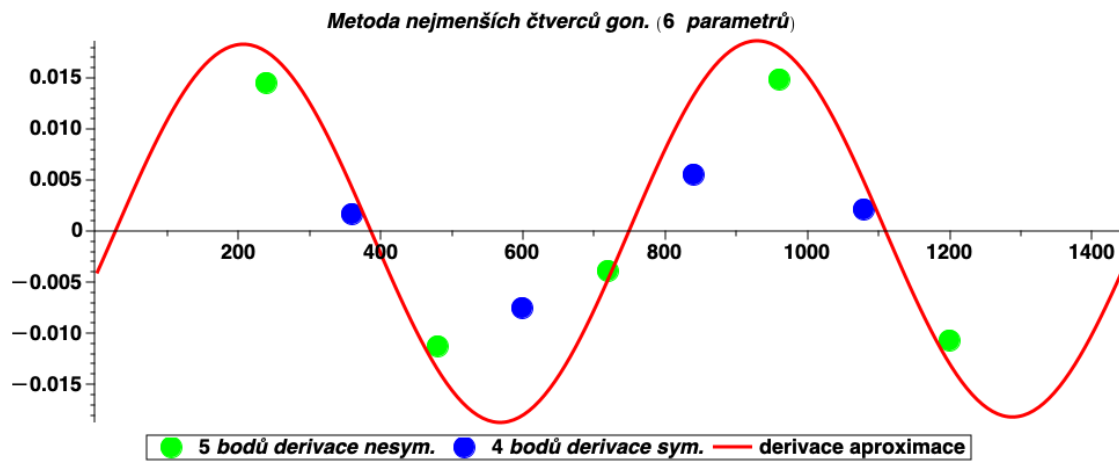


Metoda nejmenších čtverců pol. (6 parametrů)

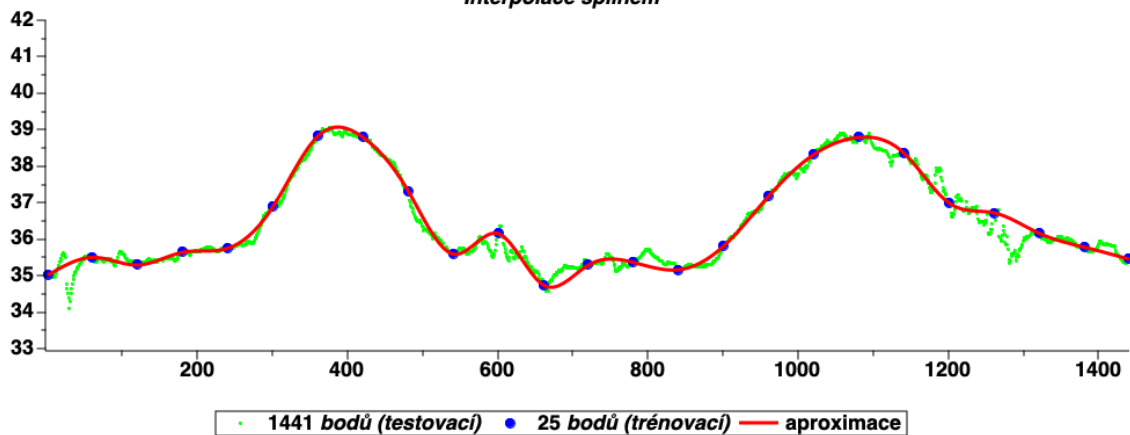


Metoda nejmenších čtverců gon. (6 parametrů)

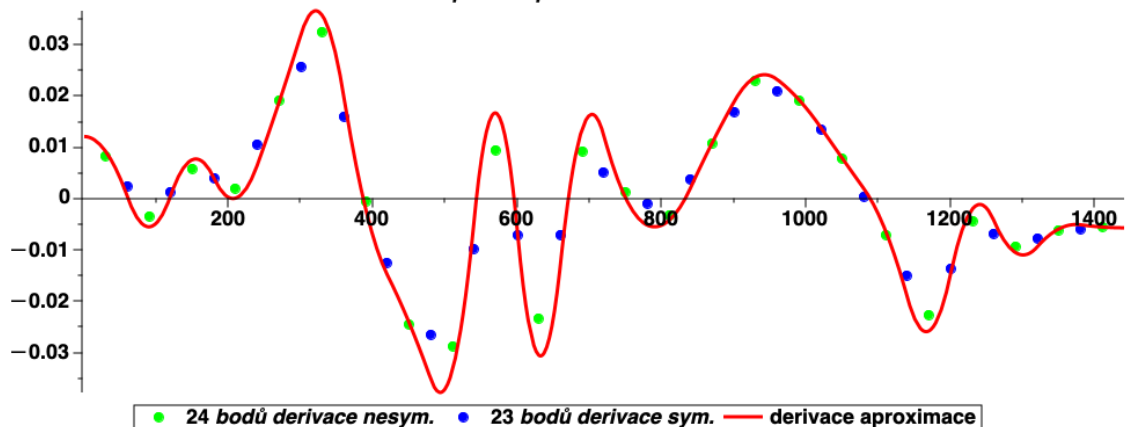




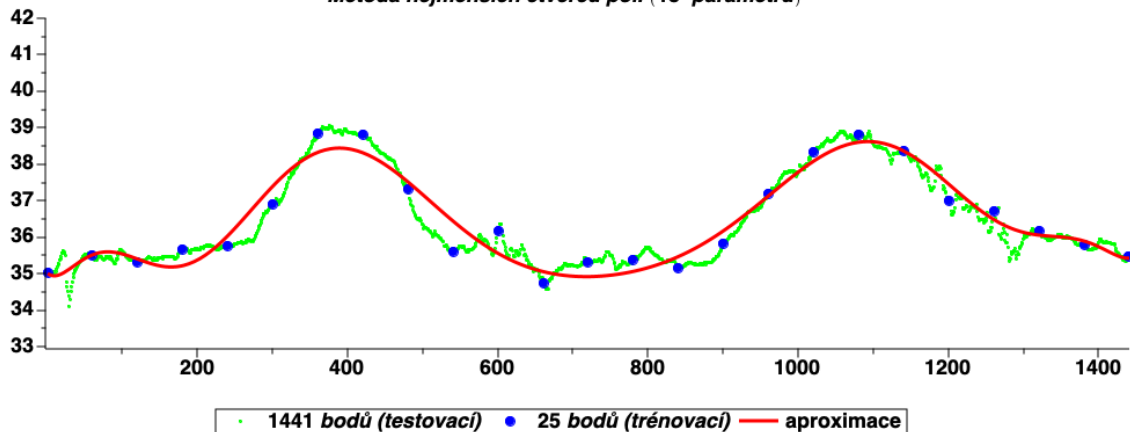
Interpolace splínem

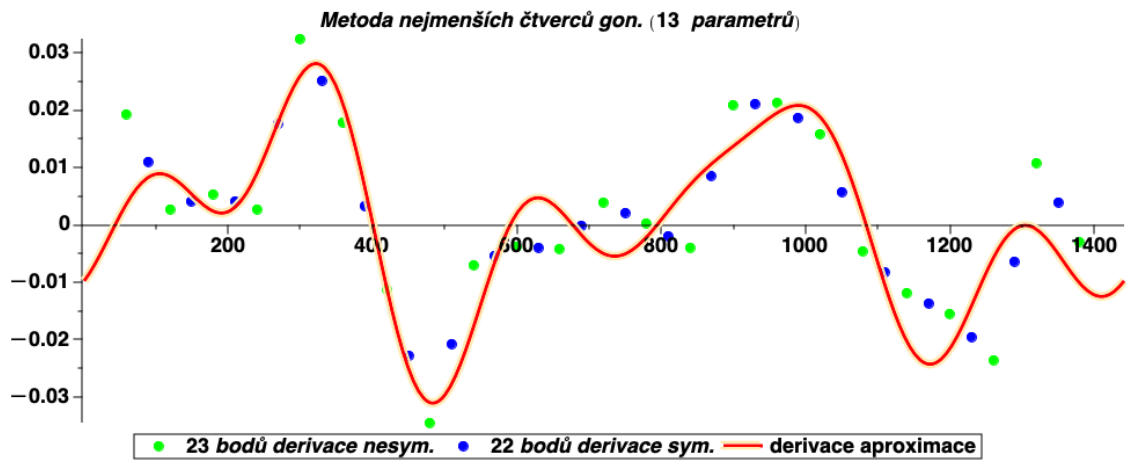
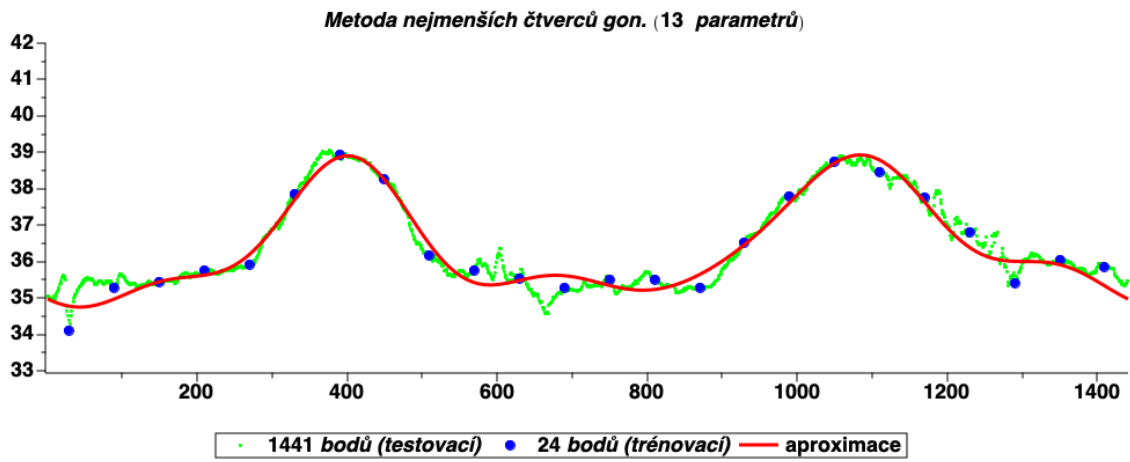
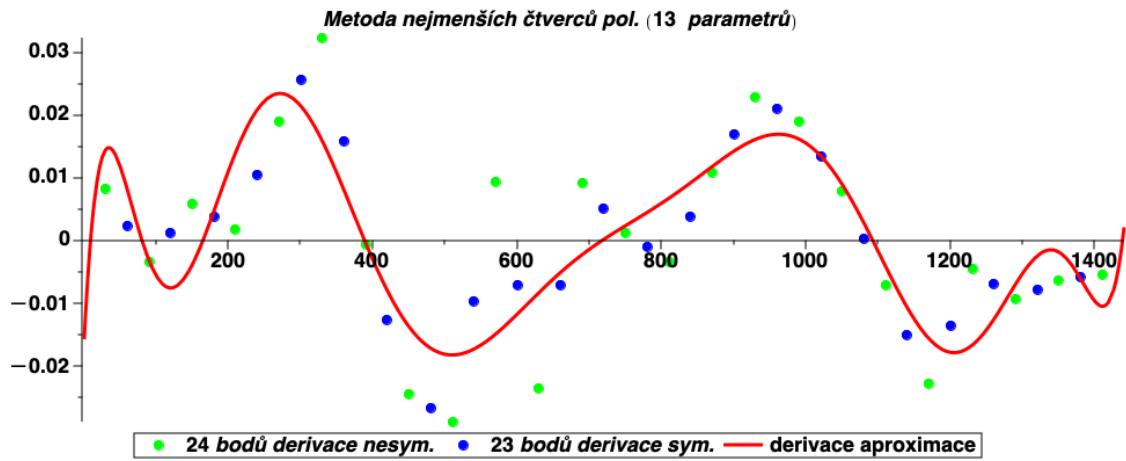


Interpolace splínem - derivace

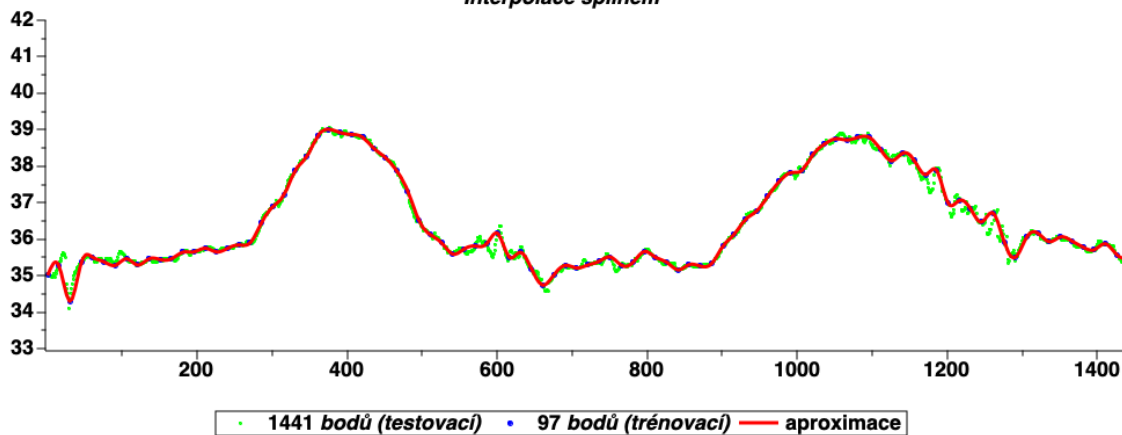


Metoda nejmenších čtverců pol. (13 parametrů)

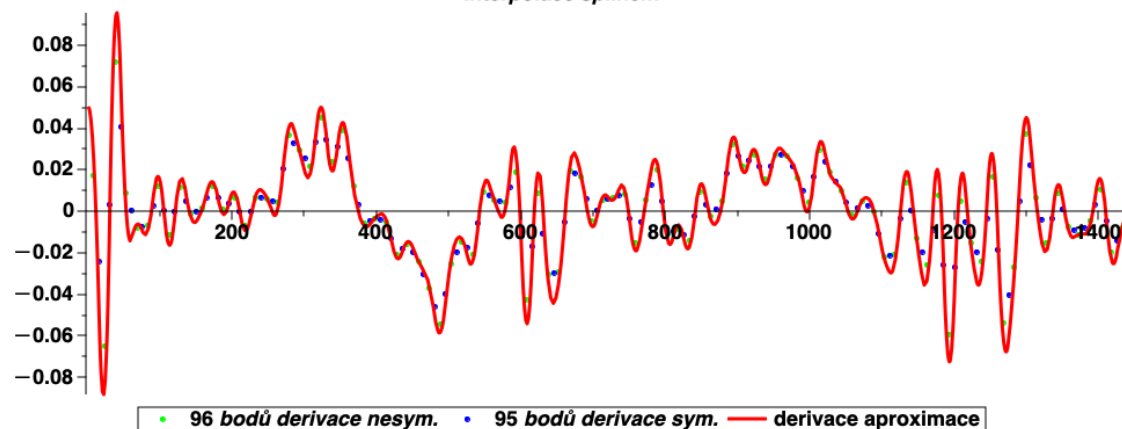




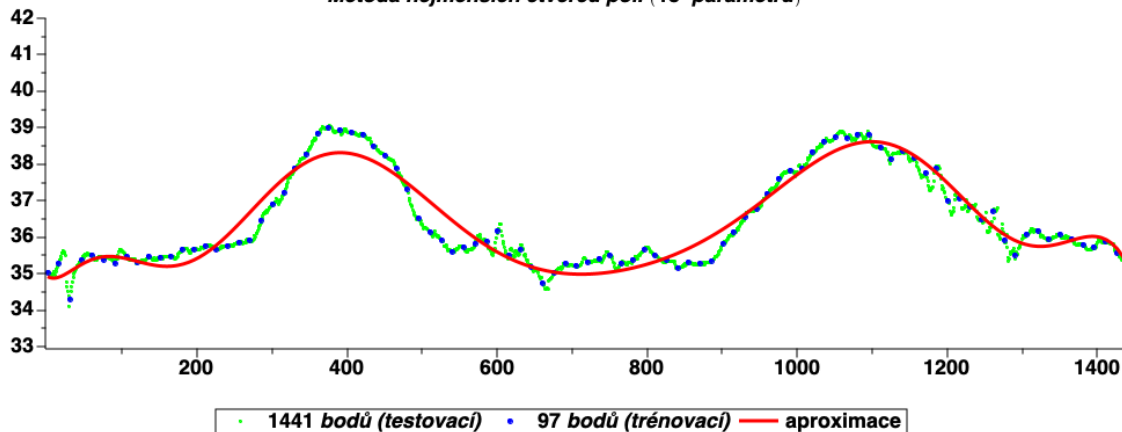
Interpolace splínem



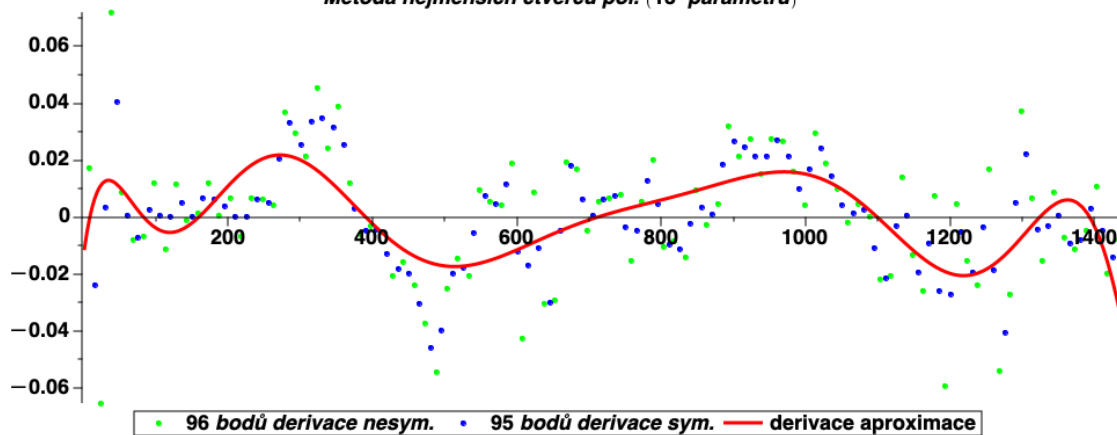
Interpolace splínem



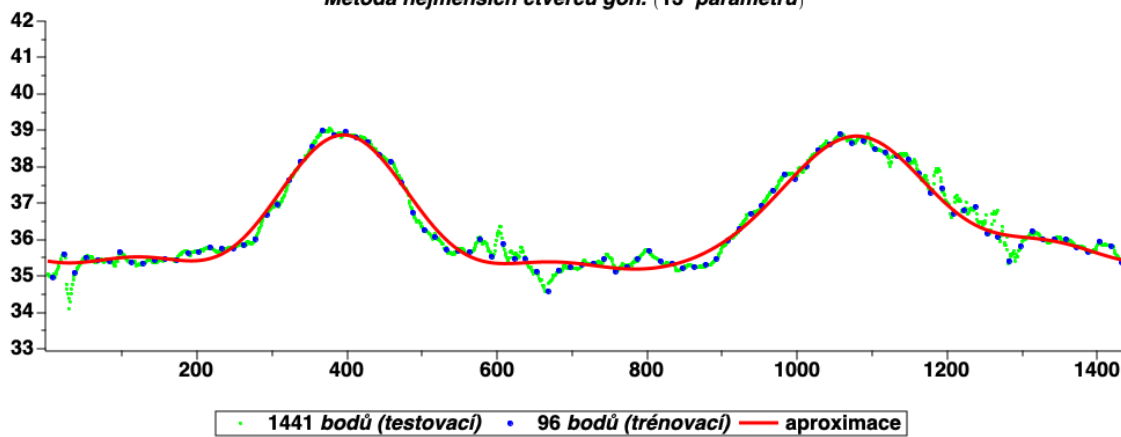
Metoda nejmenších čtverců pol. (13 parametrů)



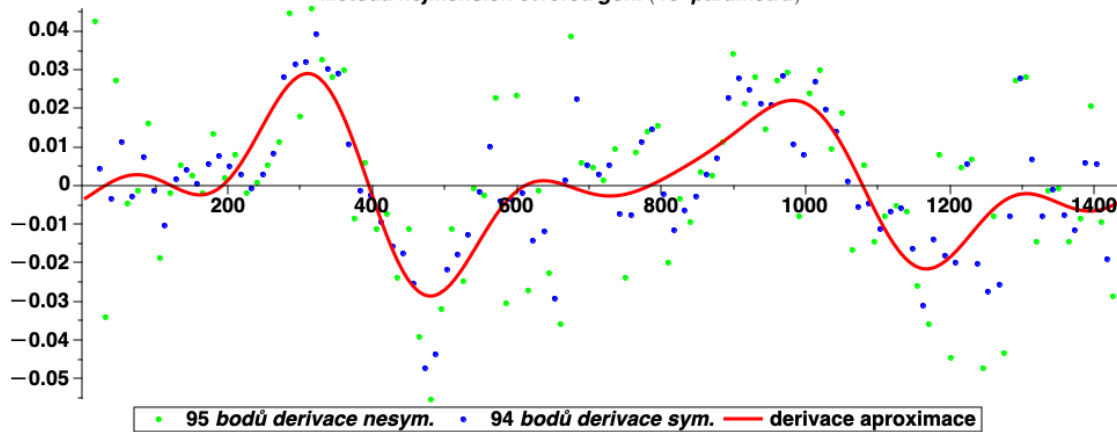
Metoda nejmenších čtverců pol. (13 parametrů)



Metoda nejmenších čtverců gon. (13 parametrů)

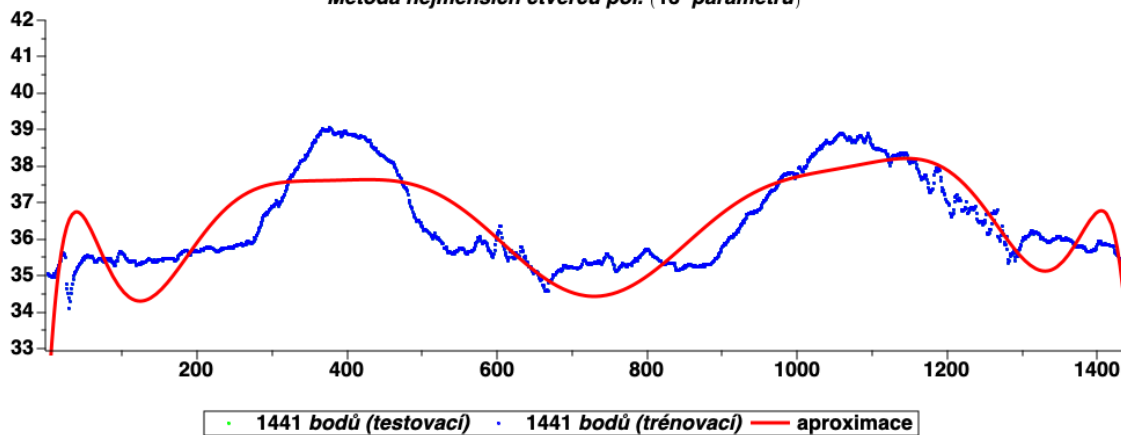


Metoda nejmenších čtverců gon. (13 parametrů)

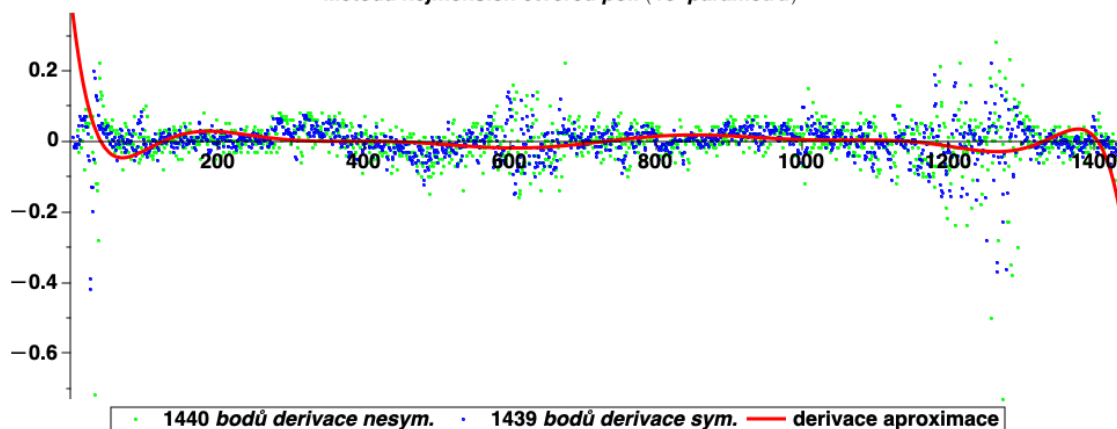




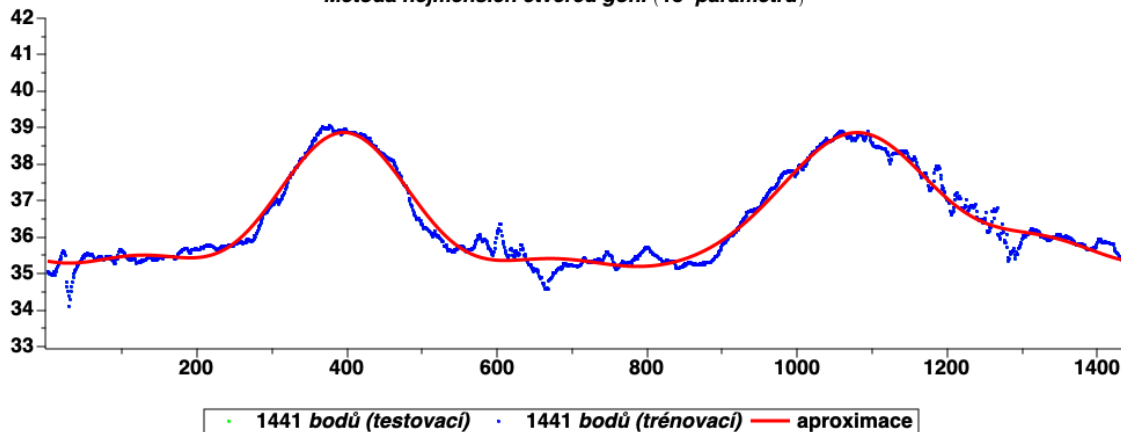
Metoda nejmenších čtverců pol. (13 parametrů)



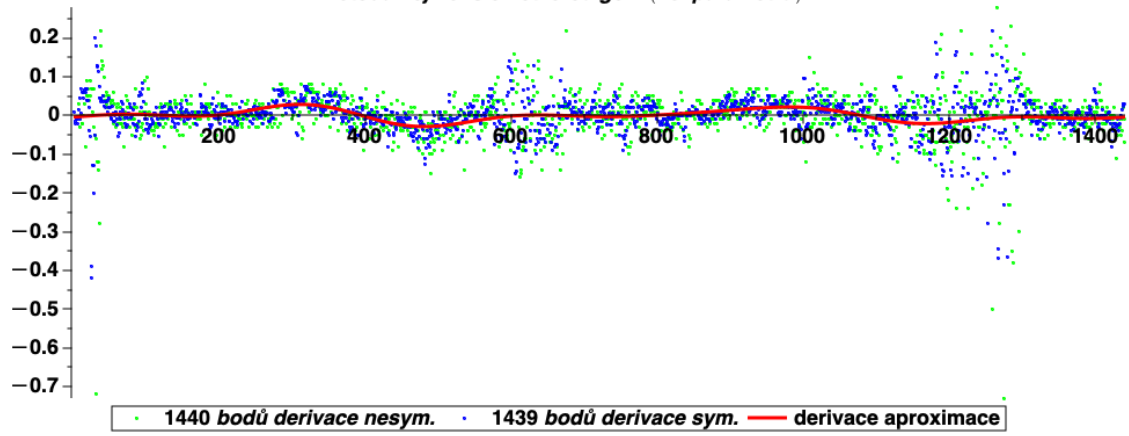
Metoda nejmenších čtverců pol. (13 parametrů)



Metoda nejmenších čtverců gon. (13 parametrů)



Metoda nejmenších čtverců gon. (13 parametrů)



## Dodatek: Přehled značení

Popis je zjednodušený a nemusí být přesný, podrobnosti jsou v textu. Značení použité jen lokálně zde není uvedeno.

### Značení specifické pro kapitoly 2 a 2

$f$  ... derivovaná funkce

$h$  ... krok při numerickém odhadu derivace

$\varphi$  ... funkce aproximující závislost na kroku

$d, d_n, d_s, d_{Rn}, d_{Rs}$  ... numerické odhady derivace,  $d(x, y) \approx f'(x)$  atd.

### Značení používané podobně v celém předmětu

$I(\dots)$  ... nejmenší interval obsahující čísla (body) v závorce, např.  $I(x_0, \dots, x_{n-1}) = \langle \min_i x_i, \max_i x_i \rangle$

$M_j$  ... horní odhad absolutní hodnoty  $j$ -té derivace derivované funkce,  $|f^{(j)}| \leq M_j$  na použitém intervalu

### 3 NUMERICKÁ INTEGRACE

**Úloha:** Odhadnout

$$I = \int_a^b f(t) dt$$

na základě hodnot funkce  $f$  v konečně mnoha uzlových bodech  $x_0, \dots, x_{n-1}$ .

Používá se někdy i v případech, kdy existuje analytické řešení.

**Příklad:**

$$\int_a^b \sin^{100} t dt$$

je náročnou úlohou pro počítačové algebraické systémy, ale z numerického hlediska není nijak zvlášť obtížný.

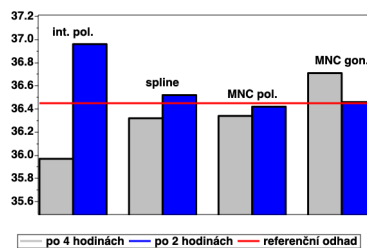
**Nápad 1:** Funkci  $f$  nahradíme aproximací  $\varphi$ , kterou umíme integrovat. Pokud je  $|f - \varphi| \leq \varepsilon$ ,

$$\begin{aligned} \left| I - \int_a^b \varphi(t) dt \right| &= \left| \int_a^b (f(t) - \varphi(t)) dt \right| \\ &\leq \int_a^b |f(t) - \varphi(t)| dt \leq (b - a) \varepsilon. \end{aligned}$$

**Příklad:** Vyjdeme z motivační úlohy (teplota pacienta).

Přesný výsledek neznáme, ale za jeho dobrý odhad můžeme považovat aritmetický průměr všech dat, 36.45.

metoda	krok	počet údajů	odhad integrálu
interpolační polynom	4 hodiny	7	35.97
interpolační polynom	2 hodiny	13	36.96
spline	4 hodiny	7	36.32
spline	2 hodiny	13	36.52
MNČ pol. (6 param.)	4 hodiny	6	36.34
MNČ pol. (6 param.)	2 hodiny	6	36.42
MNČ gon. (6 param.)	4 hodiny	12	36.71
MNČ gon. (6 param.)	2 hodiny	12	36.46
referenční odhad	1 minuta	1440	36.45



Složitost dle použité metody.

**Nápad 2:** Pokud  $X$  je náhodná veličina s rovnoměrným rozdělením na  $\langle a, b \rangle$ , tj. s hustotou  $f_X(t) = \frac{1}{b-a}$  na  $\langle a, b \rangle$  (neplést  $f$  s  $f_X$ ), pak je vzorec pro střední hodnotu

$$Ef(X) = \int_a^b f(t) f_X(t) dt = \int_a^b f(t) \frac{1}{b-a} dt = \frac{I}{b-a}.$$

Přibližný výsledek můžeme dostat simulací, kdy vybíráme  $t$  s rovnoměrným rozdělením na  $\langle a, b \rangle$  a použijeme výběrový průměr z funkčních hodnot  $f(t)$  (*metoda Monte Carlo*).

Spojité rovnoměrné rozdělení ale neumíme doopravdy simulovat, takže ho aproximujeme diskrétním.

Riskujeme, že mohou být vybrány hodnoty jen z malé části intervalu  $\langle a, b \rangle$  a výsledek může být hodně nepřesný.

Proto  $\langle a, b \rangle$  rozdělíme na  $k$  intervalů, nejlépe stejné délky,

$$\langle a_j, a_{j+1} \rangle, \quad j = 0, \dots, k-1,$$

kde  $a_0 = a$ ,  $a_k = b$ ,  $a_{j+1} - a_j = h = \frac{b-a}{k}$ , z každého vezmeme stejný počet vzorků. Dostaneme **jednoduchý vzorec**, tj. odhad  $I_j^A$  integrálu

$$I_j = \int_{a_j}^{a_{j+1}} f(t) dt.$$

Sečtením dostaneme **složený vzorec**, tj. odhad

$$I^A = \sum_{j < k} I_j^A$$

integrálu

$$I = \sum_{j < k} I_j = \int_a^b f(t) dt.$$

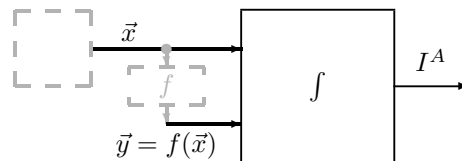
**Nápad 3:** Místo integrandu zintegrujeme jeho aproximaci interpolačním polynomem.

Místo náhrady v celém intervalu  $\langle a, b \rangle$  použijeme v dílčích intervalech náhradu polynomem nízkého stupně, vedoucí na *jednoduchý vzorec*.

**Linearita:** Integrál závisí na integrandu lineárně, proto odhad integrálu závisí lineárně na  $f(x_0), \dots, f(x_{n-1})$ :

$$I^A = \sum_{i < n} w_i f(x_i),$$

Můžeme volit pouze uzlové body  $x_0, \dots, x_{n-1}$  a jejich váhy  $w_0, \dots, w_{n-1}$ .



### 3.1 Newtonovy-Cotesovy vzorce

**Zjednodušení:** Všechny dílčí intervaly mají stejnou délku

$$h = \frac{b-a}{k} = a_{j+1} - a_j.$$

Každý dílčí interval lze lineární substitucí převést na jednotkový interval  $\langle 0, 1 \rangle$ . Obecný případ dostaneme lineární substitucí

$$u = \frac{t - a_j}{h}, \quad t = a_j + h u,$$

$$I_j = \int_{a_j}^{a_{j+1}} f(t) dt = \int_0^1 h f(a_j + h u) du = \int_0^1 g_j(u) du,$$

$$g_j(u) = h f(a_j + h u),$$

Budeme potřebovat její  $m$ -tou derivaci

$$g_j^{(m)}(u) = h^{m+1} f^{(m)}(a_j + h u).$$

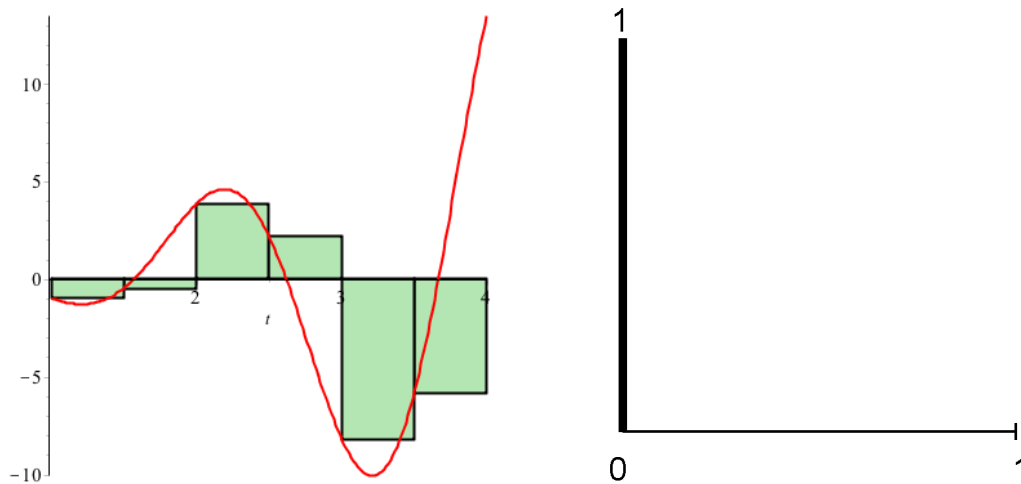
### 3.1.1 Metoda levých obdélníků

Jediný uzlový bod v krajním bodě intervalu,  $u_0 = 0$ ;  $g_j$  nahradíme konstantou  $g_j(u_0) = g_j(0)$ .  
Jednoduchý vzorec:

$$I_j^L = \int_0^1 g_j(0) dt = g_j(0) = h f(a_j).$$

Složený vzorec:

$$I^L = \sum_{j < k} I_j^L = h \sum_{j < k} f(a_j) = h \sum_{j < k} f(a + j h).$$



Rovnocenný je odhad pro volbu  $u_0 = 1$ , **metoda pravých obdélníků**.

### 3.1.2 Obdélníková metoda

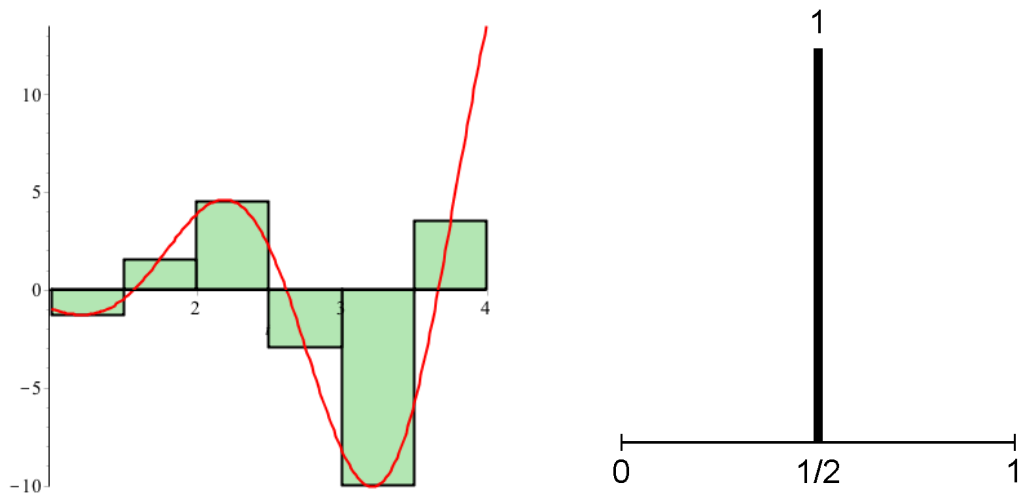
Uzlový bod ve středu intervalu,  $u_0 = 1/2$ .  
Proložíme konstantu  $g_j(u_0) = g_j(1/2)$ .  
Jednoduchý vzorec:

$$I_j^R = \int_0^1 g_j(1/2) dt = g_j(1/2) = h f(a_j + h/2).$$

Složený vzorec:

$$I^R = \sum_{j < k} I_j^R = h \sum_{j < k} f(a_j + h/2) = h \sum_{j < k} f(a_{1/2} + j h),$$

kde  $a_{1/2} = a + h/2$ .



### 3.1.3 Lichoběžníková metoda

Dva uzlové body na krajích intervalu,  $u_0 = 0, u_1 = 1$ .

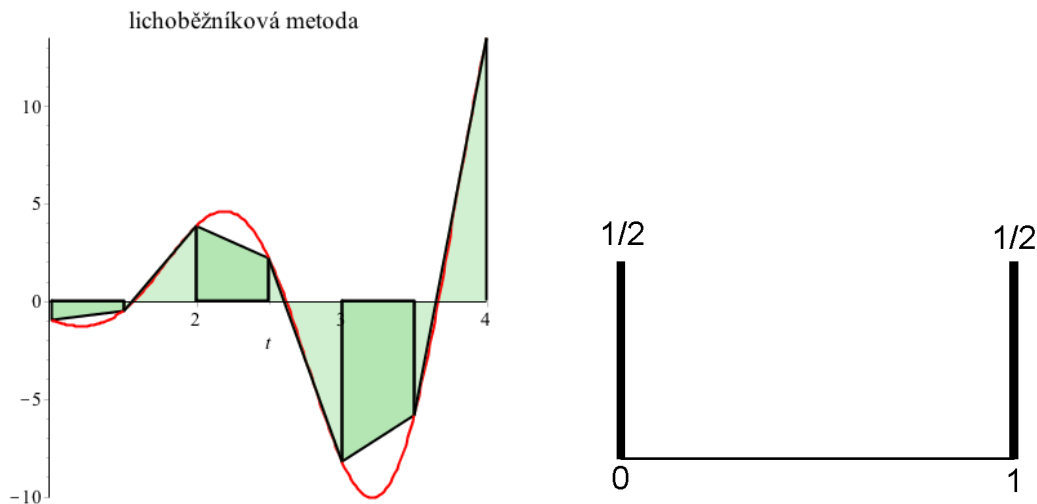
Proložíme lineární funkci, výsledkem bude plocha pod přímkou, neboli obsah lichoběžníka.

Jednoduchý vzorec:

$$I_j^T = \frac{g_j(u_0) + g_j(u_1)}{2} = \frac{g_j(0) + g_j(1)}{2} = h \frac{f(a_j) + f(a_{j+1})}{2}.$$

Složený vzorec:

$$I^T = \sum_{j < k} I_j^T = h \sum_{j < k} \frac{f(a_j) + f(a_{j+1})}{2} = h \left( \frac{f(a) + f(b)}{2} + \sum_{j=1}^{k-1} f(a + j h) \right).$$



Ekvivalentní výklad je, že funkci  $g_j$  nahrazujeme

- v levé polovině intervalu,  $\langle 0, 1/2 \rangle$  hodnotou v levém krajním bodě,  $g_j(0)$ ,
- v pravé polovině intervalu,  $(1/2, 1)$  hodnotou v pravém krajním bodě,  $g_j(1)$ .

### 3.1.4 Simpsonova metoda

Tři uzlové body; dva na krajích intervalu, jeden uprostřed,  $u_0 = 0, u_1 = 1/2, u_2 = 1$ . Proložíme kvadratický polynom a zintegrujeme.

Jednoduchý vzorec:

$$\begin{aligned} I_j^S &= w_0 g_j(u_0) + w_1 g_j(u_1) + w_2 g_j(u_2) \\ &= w_0 g_j(0) + w_1 g_j(1/2) + w_2 g_j(1). \end{aligned}$$

Vzorec bude přesný, bude-li  $g_j$  libovolný kvadratický polynom. Speciálně pro  $g_j(u) \in \{1, u, u^2\}$ :

$$\begin{aligned} w_0 + w_1 + w_2 &= \int_0^1 1 \, du = 1, \\ \frac{1}{2} w_1 + w_2 &= \int_0^1 u \, du = \frac{1}{2}, \\ \frac{1}{4} w_1 + w_2 &= \int_0^1 u^2 \, du = \frac{1}{3}. \end{aligned}$$

To je soustava 3 lineárních rovnic pro 3 neznámé  $w_0, w_1, w_2$ , řešení:

$$w_0 = \frac{1}{6}, \quad w_1 = \frac{2}{3}, \quad w_2 = \frac{1}{6}.$$

Ekvivalentní výklad je, že funkci  $g_j$  nahrazujeme

- na intervalu  $\langle 0, 1/6 \rangle$  hodnotou v levém krajním bodě,  $g_j(0)$ ,
- na intervalu  $\langle 1/6, 5/6 \rangle$  hodnotou ve středu,  $g_j(1/2)$ ,
- na intervalu  $\langle 5/6, 1 \rangle$  hodnotou v pravém krajním bodě,  $g_j(1)$ .

Jednoduchý vzorec:

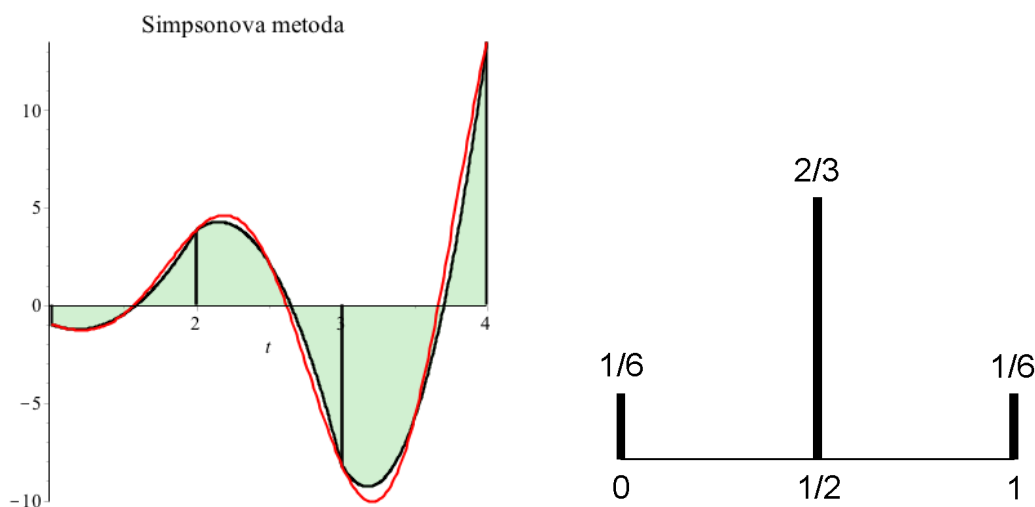
$$\begin{aligned} I_j^S &= \int_0^1 g_j(1/2) dt = \frac{1}{6} g_j(0) + \frac{2}{3} g_j(1/2) + \frac{1}{6} g_j(1) \\ &= \frac{h}{6} (f(a_j) + 4 f(a_j + h/2) + f(a_{j+1})) . \end{aligned}$$

Složený vzorec (kde  $a_{1/2} = a + h/2$ ; **pozor na meze sum!**):

$$\begin{aligned} I^S = \sum_{j < k} I_j^S &= \frac{h}{6} \left( f(a_0) + f(a_k) + 2 \sum_{j=1}^{k-1} f(a_j) + 4 \sum_{j=0}^{k-1} f(a_j + h/2) \right) \\ &= \frac{h}{6} \left( f(a) + f(b) + 2 \sum_{j=1}^{k-1} f(a + j h) + 4 \sum_{j=0}^{k-1} f(a_{1/2} + j h) \right) . \end{aligned}$$

$$I^S = \frac{h_2}{3} (f(x_0) + 4 f(x_1) + 2 f(x_2) + 4 f(x_3) + \dots + 4 f(x_{2k-1}) + f(x_{2k})) ,$$

kde  $x_i = a + i h_2$  jsou uzlové body (pro funkci  $f$ , nikoli  $g_j$ ) a  $h_2 = h/2$  je vzdálenost mezi sousedními uzlovými body. Počet intervalů délky  $h_2$  musí být sudý!



### 3.1.5 Obecné Newtonovy-Cotesovy vzorce

- **otevřené** (obdélníková metoda)
- **uzavřené** (lichoběžníková a Simpsonova metoda)
- **polootvřené** (metoda levých obdélníků)

## 3.2 Odhad chyby numerické integrace

**Zjednodušení:** pro lichoběžníkovou metodu.

Předpokládejme, že  $g_j$  má na intervalu  $\langle 0, 1 \rangle$  spojitou druhou derivaci. Funkci  $g_j$  nahrazujeme lineárním polynomem  $\varphi_j$ ; chyba interpolace v bodě  $u$  je

$$|g_j(u) - \varphi_j(u)| \leq \frac{\max_{v \in (0,1)} |g_j''(v)|}{2} |(u-0)(u-1)| ,$$



$$\begin{aligned}
|I_j^T - I_j| &= \left| \int_0^1 \varphi_j(u) \, du - \int_0^1 g_j(u) \, du \right| \leq \int_0^1 |\varphi_j(u) - g_j(u)| \, du \\
&\leq \frac{\max_{v \in (0,1)} |g_j''(v)|}{2} \int_0^1 (u - u^2) \, du = \frac{\max_{v \in (0,1)} |g_j''(v)|}{2} \left( \frac{1}{2} - \frac{1}{3} \right) \\
&= \frac{1}{12} \max_{v \in (0,1)} |g_j''(v)| = \frac{1}{12} h^3 \max_{t \in (a_j, a_{j+1})} |f''(t)|.
\end{aligned}$$

Vyjádříme pomocí

$$M_2 \geq \max_{t \in (a,b)} |f''(t)|,$$

$$|I_j^T - I_j| \leq \frac{1}{12} h^3 M_2,$$

složený vzorec

$$|I^T - I| \leq \frac{k}{12} h^3 M_2,$$

po náhradě konstantního součinu  $kh = b - a$

$$|I^T - I| \leq \frac{(b-a) M_2}{12} h^2.$$

### 3.3 Řád metod integrace

**Definice 3.1** *Nechť funkce  $I^A$  vyjadřuje výsledek integrační metody v závislosti na délce kroku,  $I$  je správný výsledek. Pokud existuje*

$$\lim_{h \rightarrow 0} \frac{\ln |I^A(h) - I|}{\ln h},$$

*nazývá se **řád metody**.*

**Poznámka 3.1** *V logaritmických souřadnicích má řád metody význam směrnice asymptoty v  $(-\infty, -\infty)$ .*

Platí analogie věty 2.1:

**Věta 3.1** *Nechť funkce  $I^A$  vyjadřuje výsledek integrační metody v závislosti na délce kroku,  $I$  je správný výsledek. Nechť  $p$  je exponent u  $h$  v nejnižším nenulovém členu Taylorova rozvoje chyby metody podle  $h$  v okolí bodu 0. Pokud má  $I^A$  spojitou  $p$ -tou derivaci v 0, pak  $p$  je řád metody ve smyslu Definice 3.1.*

metoda	horní odhad chyby	řád
levých obdélníků	$\frac{(b-a) M_1}{2} h$	1
lichoběžníková	$\frac{(b-a) M_2}{12} h^2$	2
obdélníková	$\frac{(b-a) M_2}{24} h^2$	2
Simpsonova	$\frac{(b-a) M_4}{2880} h^4 = \frac{(b-a) M_4}{180} h_2^4$	4

Simpsonova metoda dává chybu nikoli třetího, ale čtvrtého řádu. Je-li  $f$ , a tedy i  $g_j$ , polynom stupně nejvýše 3, pak chyba interpolace kvadratickým polynomem je úměrná

$$W(u) = (u - 0)(u - 1/2)(u - 1).$$

Na hodnotě integrálu se to neprojeví, neboť

$$\int_0^1 W(u) \, du = 0.$$

**Příklad 1:**

$$I = \int_0^2 e^{-t^2} dt$$

s přesností  $\varepsilon = 10^{-6}$ . Stanovte postačující počet kroků pro jednotlivé metody.

met.	$M_p$	horní odhad $h$	počet kroků
$I^L$	$\sqrt{\frac{2}{e}} \doteq 0.86$	$\frac{2 \cdot 10^{-6}}{2 \cdot 0.86} \doteq 1.16 \cdot 10^{-6}$	1 720 000
$I^T$	2	$\sqrt{\frac{12 \cdot 10^{-6}}{2 \cdot 2}} \doteq 1.7 \cdot 10^{-3}$	1155
$I^R$	2	$\sqrt{\frac{24 \cdot 10^{-6}}{2 \cdot 2}} \doteq 2.45 \cdot 10^{-3}$	817
$I^S$	12	$\sqrt[4]{\frac{2880 \cdot 10^{-6}}{2 \cdot 12}} \doteq 0.1$	20

Pro  $k = 20$

$$I^L \doteq 0.9311046,$$

$$I^R \doteq 0.8821118,$$

$$I^T \doteq 0.8820204,$$

$$I^S \doteq 0.8820813,$$

$$\int_0^2 \exp(-t^2) dt \doteq 0.882081390.$$

### 3.4 Numerická integrace reálných dat

Vyjdeme z motivační úlohy (teplota pacienta).

Přesný výsledek neznáme, ale za jeho dobrý odhad můžeme považovat aritmetický průměr všech dat, 36.45.

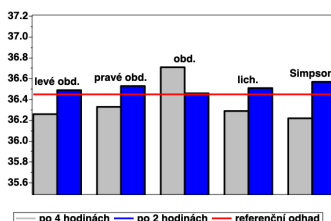
Zkusíme ho odhadnout z podstatně menšího počtu dat.

To bude pohodlnější pro pacienta i pro personál.

Obrázky by zde moc nového neřekly.

metoda	krok	interval měření	počet údajů	odhad integrálu
levých obdélníků	4 hodiny	4 hodiny	6	36.26
levých obdélníků	2 hodiny	2 hodiny	12	36.49
pravých obdélníků	4 hodiny	4 hodiny	6	36.33
pravých obdélníků	2 hodiny	2 hodiny	12	36.53
obdélníková	4 hodiny	4 hodiny	6	36.71
obdélníková	2 hodiny	2 hodiny	12	36.46
lichoběžníková	4 hodiny	4 hodiny	7	36.29
lichoběžníková	2 hodiny	2 hodiny	13	36.51
Simpsonova	8 hodin	4 hodiny	7	36.22
Simpsonova	4 hodiny	2 hodiny	13	36.57
referenční odhad	1 minuta	1 minuta	1440	36.45

Zde se výhoda Simpsonovy metody moc neuplatnila; přesnost jejího výsledku se opírala o 2× více dat.



To jsme dosáhli s **lineární** složitostí.

Pro **průběh** integrálu by bylo lépe použít aproximaci v celém rozsahu.

### 3.5 Gaussova metoda integrace

Na intervalu  $\langle -1, 1 \rangle$  volíme za uzlové body kořeny  $z_0, \dots, z_{s-1} \in \langle -1, 1 \rangle$  tzv. **Legendreových polynomů**. Lineární transformací

$$u = \frac{z + 1}{2}, \quad z = 2u - 1$$

dostaneme uzlové body  $u_0, \dots, u_{s-1} \in (0, 1)$ . Uzlové body a jejich váhy  $w_0, \dots, w_{s-1}$  jsou tabelovány nebo raději počítány algoritmem. Volíme pouze jejich počet  $s$ , a tím i řád metody.

$s$	uzlové body	váhy
1	0	2
2	$\pm \frac{1}{\sqrt{3}} \doteq \pm 0.577350$	1
3	$\pm \sqrt{\frac{3}{5}} \doteq \pm 0.774597$	$\frac{5}{9}$
	0	$\frac{8}{9}$
4	$\pm \sqrt{\frac{30+4\sqrt{30}}{70}} \doteq \pm 0.861136$	0.347855
	$\pm \sqrt{\frac{30-4\sqrt{30}}{70}} \doteq \pm 0.339981$	0.652145
5	$\pm \sqrt{\frac{70+4\sqrt{70}}{126}} \doteq \pm 0.906180$	0.236927
	$\pm \sqrt{\frac{70-4\sqrt{70}}{126}} \doteq \pm 0.538469$	0.478629
	0	0.568889

Jednoduchý vzorec:

$$I_j^{G(s)} = \sum_{i < s} w_i g_j(u_i) = h \sum_{i < s} w_i f(a_j + h u_i).$$

Složený vzorec:

$$I^{G(s)} = \sum_{j < k} I_j^{G(s)} = h \sum_{j < k} \sum_{i < s} w_i f(a_j + h u_i) = h \sum_{i < s} \left( w_i \sum_{j < k} f(d_i + j h) \right),$$

kde  $d_i = a + h u_i \in \langle a_0, a_1 \rangle$ . Horní odhad chyby

$$|I^{G(s)} - I| \leq \frac{(b-a)(s!)^4 M_{2s}}{(2s+1)((2s)!)^3} h^{2s},$$

kde

$$M_{2s} \geq \max_{v \in \langle a, b \rangle} |f^{(2s)}(v)|.$$

Chyba metody je řádu  $2s$ , díky volbě  $s$  uzlových bodů a  $s$  vah, tj.  $2s$  parametrů. V Newtonových-Cotesových vzorcích jsme volbou  $s$  vah (při daných uzlových bodech) dostali metody řádu  $s$  nebo  $s + 1$ .

počet uzlových bodů	horní odhad chyby	řád
1	$\frac{(b-a) M_2}{24} h^2$	2
2	$\frac{(b-a) M_4}{4320} h^4$	4
3	$\frac{(b-a) M_6}{2\,016\,000} h^6$	6
4	$\frac{(b-a) M_8}{1\,778\,112\,000} h^8$	8
5	$\frac{(b-a) M_{10}}{2\,534\,876\,467\,200} h^{10}$	10

**Příklad 1 (pokračování):**

$$I = \int_0^2 e^{-t^2} dt$$

s přesností  $\varepsilon = 10^{-6}$ .

$I^A$	$M_p$	horní odhad $h$	p. kroků
$I^L$	$\sqrt{\frac{2}{e}} \doteq 0.86$	$\frac{2 \cdot 10^{-6}}{2 \cdot 0.86} \doteq 1.16 \cdot 10^{-6}$	1 720 000
$I^T$	2	$\sqrt{\frac{12 \cdot 10^{-6}}{2 \cdot 2}} \doteq 1.7 \cdot 10^{-3}$	1155
$I^R$	2	$\sqrt{\frac{24 \cdot 10^{-6}}{2 \cdot 2}} \doteq 2.45 \cdot 10^{-3}$	817
$I^S$	12	$\sqrt[4]{\frac{2880 \cdot 10^{-6}}{2 \cdot 12}} \doteq 0.1$	20
$I^{G(2)}$	12	$\sqrt[4]{\frac{4320 \cdot 10^{-6}}{2 \cdot 12}} \doteq 0.115$	18
$I^{G(4)}$	1680	$\sqrt[8]{\frac{1\,778\,112\,000 \cdot 10^{-6}}{2 \cdot 1680}} \doteq 0.92$	3

### 3.6 Richardsonova extrapolace při integraci

Richardsonovou extrapolací dostaneme nový odhad

$$I^B(h) = \frac{q^p I^A(\frac{h}{q}) - I^A(h)}{q^p - 1} = I^A(\frac{h}{q}) + \frac{I^A(\frac{h}{q}) - I^A(h)}{q^p - 1}.$$

Výraz

$$\frac{I^A(\frac{h}{q}) - I^A(h)}{q^p - 1} \doteq I - I^A(\frac{h}{q})$$

můžeme rovněž považovat za odhad chyby výsledku  $I^A(\frac{h}{q})$ .

Speciálně pro  $q = 2$  (**metoda polovičního kroku**):

$$I^B(h) = \frac{2^p I^A(\frac{h}{2}) - I^A(h)}{2^p - 1} = I^A(\frac{h}{2}) + \frac{I^A(\frac{h}{2}) - I^A(h)}{2^p - 1},$$

$$I - I^A(\frac{h}{2}) \doteq \frac{I^A(\frac{h}{2}) - I^A(h)}{2^p - 1}.$$

Extrapolaci nemusíme provádět až ve výsledném (složeném) vzorci, že ji přenést do jednoduchého vzorce a dokonce až do jeho vah. Jednoduché vzorce pro kroky  $h$  a  $h/q$  (kde  $q$  je celé číslo  $> 1$ ) a výsledek Richardsonovy extrapolace lze napsat ve tvaru

$$I_j^A(h) = \sum_i w_i g_j(u_i),$$

$$I_j^A(h/q) = \sum_i \bar{w}_i g_j(u_i),$$

$$I_j^B(h) = \sum_i \bar{\bar{w}}_i g_j(u_i),$$

kde  $u_i$  označují všechny uzlové body potřebné pro obě délky kroků (nejsou-li některé uzlové body použity, odpovídající váhy jsou nulové). Index  $j$  označuje pořadí intervalu při kroku  $h$ , ostatní výsledky jsou tomu přizpůsobeny.

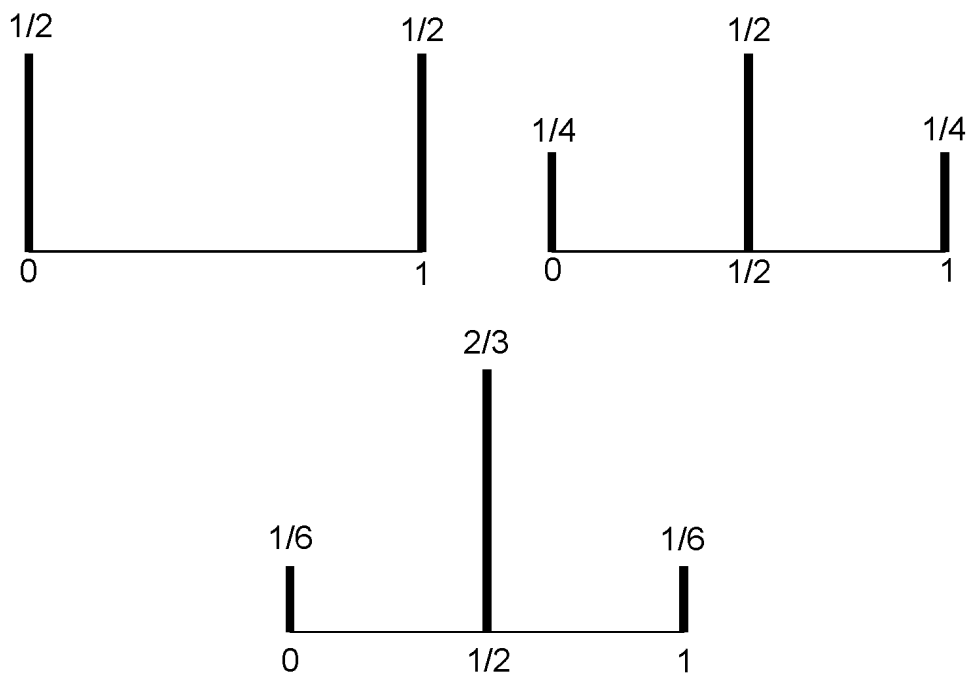
$$I_j^B(h) = \frac{q^p I_j^A(\frac{h}{q}) - I_j^A(h)}{q^p - 1} = \frac{q^p \sum_i \bar{w}_i g_j(u_i) - \sum_i w_i g_j(u_i)}{q^p - 1}$$

$$= \sum_i \frac{q^p \bar{w}_i - w_i}{q^p - 1} g_j(u_i) = \sum_i \bar{\bar{w}}_i g_j(u_i),$$

$$\bar{\bar{w}}_i = \frac{q^p \bar{w}_i - w_i}{q^p - 1},$$

což odpovídá Richardsonově extrapolaci uplatněné na váhy.

Pro lichoběžníkovou metodu lze doporučit  $q = 2$



Polovina nových uzlových bodů (pro krok  $h/2$ ) se kryje se starými (pro krok  $h$ ); dostaneme odhad

$$I^T\left(\frac{h}{2}\right) + \frac{I^T\left(\frac{h}{2}\right) - I^T(h)}{3} = \frac{2I^R(h) + I^T(h)}{3},$$

shodný se Simpsonovou metodou.

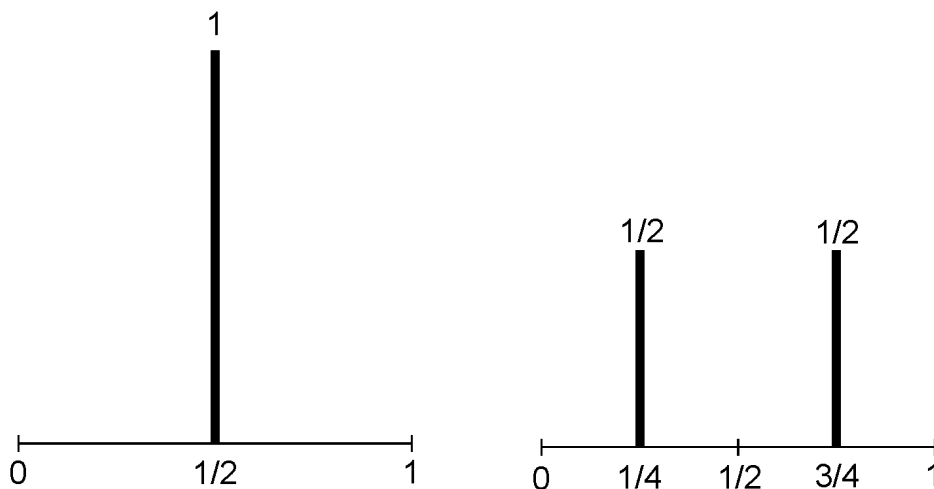
Richardsonovou extrapolací lze zpřesnit i Simpsonovu metodu, dostaneme odhad 6. řádu

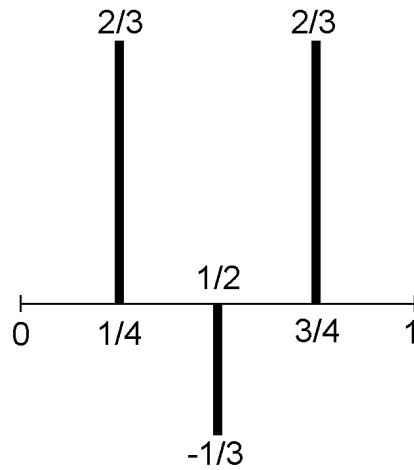
$$I^S\left(\frac{h}{2}\right) + \frac{I^S\left(\frac{h}{2}\right) - I^S(h)}{15}.$$

Richardsonovou extrapolací pro obdélníkovou metodu s polovičním krokem dostaneme odhad

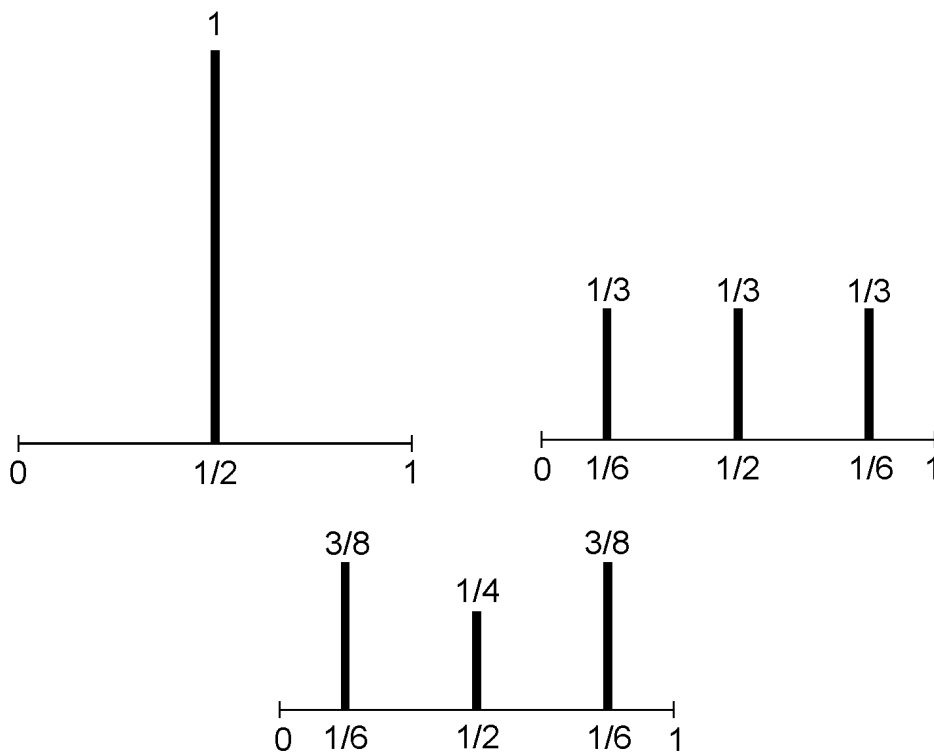
$$I^R\left(\frac{h}{2}\right) + \frac{I^R\left(\frac{h}{2}\right) - I^R(h)}{3},$$

který se však nehodí:





Vhodnější je třetinový krok,  $q = 3$ ,  $I^R(\frac{h}{3}) + \frac{I^R(\frac{h}{3}) - I^R(h)}{8}$ .



### 3.6.1 Rombergova metoda

Vychází z více odhadů získaných lichoběžníkovou metodou pro kroky  $h, h/2, h/4, \dots$

Taylorův rozvoj chyby lichoběžníkové metody má nenulové pouze členy sudého řádu. Proto se každou Richardsonovou extrapolací zvýší řád o dvě.

řád	2	4	6	8
$k$	$I^T(\frac{h}{k})$	$I^S(\frac{2h}{k})$		
$k_0$	$I^T(h) = I_{0,0}^T$			
$2k_0$	$I^T(\frac{h}{2}) = I_{1,0}^T$	$\searrow$ $\rightarrow$	$I_{1,1}^T$	
$4k_0$	$I^T(\frac{h}{4}) = I_{2,0}^T$	$\searrow$ $\rightarrow$	$I_{2,1}^T$	$\rightarrow$ $\searrow$
$8k_0$	$I^T(\frac{h}{8}) = I_{3,0}^T$	$\searrow$ $\rightarrow$	$I_{3,1}^T$	$\rightarrow$ $\searrow$
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$ $\searrow$
				$I_{3,3}^T$

Obecně ve sloupci  $j + 1$ :

$$I_{i,j}^T = I_{i,j-1}^T + \frac{I_{i,j-1}^T - I_{i-1,j-1}^T}{4^j - 1}.$$

Za výsledek bereme  $I_{i,i}^T$ , chyba je řádu  $2^{-i}$  a odhadujeme ji zhruba výrazem  $|I_{i,i-1}^T - I_{i-1,i-1}^T|$  nebo  $|I_{i,i}^T - I_{i-1,i-1}^T|$ .

**Příklad 1 (pokračování):** Výsledky Rombergovy metody pro  $\int_0^2 e^{-t^2} dt$  s počáteční volbou 4 intervalů dělení:

řád	2	4	6	8
$k$	$I^T(\frac{h}{k})$	$I^S(\frac{2h}{k})$		
4	0.88061			
8	0.88170	0.8820655		
16	0.88170	0.8820803	<i>0.88208139</i>	
32	0.88205	<i>0.8820813</i>	<i>0.88208138</i>	<i>0.88208138</i>

S platnými ciframi *0.882081* se shodují výsledky vyznačené kurzívou.

Zde je výhoda Simpsonovy metody (sloupec s výsledky 4. řádu) zjevná.

**Příklad 2:** Výsledky Rombergovy metody pro  $\int_0^\pi \sin^4 t dt$ , s počáteční volbou 1 intervalu dělení:

řád	2	4	6	8	10
$k$	$I^T(\frac{h}{k})$	$I^S(\frac{2h}{k})$			
1	0				
2	1.57080	2.09440			
4	<i>1.17810</i>	1.0472	0.97738		
8	<i>1.17810</i>	<i>1.17810</i>	1.18683	1.19015	
16	<i>1.17809</i>	<i>1.17809</i>	<i>1.17809</i>	<i>1.17795</i>	<i>1.17790</i>

S platnými ciframi *1.178* se shodují výsledky vyznačené kurzívou. Zde extrapolace nebyla úspěšná, naopak zhoršila výsledky. Důvodem je nevhodně zvolený malý počet intervalů na začátku.

### 3.7 Praktické stanovení počtu intervalů

- Z horního odhadu chyby.
- Metoda dvojího (nejčastěji polovičního) kroku.

**Příklad 1 (pokračování):**

$$I = \int_0^2 e^{-t^2} dt$$

s přesností  $\varepsilon = 10^{-6}$ . Simpsonova metoda s kroky 2 a 1:

$$\begin{aligned} I^S(2) &\doteq 0.8299444, \\ I^S(1) &\doteq 0.8818124. \end{aligned}$$

Odhad chyby metodou polovičního kroku je

$$\frac{|I^S(1) - I^S(2)|}{15} \doteq 0.0034578,$$

požadovaná chyba je zhruba  $3458 \times$  menší, což vyžaduje zvýšit počet kroků v poměru alespoň  $\sqrt[4]{3458} \doteq 7.7$ . Pro  $4 \times$  a  $8 \times$  menší krok, tj. pro 8 a 16 intervalů dělení:

$$\begin{aligned} I^S\left(\frac{2}{8}\right) &\doteq 0.882080396576, \\ I^S\left(\frac{2}{16}\right) &\doteq 0.882081328646. \end{aligned}$$

Odhad chyby posledního výsledku je

$$\frac{|I^S\left(\frac{2}{16}\right) - I^S\left(\frac{2}{8}\right)|}{15} \doteq 6.3 \cdot 10^{-8}.$$

(Již víme, že postačuje 20 intervalů dělení.) Richardsonova extrapolace:

$$I^S\left(\frac{2}{16}\right) + \frac{I^S\left(\frac{2}{16}\right) - I^S\left(\frac{2}{8}\right)}{15} \doteq 0.882081390784,$$

Přesnější výsledek je

$$0.8820813907624216800.$$

**Příklad 3:**

$$I = \int_0^1 \frac{\sin t}{\sqrt{t}} dt$$

s přesností  $\varepsilon = 10^{-8}$ . Zkusíme 5-bodovou Gaussovou metodu (10. řádu) s kroky 1 a  $\frac{1}{2}$ :

$$\begin{aligned} I^{G(5)}(1) &\doteq 0.621166517, \\ I^{G(5)}\left(\frac{1}{2}\right) &\doteq 0.620759367. \end{aligned}$$

Odhad chyby metodou polovičního kroku je

$$\frac{|I^{G(5)}\left(\frac{1}{2}\right) - I^{G(5)}(1)|}{2^{10} - 1} \doteq 4 \cdot 10^{-7},$$

požadovaná chyba je zhruba  $40 \times$  menší, což vyžaduje zvýšit počet kroků v poměru alespoň  $\sqrt[10]{40} \doteq 1.5$ . Měl by tedy stačit  $2 \times$  menší krok, tj. 4 intervaly dělení:

$$\begin{aligned} I^{G(5)}\left(\frac{1}{2}\right) &\doteq 0.620759367 \\ I^{G(5)}\left(\frac{1}{4}\right) &\doteq 0.620615367. \end{aligned}$$

Odhad chyby posledního výsledku je

$$\frac{|I^{G(5)}\left(\frac{1}{4}\right) - I^{G(5)}\left(\frac{1}{2}\right)|}{2^{10} - 1} \doteq 1.4 \cdot 10^{-7},$$

tedy jen asi třikrát menší, ač se měl zmenšit v poměru  $2^{10} = 1024$ .

Přesnější výsledek je

$$0.62053660344676220362.$$

## 3.8 Řešení obtížnějších úloh úpravou zadání

### 3.8.1 Integrace přes nekonečný interval

**Příklad 4:**

$$I = \int_2^\infty e^{-t^2} dt$$

s přesností  $\varepsilon = 10^{-6}$ .



I nekonečný obor integrace lze (nelineární) substitucí převést na konečný  $\langle 0, 1 \rangle$ , zde např.  $t = 1/u$ :

$$I = \int_0^{\frac{1}{2}} \frac{e^{-\frac{1}{u^2}}}{u^2} du.$$

Můžeme využít známé určité integrály, např.

$$I = \int_2^{\infty} e^{-t^2} dt = \underbrace{\int_0^{\infty} e^{-t^2} dt}_{\sqrt{\pi}/2} - \int_0^2 e^{-t^2} dt,$$

Můžeme se omezit na konečný interval a zbytek zanedbat. V našem případě lze použít odhad (se substitucí  $t - x = u$ )

$$\begin{aligned} \int_x^{\infty} e^{-t^2} dt &= e^{-x^2} \int_0^{\infty} e^{-2xu - u^2} du \\ &\leq e^{-x^2} \int_0^{\infty} e^{-2xu} du = \frac{e^{-x^2}}{2x}. \end{aligned}$$

Pro  $x \geq 3.85$  je tento výraz menší než  $\frac{\varepsilon}{2}$ , takže stačí vypočítat

$$\int_2^{3.85} e^{-t^2} dt$$

s přesností  $\frac{\varepsilon}{2}$ .

### 3.8.2 Omezení intervalu

se může hodit, i když obor integrace je konečný:

**Příklad 5:**

$$\int_2^{1000} e^{-t^2} dt$$

Simpsonovou metodou s 1000 kroky:

$$0.0043821,$$

4-bodovou Gaussovou metodou se 100 kroky:

$$0.0012304,$$

Dopustíme se chyby menší než  $\frac{\varepsilon}{2} = 5 \cdot 10^{-7}$ , snížíme-li horní mez na 3.85. Pak stačí Simpsonova metoda s 23 kroky

$$0.00414549.$$

Přesnější výsledek je

$$0.00414553469.$$

### 3.8.3 Pomalu konvergentní integrály

**Přičtení známého určitého integrálu** může zásadně změnit obtížnost numerického výpočtu:

**Příklad 3 (pokračování):**

$$\int_0^1 \frac{\sin t}{\sqrt{t}} dt$$

Integrand má v okolí nuly neomezenou derivaci. V okolí nuly je  $\sin t \approx t$ ,  $\frac{\sin t}{\sqrt{t}} \approx \sqrt{t}$ . Derivace je sice nadále neomezená, ale známe

$$\int_0^1 \sqrt{t} dt = \frac{2}{3} t^{\frac{3}{2}}.$$

Rozdíl  $\frac{\sin t}{\sqrt{t}} - \sqrt{t}$  má derivace omezené a jeho integrace nečiní zvláštní potíže. Výpočet 5-bodovou Gaussovou metodu (10. řádu) se dvěma a čtyřmi intervaly dělení dává

$$\begin{aligned} \int_0^1 \left( \frac{\sin t}{\sqrt{t}} - \sqrt{t} \right) dt &\doteq \\ I^{G(5)}\left(\frac{1}{2}\right) &\doteq -0.046130081752, \\ I^{G(5)}\left(\frac{1}{4}\right) &\doteq -0.046130064858. \end{aligned}$$

Odhad chyby metodou polovičního kroku:

$$\frac{|I^{G(5)}\left(\frac{1}{4}\right) - I^{G(5)}\left(\frac{1}{2}\right)|}{2^{10} - 1} \doteq 1.7 \cdot 10^{-11}.$$

Přesnější výsledek je

$$-0.04613006321990446305$$

Výsledek původního zadání je

$$\begin{aligned} \int_0^1 \frac{\sin t}{\sqrt{t}} dt &= \int_0^1 \sqrt{t} dt + \int_0^1 \left( \frac{\sin t}{\sqrt{t}} - \sqrt{t} \right) dt \\ &\doteq \frac{2}{3} - 0.04613006486 \\ &\doteq 0.62053660181 \end{aligned}$$

(přesněji 0.6205366034467622036).

**Substituce** funkcí, která má v odpovídajícím bodě  $c$  nulové derivace dostatečně mnoha řádů, např.  $t = c + u^s$ , kde exponent  $s$  volíme raději vyšší než nižší.

**Příklad 3 (pokračování):**

$$\int_0^1 \frac{\sin t}{\sqrt{t}} dt$$

Substitucí  $t = u^2$  dostaneme

$$\int_0^1 \frac{\sin t}{\sqrt{t}} dt = \int_0^1 2 \sin u^2 du.$$

5-bodová Gaussova metoda (10. řádu) s jedním a dvěma intervaly dělení:

$$\begin{aligned} I^{G(5)}(1) &\doteq 0.620536620796, \\ I^{G(5)}\left(\frac{1}{2}\right) &\doteq 0.620536603496, \end{aligned}$$

odhad chyby metodou polovičního kroku

$$\frac{|I^{G(5)}\left(\frac{1}{2}\right) - I^{G(5)}(1)|}{2^{10} - 1} \doteq 1.7 \cdot 10^{-11}.$$

Chtěli bychom, aby se integrand v okolí problémového bodu blížil konstantě; mohli jsme použít též substituci  $\frac{2}{3}t^{\frac{3}{2}} = u$  s dobrým výsledkem.

**Příklad 6:**

$$\int_1^\infty \frac{\sin t}{\sqrt{t}} dt$$

s přesností  $10^{-8}$ . Omezení na konečný obor nepomůže, neboť např.

$$\int_{999997}^{1000000} \frac{\sin t}{\sqrt{t}} dt \doteq 0.0019.$$

Hledaný integrál není absolutně konvergentní, nicméně

$$\int_0^{\infty} \frac{\sin t}{\sqrt{t}} dt = \lim_{M \rightarrow \infty} \int_0^M \frac{\sin t}{\sqrt{t}} dt = \sqrt{\frac{\pi}{2}}.$$

Potřebujeme

$$\begin{aligned} \int_1^{\infty} \frac{\sin t}{\sqrt{t}} dt &= \int_0^{\infty} \frac{\sin t}{\sqrt{t}} dt - \int_0^1 \frac{\sin t}{\sqrt{t}} dt \\ &\doteq \sqrt{\frac{\pi}{2}} - 0.620536601808 \doteq 0.632777535507, \end{aligned}$$

kde ovšem integrál  $\int_0^1 \frac{\sin t}{\sqrt{t}} dt$  byl rovněž problémový; využili jsme řešení příkladu 3. Přesnější výsledek je

$$0.6327775338746013102.$$

## Dodatek: Přehled značení

Popis je zjednodušený a nemusí být přesný, podrobnosti jsou v textu. Značení použité jen lokálně zde není uvedeno.

### Značení specifické pro tuto kapitolu

$I = \int_a^b f(t) dt$  ... hledaný integrál

$k$  ... počet kroků numerické integrace

$h$  ... délka kroku numerické integrace

$a_0 = a, a_1, a_2, \dots, a_k = b$  ... dělicí body intervalu  $\langle a, b \rangle$

$h = 2h = \frac{b-a}{k} = a_{j+1} - a_j$  ... délka kroku numerické integrace

$I_j = \int_{a_j}^{a_{j+1}} f(t) dt, j = 1, 2, \dots$  ... dílčí integrály (výsledky jednoduchých vzorců)

$g_j(u) = h f(a_j + hu)$  ... integrand po lineární transformaci argumentu  $\langle a_j, a_{j+1} \rangle \rightarrow [0, 1]$

$I^A, I^L, I^R, I^T, I^S, I^{G(s)}$  ... aproximace integrálu jednotlivými metodami (obecná, levých obdélníků, obdélníková, lichoběžníková, Simpsonova, Gaussova  $s$ -bodová);  $s$  indexem  $j$  výsledky jednoduchých vzorců na  $\langle a_j, a_{j+1} \rangle$ ;

následuje-li argument v závorce, je to délka kroku

$u_0, u_1, \dots$  ... uzlové body na intervalu  $[0, 1]$

$w_0, w_1, \dots$  ... váhy uzlových bodů na intervalu  $[0, 1]$

$x_0, x_1, \dots$  ... uzlové body na intervalu  $\langle a, b \rangle$

$p$  ... řád metody

### Značení používané podobně v celém předmětu

$M_j$  ... horní odhad absolutní hodnoty  $j$ -té derivace funkce,  $|f^{(j)}| \leq M_j$  na použitém intervalu

## 4 NUMERICKÉ ŘEŠENÍ NELINEÁRNÍCH ROVNIC

### 4.1 Formulace problému

**Úloha:** Hledáme reálné řešení rovnice  $f(x) = 0$ , kde  $f$  je spojitá reálná funkce na intervalu  $\langle a_0, b_0 \rangle$ . Nutno upřesnit:

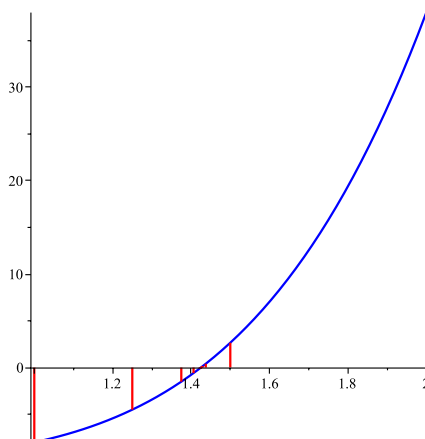
**Úloha:** Hledáme reálné řešení rovnice  $f(x) = 0$ , kde  $f$  je spojitá funkce na intervalu  $\langle a_0, b_0 \rangle$ . Přitom předpokládáme, že  $f(a_0) \cdot f(b_0) < 0$  (tj.  $f(a_0), f(b_0)$  mají opačná znaménka) a že  $f$  má v intervalu  $\langle a_0, b_0 \rangle$  právě jeden kořen,  $\bar{x}$ . Řešení máme stanovit s danou přesností  $\varepsilon > 0$ , tj. máme najít nějakou hodnotu, která se nalézá v intervalu  $\langle \bar{x} - \varepsilon, \bar{x} + \varepsilon \rangle$ .

Tomu předchází **separace kořenů**, která není algoritmizovatelná.

### 4.2 Metoda půlení intervalu neboli bisekce

$$x_i = \frac{a_i + b_i}{2}$$

- je-li  $f(x_i) \cdot f(a_i) < 0$ , pak  $a_{i+1} = a_i$ ,  $b_{i+1} = x_i$ ,
- je-li  $f(x_i) \cdot f(b_i) < 0$ , pak  $a_{i+1} = x_i$ ,  $b_{i+1} = b_i$ ,
- je-li  $f(x_i) = 0$ , pak  $\bar{x} = x_i$ .



**Podmínka ukončení:**

$$\frac{b_i - a_i}{2} \leq \varepsilon \quad (3)$$

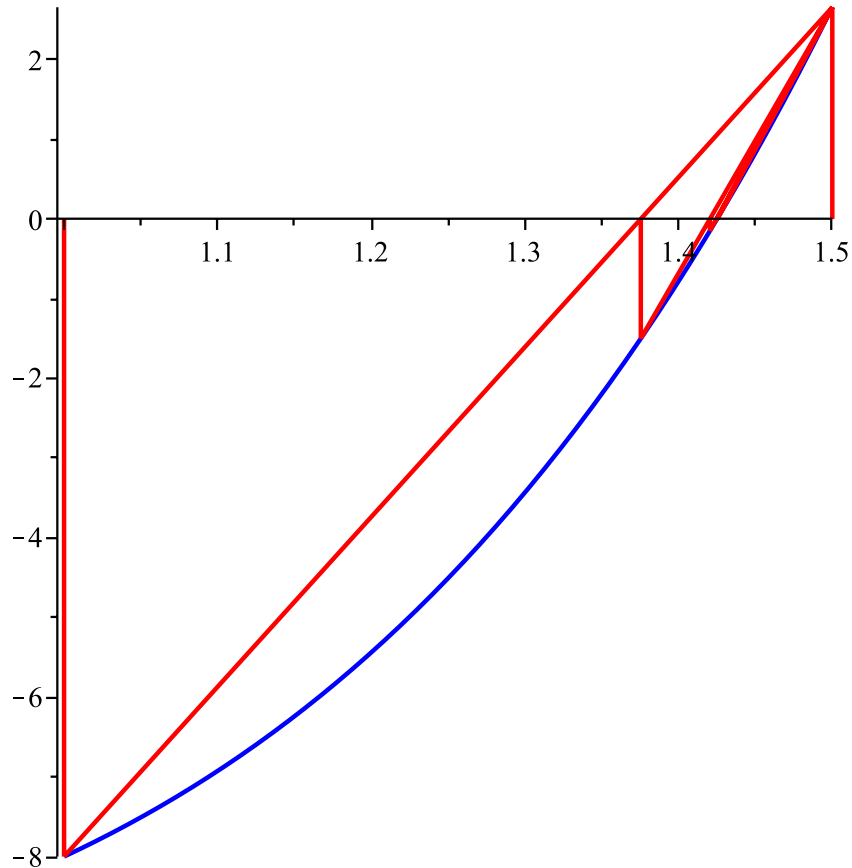
Konverguje vždy stejně rychle:  
zpřesnění o 3 desetinná místa během 10 kroků

### 4.3 Metoda regula falsi

Interval  $\langle a_i, b_i \rangle$  rozdělíme v poměru  $\frac{|f(a_i)|}{|f(b_i)|}$ , tj. sečnou dle (1), nový dělicí bod  $x_i$  je její nulové místo:

$$\frac{x_i - a_i}{x_i - b_i} = \frac{f(a_i)}{f(b_i)},$$

$$x_i = \frac{a_i f(b_i) - b_i f(a_i)}{f(b_i) - f(a_i)}.$$



Typicky se od  $j$ -tého kroku jeden krajní bod intervalu nemění (např. pokud  $f''$  nemění znaménko).

$$b_i - a_i \not\rightarrow 0$$

$$\lim_{i \rightarrow \infty} (b_i - a_i) \in \{|\bar{x} - a_j|, |\bar{x} - b_j|\}$$

**Podmínka ukončení:**

$$|f(x_i)| \leq \delta \quad (4)$$

#### 4.4 Univerzální odhad chyby

Taylorův rozvoj funkce  $f$  se středem  $x_i$  vyhodnotíme v bodě  $\bar{x}$ :

$$\underbrace{f(\bar{x})}_0 = f(x_i) + (\bar{x} - x_i) f'(\theta_i)$$

pro nějaké  $\theta_i \in I(x_i, \bar{x})$

$$\bar{x} - x_i = \frac{-f(x_i)}{f'(\theta_i)}$$

Pokud  $\exists m_1 > 0 \forall x \in I(x_i, \bar{x}) : m_1 \leq |f'(x)|$ ,  
přechodem k absolutním hodnotám dostaneme

$$|\bar{x} - x_i| \leq \frac{|f(x_i)|}{m_1}$$

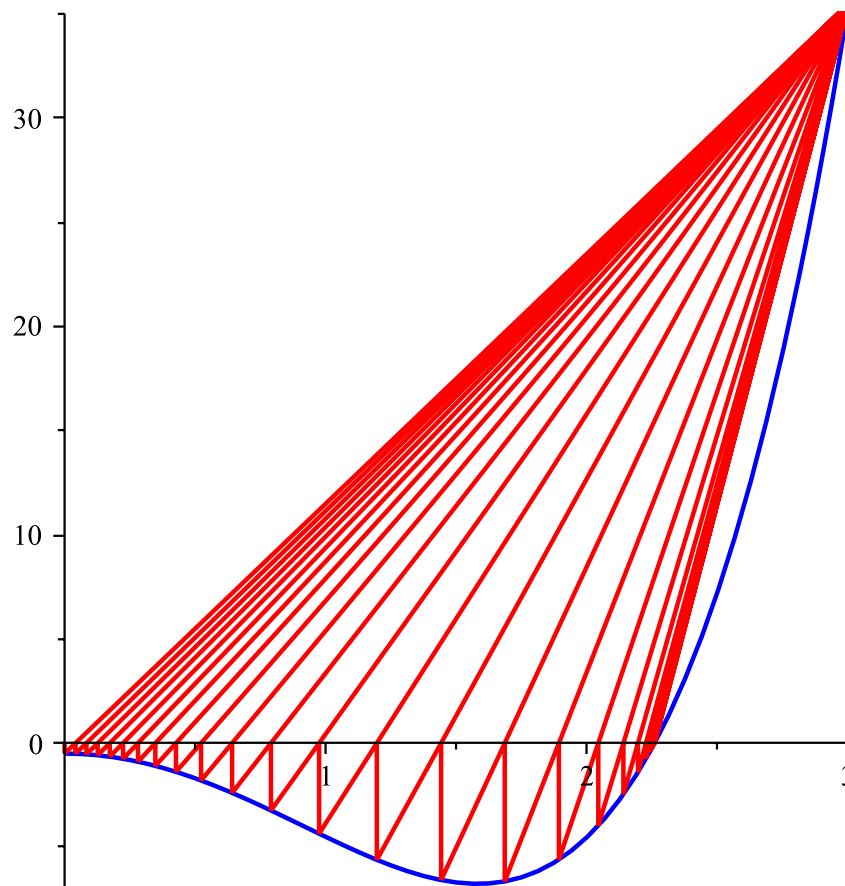
**Věta 4.1** *Nechť funkce  $f$  má na intervalu  $I(x_i, \bar{x})$  spojitou derivaci a*

$$\exists m_1 > 0 \forall t \in I(x_i, \bar{x}) : m_1 \leq |f'(t)|.$$

*Pak*

$$|\bar{x} - x_i| \leq \frac{|f(x_i)|}{m_1} \leq \frac{\delta}{m_1}.$$

Větu nelze použít, neexistuje-li derivace nebo je-li kořen násobný (metoda stále může být použitelná). Metoda regula falsi konverguje rychleji, pokud zadaná funkce je (v okolí kořene) přibližně lineární.

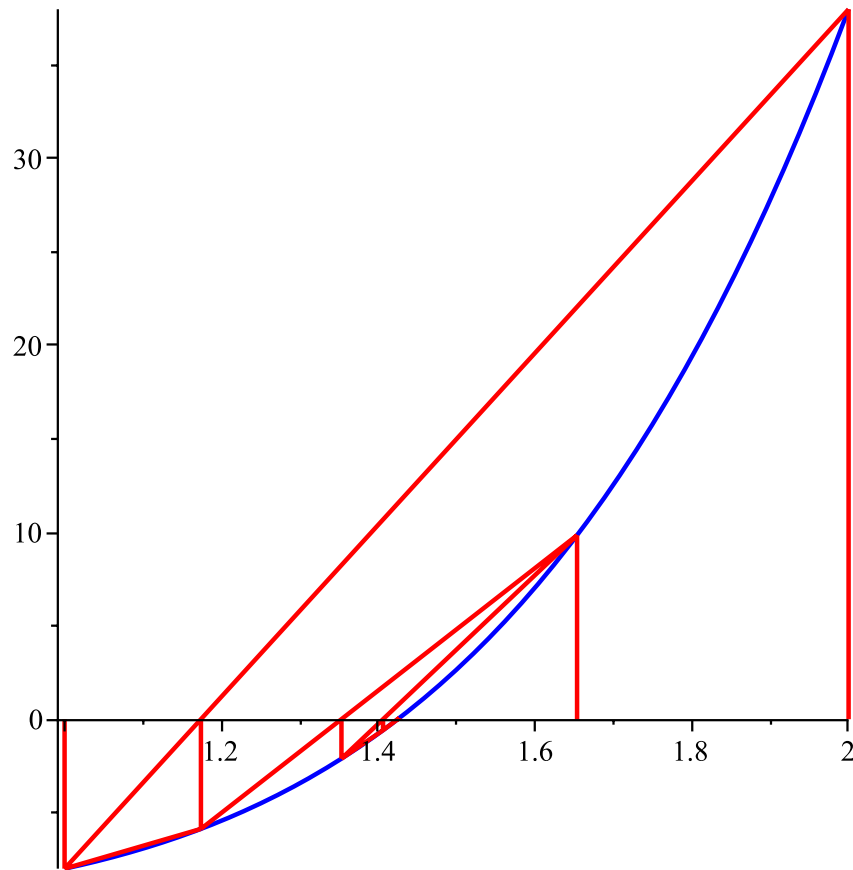


#### 4.5 Metoda sečen

Modifikace metody regula falsi: pro další výpočet vždy použijeme vzorec (1) pro dva posledně vypočtené body:

$$x_0 = b_0, \quad x_1 = a_0$$

$$x_i = \frac{x_{i-2}f(x_{i-1}) - x_{i-1}f(x_{i-2})}{f(x_{i-1}) - f(x_{i-2})}$$



**Podmínka ukončení:**

$$|f(x_i)| \leq \delta \quad (5)$$

nebo

$$|x_i - x_{i-1}| \leq \eta \quad (6)$$

Konvergence bývá rychlejší, ale není zaručena.

## 4.6 Newtonova metoda (metoda tečen)

Metody

- **jednobodové**
- **dvoubodové**
- **vícebodové**

Tečna  $t_{i-1}$  ke grafu funkce  $f$  v bodě  $(x_{i-1}, f(x_{i-1}))$ :

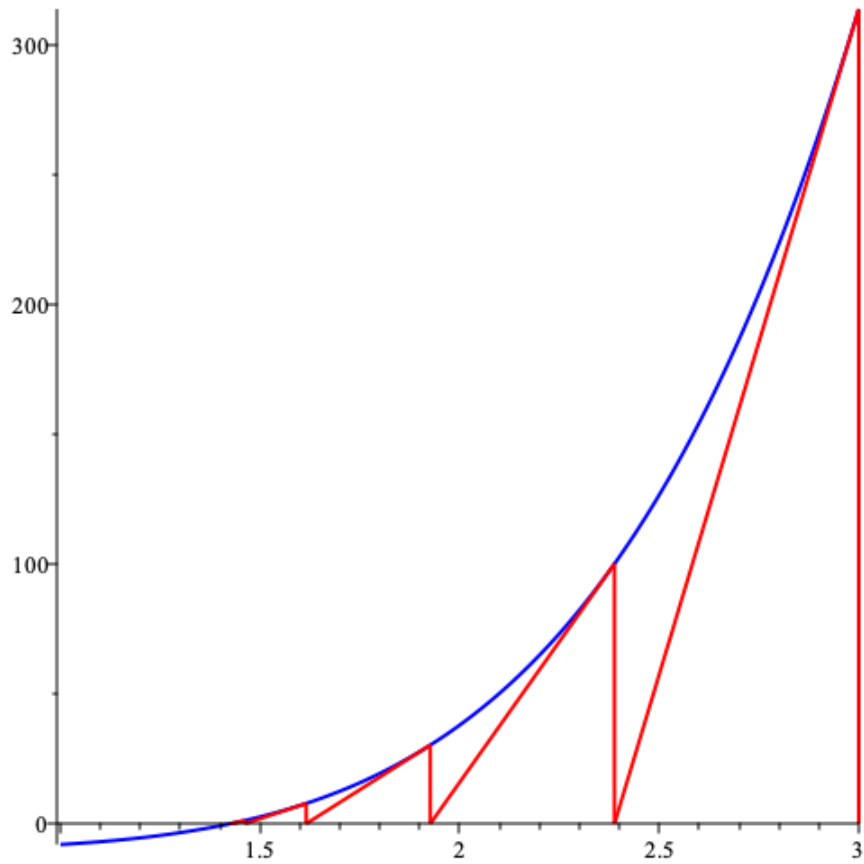
$$t_{i-1}(x) = f(x_{i-1}) + (x - x_{i-1}) \cdot f'(x_{i-1}),$$

$x_i$  je její nulový bod:

$$x_i = x_{i-1} - \frac{f(x_{i-1})}{f'(x_{i-1})}.$$

Předpokládá existenci a **znalost** první derivace, nutno ošetřit případné přetečení nebo dělení nulou.





**Podmínka ukončení:**

$$|f(x_i)| \leq \delta \quad (7)$$

nebo

$$|x_i - x_{i-1}| \leq \eta \quad (8)$$

Konvergence bývá rychlejší, ale není zaručena.

#### 4.6.1 Odhad chyby Newtonovy metody

Taylorův rozvoj funkce  $f$  se středem  $x_{i-1}$  vyhodnotíme v bodě  $x_i$ :

$$f(x_i) = \underbrace{f(x_{i-1}) + (x_i - x_{i-1}) f'(x_{i-1})}_0 + \frac{1}{2} (x_i - x_{i-1})^2 f''(\xi_i),$$

kde  $\xi_i \in I(x_i, x_{i-1})$ . Dosadíme do univerzálního odhadu:

$$\bar{x} - x_i = \frac{-f(x_i)}{f'(\theta_i)} = \frac{-f''(\xi_i)}{2f'(\theta_i)} (x_i - x_{i-1})^2, \quad \theta_i \in I(x_i, \bar{x}).$$

Pokud lze najít odhady

$$\begin{aligned} \exists M_2 \forall x \in I(x_i, x_{i-1}) : & |f''(x)| \leq M_2 \\ \exists m_1 > 0 \forall x \in I(x_i, \bar{x}) : & |f'(x)| \geq m_1 \end{aligned}$$

pak přechodem k absolutním hodnotám dostaneme

$$|\bar{x} - x_i| \leq \frac{M_2}{2m_1} (x_i - x_{i-1})^2 \quad (9)$$

## Odhad chyby Newtonovy metody

**Věta 4.2** Necht  $\bar{x}$  je **jednoduchý** kořen funkce  $f$ , která má na intervalu  $I(x_i, x_{i-1}, \bar{x})$  (kde  $x_i$  je výsledek jednoho kroku Newtonovy metody aplikované na odhad  $x_{i-1}$ ) spojitou druhou derivaci. Necht existují reálná čísla  $M_2, m_1 > 0$  taková, že

$$\begin{aligned} \forall x \in I(x_i, \bar{x}) : & |f'(x)| \geq m_1 \\ \forall x \in I(x_i, x_{i-1}) : & |f''(x)| \leq M_2 \end{aligned}$$

Pak platí odhad chyby (9)

$$|\bar{x} - x_i| \leq \frac{M_2}{2m_1} (x_i - x_{i-1})^2$$

**Důsledek 4.1** Při splnění podmínky ukončení (8)  $|x_i - x_{i-1}| \leq \eta$  dostáváme odhad chyby

$$|\bar{x} - x_i| \leq \frac{M_2}{2m_1} \eta^2$$

**Jednoduché pravidlo:** pokud Newtonova metoda konverguje a aproximace se nachází v blízkosti kořene, v každém kroku se zhruba zdvojnásobí počet míst za desetinnou čárkou, která jsou správně vypočtená.

**Správně:** pokud je chyba mnohem menší než 1 a absolutní hodnoty první a druhé derivace funkce  $f$  jsou přibližně stejně velké, pak činitel  $\frac{M_2}{2m_1}$  můžeme zanedbat a pravidlo platí, neboť

$$|\bar{x} - x_i| \approx (x_i - x_{i-1})^2 \approx (\bar{x} - x_{i-1})^2$$

Pokud se však poměr  $\frac{M_2}{2m_1}$  hodně liší od jednotky, pravidlo nemůžeme použít.

### 4.6.2 Konvergence Newtonovy metody

Není zaručena. Metoda může divergovat zejména při špatném počátečním odhadu.

**Předpoklad:**  $f''$  je spojitá v okolí **jednoduchého** kořene  $\bar{x}$ .

Pak  $f'(\bar{x}) \neq 0$  a  $f'$  je spojitá v okolí  $\bar{x}$

$\implies$  lze najít uzavřené okolí  $I$  bodu  $\bar{x}$  takové, že

$$\begin{aligned} \exists m_1 > 0 \forall x \in I : & |f'(x)| \geq m_1 \\ \exists M_2 \forall x \in I : & |f''(x)| \leq M_2 \end{aligned}$$

Necht  $x_{i-1} \in I \setminus \{\bar{x}\}$ .

Taylorův rozvoj funkce  $f$  se středem  $x_{i-1}$  vyhodnotíme v bodě  $\bar{x}$ :

$$\underbrace{f(\bar{x})}_0 = f(x_{i-1}) + (\bar{x} - x_{i-1}) f'(x_{i-1}) + \frac{1}{2} (\bar{x} - x_{i-1})^2 f''(\xi_i),$$

kde  $\xi_i \in I(\bar{x}, x_{i-1})$ . Odečteme

$$0 = f(x_{i-1}) + (x_i - x_{i-1}) f'(x_{i-1})$$

$$\begin{aligned} 0 &= (\bar{x} - x_i) f'(x_{i-1}) + \frac{1}{2} (\bar{x} - x_{i-1})^2 f''(\xi_i) \\ \bar{x} - x_i &= -\frac{f''(\xi_i)}{2f'(x_{i-1})} (\bar{x} - x_{i-1})^2 \\ \frac{\bar{x} - x_i}{(\bar{x} - x_{i-1})^2} &= -\frac{f''(\xi_i)}{2f'(x_{i-1})} \\ \frac{|\bar{x} - x_i|}{(\bar{x} - x_{i-1})^2} &\leq \frac{M_2}{2m_1} \\ \frac{|\bar{x} - x_i|}{|\bar{x} - x_{i-1}|} &\leq \frac{M_2}{2m_1} |\bar{x} - x_{i-1}| \end{aligned}$$

## Konvergence Newtonovy metody

Pro  $x_{i-1}$  dostatečně blízko  $\bar{x}$ :

$$\begin{aligned}\frac{M_2}{2m_1} |\bar{x} - x_{i-1}| &\leq q \\ \frac{|\bar{x} - x_i|}{|\bar{x} - x_{i-1}|} &\leq q\end{aligned}$$

pro nějaké (předem dané)  $q < 1$ , tj. chyba se v jednom kroku zmenší v poměru aspoň  $q$  a metoda konverguje.

**Věta 4.3** *Nechť funkce  $f$  má spojitou druhou derivaci v okolí **jednoduchého** kořene  $\bar{x}$ . Pak Newtonova metoda konverguje v nějakém okolí kořene  $\bar{x}$ .*

### 4.6.3 Náhrada derivace numerickým odhadem

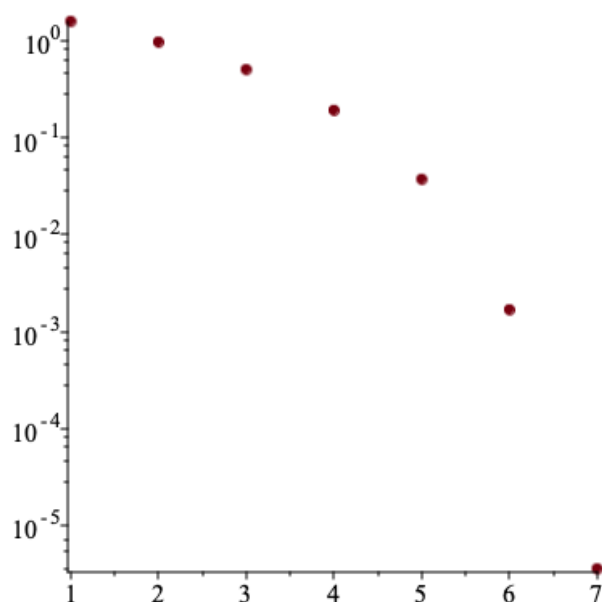
Alternativou je numerický výpočet derivace, který zde lze začlenit do metody.

Výpočtu dalších funkčních hodnot se vyhneme použitím posledních dvou vypočtených,  $f(x_{i-1})$ ,  $f(x_{i-2})$ ; derivaci nahradíme směrnici sečny:

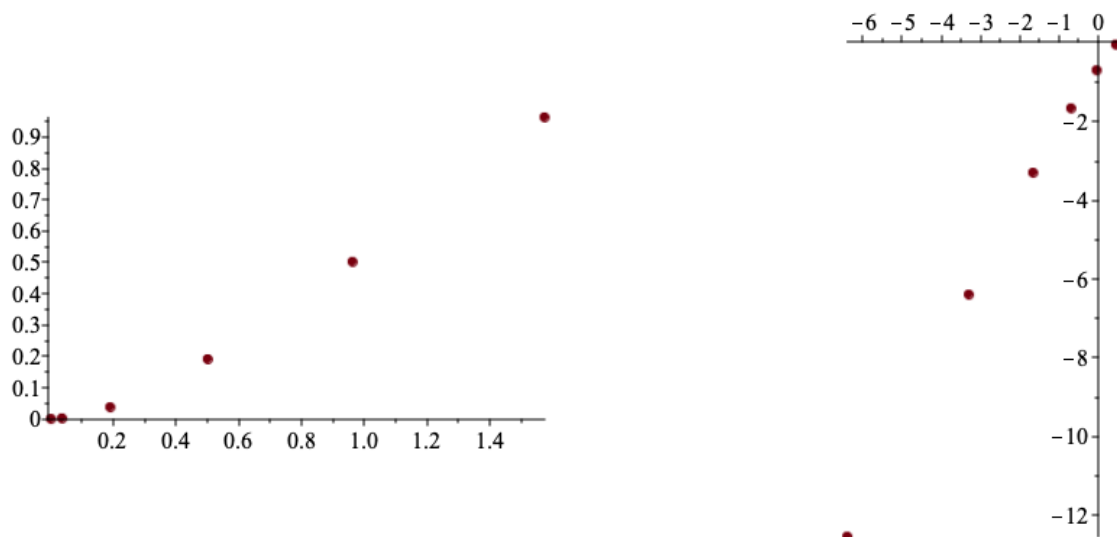
$$\begin{aligned}f'(x_{i-1}) &\approx \frac{f(x_{i-1}) - f(x_{i-2})}{x_{i-1} - x_{i-2}} \\ x_i &= x_{i-1} - \frac{f(x_{i-1})}{\frac{f(x_{i-1}) - f(x_{i-2})}{x_{i-1} - x_{i-2}}} = \frac{x_{i-2}f(x_{i-1}) - x_{i-1}f(x_{i-2})}{f(x_{i-1}) - f(x_{i-2})}\end{aligned}$$

Nic nového pod sluncem: metoda sečen (ale myšlenka byla správná).

## 4.7 Konvergence a její rychlost (řád metody)



Závislost logaritmu chyby na kroku pro Newtonovu metodu



Chyba v posledním kroku (svisle) a v předposledním kroku (vodorovně) pro Newtonovu metodu; vlevo lineární měřítko, vpravo logaritmické (skutečná chyba je nahrazena rozdílem posledních dvou iterací)

**Konvergence** závisí na

$$L(1) := \lim_{i \rightarrow \infty} \frac{|\bar{x} - x_i|}{|\bar{x} - x_{i-1}|}.$$

Platí:

$L(1) < 1 \implies$  metoda konverguje,

$L(1) > 1 \implies$  metoda diverguje.

**Rychlost konvergence** závisí na **řádu metody**,

$$p = \lim_{i \rightarrow \infty} \frac{\ln |\bar{x} - x_i|}{\ln |\bar{x} - x_{i-1}|}.$$

**Poznámka 4.1** Vyneseme-li body  $(|\bar{x} - x_i|, |\bar{x} - x_{i-1}|)$  v logaritmických souřadnicích, má řád metody význam směrnice asymptoty v  $(-\infty, -\infty)$ .

**Věta 4.4** Necht metoda řešení rovnice  $f(x) = 0$  řádu  $p$  dává za výsledek posloupnost aproximací  $x_i$ ,  $i \in \mathbb{N}$ , konvergující ke kořeni  $\bar{x}$ . Necht  $r > 0$ ,

$$L(r) := \lim_{i \rightarrow \infty} \frac{|\bar{x} - x_i|}{|\bar{x} - x_{i-1}|^r}. \quad (10)$$

Pak

$$r < p \implies L(r) = 0,$$

$$r > p \implies L(r) = \infty,$$

$$r = p \implies L(r) = L(p) \text{ může (ale nemusí) existovat a může být konečná a nenulová.}$$

**Důkaz.** Limitu zlogaritmujeme:

$$\ell(r) := \ln L(r) = \lim_{i \rightarrow \infty} \ln \frac{|\bar{x} - x_i|}{|\bar{x} - x_{i-1}|^r} = \lim_{i \rightarrow \infty} \left( \ln |\bar{x} - x_i| - \underbrace{r \ln |\bar{x} - x_{i-1}|}_{V_r} \right).$$

Nezmění se, pokud vyznačený výraz vynásobíme výrazem  $\frac{\ln |\bar{x} - x_i|}{p \ln |\bar{x} - x_{i-1}|} \rightarrow 1$  (limita součtu/součinu je součet/součin limit, pokud všechny výrazy jsou definovány, což dodatečně ověříme).

$$\begin{aligned} \ell(r) &= \lim_{i \rightarrow \infty} \left( \ln |\bar{x} - x_i| - \overbrace{r \ln |\bar{x} - x_{i-1}|}^{V_r} \frac{\ln |\bar{x} - x_i|}{p \ln |\bar{x} - x_{i-1}|} \right) \\ &= \lim_{i \rightarrow \infty} \underbrace{(\ln |\bar{x} - x_i|)}_{\rightarrow -\infty} \underbrace{\left(1 - \frac{r}{p}\right)}_{V_r}. \end{aligned}$$

$$r < p \implies V_r > 0, \ell(r) = -\infty, L(r) = \exp(\ell(r)) = 0,$$

$$r > p \implies V_r < 0, \ell(r) = \infty, L(r) = \exp(\ell(r)) = \infty,$$

$$r = p \implies V_r = V_p = 0, \text{ může existovat konečná limita } \ell(r) \text{ a konečná limita } L(r) = \exp(\ell(r)) > 0.$$

(Zpětně vidíme, že předpoklady o existenci limit součtů a součinů nenarušily platnost výsledků.)  $\square$

Řád metody se obvykle zavádí jako takové  $r$ , pro které limita (10) existuje, je konečná a nenulová. Není to však návod, jak řád vypočítat nebo odhadnout z experimentu. To neumožňuje přímo ani zde použitá definice, protože je v ní použit neznámý kořen  $\bar{x}$ . Bez něj se však lze obejít:

**Věta 4.5** Necht metoda řešení rovnice  $f(x) = 0$  dává za výsledek posloupnost aproximací  $x_i$ ,  $i \in \mathbb{N}$ , konvergující ke kořeni  $\bar{x}$  a

$$L(1) = \lim_{i \rightarrow \infty} \frac{|\bar{x} - x_i|}{|\bar{x} - x_{i-1}|} < 1.$$

Pak řád metody je

$$\lim_{i \rightarrow \infty} \frac{\ln |x_{i+1} - x_i|}{\ln |x_i - x_{i-1}|},$$

pokud limita existuje.

**Důkaz.** Jelikož

$$\left| \frac{\bar{x} - x_i}{\bar{x} - x_{i-1}} \right| \rightarrow L(1) < 1,$$

tak pro nějaké  $\delta > 0$  a všechna dostatečně velká  $i$  je

$$\begin{aligned} \left| \frac{\bar{x} - x_i}{\bar{x} - x_{i-1}} \right| &< 1 - \delta, \\ -(1 - \delta) &< \frac{\bar{x} - x_i}{\bar{x} - x_{i-1}} < 1 - \delta. \end{aligned}$$

To dovoluje omezit výraz

$$\frac{|x_i - x_{i-1}|}{|\bar{x} - x_{i-1}|} = \frac{|(\bar{x} - x_{i-1}) - (\bar{x} - x_i)|}{|\bar{x} - x_{i-1}|} = \left| \frac{\bar{x} - x_{i-1}}{\bar{x} - x_{i-1}} - \frac{\bar{x} - x_i}{\bar{x} - x_{i-1}} \right| = \left| 1 - \underbrace{\frac{\bar{x} - x_i}{\bar{x} - x_{i-1}}}_{\in \langle -(1-\delta), 1-\delta \rangle} \right| \in \langle \delta, 2-\delta \rangle$$

a poměr logaritmů

$$\begin{aligned} \frac{\ln |x_i - x_{i-1}|}{\ln |\bar{x} - x_{i-1}|} &= \frac{\ln \left( |\bar{x} - x_{i-1}| \frac{|x_i - x_{i-1}|}{|\bar{x} - x_{i-1}|} \right)}{\ln |\bar{x} - x_{i-1}|} = \\ &= 1 + \frac{\underbrace{\ln \frac{|x_i - x_{i-1}|}{|\bar{x} - x_{i-1}|}}_{\substack{\in \langle \delta, 2-\delta \rangle \\ \rightarrow 0}}}{\underbrace{\ln |\bar{x} - x_{i-1}|}_{\rightarrow -\infty}} = 1 + \frac{\underbrace{\ln |x_i - x_{i-1}|}_{\in \langle \ln \delta, \ln(2-\delta) \rangle}}{\underbrace{\ln |\bar{x} - x_{i-1}|}_{\rightarrow -\infty}} \rightarrow 1. \end{aligned}$$

Stejně (substitucí  $i := i + 1$ ) dokážeme, že

$$\frac{\ln |x_{i+1} - x_i|}{\ln |\bar{x} - x_i|} \rightarrow 1$$

a dostaneme

$$\begin{aligned} \lim_{i \rightarrow \infty} \frac{\ln |x_{i+1} - x_i|}{\ln |x_i - x_{i-1}|} &= \lim_{i \rightarrow \infty} \frac{\underbrace{\ln |x_{i+1} - x_i|}_{\rightarrow 1}}{\underbrace{\ln |\bar{x} - x_i|}_{\rightarrow 1}} \frac{\ln |\bar{x} - x_{i-1}|}{\ln |\bar{x} - x_{i-1}|} = \\ &= \lim_{i \rightarrow \infty} \frac{\ln |\bar{x} - x_i|}{\ln |\bar{x} - x_{i-1}|} = p. \end{aligned}$$

□

#### 4.7.1 Řád Newtonovy metody

**Věta 4.6** *Nechť funkce  $f$  má nenulovou spojitou druhou derivaci v okolí jednoduchého kořene  $\bar{x}$ . Pokud Newtonova metoda konverguje k  $\bar{x}$ , je řádu 2.*

**Důkaz.**

$$\bar{x} - x_i = -\frac{f''(\xi_i)}{2f'(\theta_i)} (\bar{x} - x_{i-1})^2,$$

kde  $\xi_i \in I(\bar{x}, x_{i-1})$ ,  $\theta_i \in I(x_i, \bar{x})$ . Přejdeme k absolutním hodnotám a zlogaritmuje:

$$\begin{aligned} \ln |\bar{x} - x_i| &= 2 \ln |\bar{x} - x_{i-1}| + \ln \frac{|f''(\xi_i)|}{2|f'(\theta_i)|}, \\ \frac{\ln |\bar{x} - x_i|}{\ln |\bar{x} - x_{i-1}|} &= 2 + \frac{\ln \frac{|f''(\xi_i)|}{2|f'(\theta_i)|}}{\ln |\bar{x} - x_{i-1}|} \rightarrow 2, \end{aligned}$$

neboť čitatel posledního zlomku konverguje ke konstantě  $\ln \frac{|f''(\bar{x})|}{2|f'(\bar{x})|} \in (0, \infty)$  a jmenovatel k  $-\infty$ . □

#### 4.7.2 Řád metody regula falsi

**Věta 4.7** *Metoda regula falsi je 1. řádu, pokud druhá derivace funkce  $f$  nemění na uvažovaném intervalu znaménko.*

**Důkaz.** Taylorův rozvoj se středem  $\bar{x}$  dává v  $x_{i-1}$

$$f(x_{i-1}) = \underbrace{f(\bar{x})}_0 + (x_{i-1} - \bar{x}) f'(\theta_i),$$

kde  $\theta_i \in I(\bar{x}, x_{i-1})$ . Pokud  $x_{i-1}$  není kořen,

$$\frac{f(x_{i-1})}{x_{i-1} - \bar{x}} = f'(\theta_i) \neq 0$$

s limitou

$$\lim_{i \rightarrow \infty} \frac{f(x_{i-1})}{x_{i-1} - \bar{x}} = \lim_{i \rightarrow \infty} f'(\theta_i) = f'(\bar{x}) \neq 0.$$

(Předpoklady věty a metody regula falsi nepřipouštějí násobný kořen.)

BÚNO: Horní mez intervalu se nemění, dolní ano,

$$\forall i \in \mathbb{N} : b_i = b, a_{i+1} = x_i < \bar{x},$$

$$x_i = \frac{a_i f(b) - b f(a_i)}{f(b) - f(a_i)} = \frac{x_{i-1} f(b) - b f(x_{i-1})}{f(b) - f(x_{i-1})},$$

$$\bar{x} - x_i = \frac{(\bar{x} - x_{i-1}) f(b) - (\bar{x} - b) f(x_{i-1})}{f(b) - f(x_{i-1})} =$$

$$= (\bar{x} - x_{i-1}) \left( \frac{f(b)}{f(b) - f(x_{i-1})} - \frac{\bar{x} - b}{f(b) - f(x_{i-1})} \cdot \frac{f(x_{i-1})}{\bar{x} - x_{i-1}} \right),$$

$$\ln(\bar{x} - x_i) = \ln(\bar{x} - x_{i-1}) + \ln \left( \frac{f(b)}{f(b) - f(x_{i-1})} - \frac{\bar{x} - b}{f(b) - f(x_{i-1})} \cdot \frac{f(x_{i-1})}{\bar{x} - x_{i-1}} \right).$$

Jelikož  $\lim_{i \rightarrow \infty} f(x_{i-1}) = f(\bar{x}) = 0$ , limita jmenovatele

$$\lim_{i \rightarrow \infty} (f(b) - f(x_{i-1})) = f(b) \neq 0,$$

$$\begin{aligned} \lim_{i \rightarrow \infty} \ln \left( \frac{f(b)}{f(b) - f(x_{i-1})} - \frac{\bar{x} - b}{f(b) - f(x_{i-1})} \cdot \frac{f(x_{i-1})}{\bar{x} - x_{i-1}} \right) &= \\ = \ln \left( 1 - \frac{\bar{x} - b}{f(b)} \cdot f'(\bar{x}) \right), \end{aligned}$$

což je konstanta, která po vydělení  $\ln(\bar{x} - x_{i-1})$  dává limitu 0, takže

$$\lim_{i \rightarrow \infty} \frac{\ln(\bar{x} - x_i)}{\ln(\bar{x} - x_{i-1})} = 1.$$

□

metoda	řád	podmínka
bisekce	neřad. ( $\sim 1$ )	
regula falsi	1	druhá derivace nemění znaménko
sečen	$(1 + \sqrt{5})/2 \doteq 1.6$	jednoduchý kořen
Newtonova	2	jednoduchý kořen

Metoda bisekce nekonverguje monotónně, takže nemá řád, ale kdybychom místo skutečných chyb použili ve vzorci jejich horní odhady, dostali bychom

$$\lim_{i \rightarrow \infty} \frac{\ln(b_i - a_i)}{\ln(b_{i-1} - a_{i-1})} = \lim_{i \rightarrow \infty} \frac{\ln(b_{i-1} - a_{i-1}) - \ln 2}{\ln(b_{i-1} - a_{i-1})} = 1,$$

což je podobná situace jako u metod 1. řádu.

## 4.8 Kombinace startovacích a zpřesňujících metod

Kombinace dvou metod – **startovací** a **zpräšňující**.

Výpočet zahájíme startovací metodou, od níž se požaduje zaručená konvergence, byť třeba pomalá (např. metoda bisekce nebo regula falsi). Ta vlastně jen vylepší separaci kořene.

Poté zkusíme uplatnit zpřesňující metodu, která by měla rychleji konvergovat a urychlit tak zpřesnění nalezeného odhadu (např. Newtonova metoda). Její konvergence nebývá zaručena, ale můžeme se vrátit ke startovací metodě.

## 4.9 Metoda prosté iterace (MPI)

Rovnici  $f(x) = 0$  převedeme na ekvivalentní tvar  $\varphi(x) = x$ , např.  $\varphi(x) = f(x) + x$ .

Počáteční odhad  $x_0$ ,  $x_i = \varphi(x_{i-1})$ .

**Podmínka ukončení:**  $|x_i - x_{i-1}| < \eta$ .

**Tvrzení 4.1** Pokud MPI konverguje k  $\tilde{x}$  a  $\varphi$  je v  $\tilde{x}$  spojitá, pak  $\varphi(\tilde{x}) = \tilde{x}$ ,  $f(\tilde{x}) = 0$ .

**Důkaz.**

$$\varphi(\tilde{x}) = \varphi\left(\lim_{i \rightarrow \infty} x_i\right) = \lim_{i \rightarrow \infty} \varphi(x_i) = \lim_{i \rightarrow \infty} x_{i+1} = \lim_{i \rightarrow \infty} x_i = \tilde{x}.$$

□

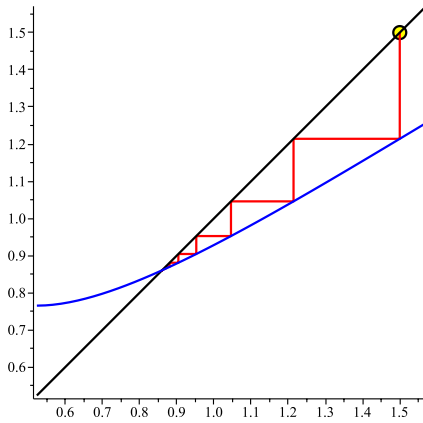
### Příklad použití MPI

**Příklad 4.1** Hledáme nejmenší kladné řešení rovnice  $f(x) = 0$ , kde  $f(x) = x - \cotg x$ .

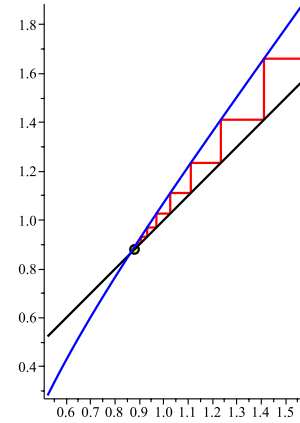
$$f\left(\frac{\pi}{4}\right) = \frac{\pi}{4} - 1 < 0, \quad f\left(\frac{\pi}{2}\right) = \frac{\pi}{2} > 0 \implies \bar{x} \in \left(\frac{\pi}{4}, \frac{\pi}{2}\right)$$

Zvolíme  $\varphi(x) = \lambda f(x) + x$ , kde  $\lambda \neq 0$ ; podmínka ukončení pro  $\eta = 0.001$ .

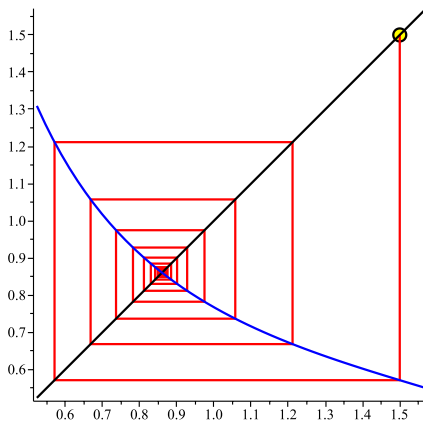
Vyzkoušíme  $\lambda \in \{-0.2, 0.2, -0.65, -0.8\}$ .



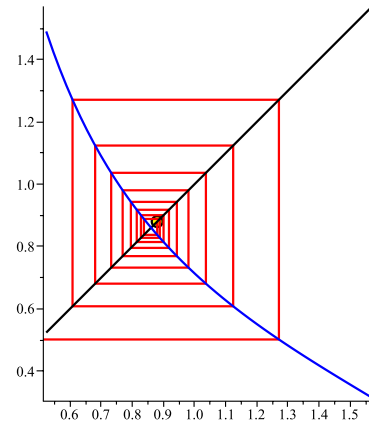
$$\begin{aligned} x_{i+1} &= 0.8x_i + 0.2 \cotg x_i, \\ x_0 &= 1.5, \\ &\text{konverguje monotónně} \end{aligned}$$



$$\begin{aligned} x_{i+1} &= 1.2x_i - 0.2 \cotg x_i, \\ x_0 &= 0.88 \\ &\text{diverguje monotónně} \end{aligned}$$



$$\begin{aligned} x_{i+1} &= 0.35x_i + 0.65 \cotg x_i, \\ x_0 &= 1.5, \\ &\text{konverguje nemonotónně} \end{aligned}$$



$$\begin{aligned} x_{i+1} &= 0.2x_i + 0.8 \cotg x_i, \\ x_0 &= 0.88, \\ &\text{diverguje nemonotónně} \end{aligned}$$



### 4.9.1 Kontraktivní funkce

**Definice 4.1** Řekneme, že funkce  $\varphi$  je na intervalu  $I$  **kontraktivní** (s koeficientem  $q$ ), jestliže

$$\exists q < 1 \forall u, v \in I : |\varphi(u) - \varphi(v)| \leq q \cdot |u - v|.$$

kontraktivita  $\implies$  spojitost

**Věta 4.8 (Postačující podmínka pro kontraktivitu)** Necht funkce  $\varphi$  má na intervalu  $I$  spojitou derivaci a existuje  $q < 1$  takové, že

$$\forall x \in I : |\varphi'(x)| \leq q.$$

Pak  $\varphi$  je na  $I$  kontraktivní s koeficientem  $q$ .

**Důkaz.**  $|\varphi(u) - \varphi(v)| = \left| \int_v^u \varphi'(x) dx \right| \leq \int_v^u |\varphi'(x)| dx \leq \int_v^u q dx = q \cdot |u - v|.$   $\square$

### 4.9.2 Věta o pevném bodě

**Věta 4.9 (Banachova věta o pevném bodě pro reálné funkce)** Necht  $\varphi$  je funkce kontraktivní s koeficientem  $q < 1$  na nějakém uzavřeném intervalu  $I = \langle a, b \rangle$  taková, že zobrazuje  $I$  do  $I$ . Pak rovnice  $\varphi(x) = x$  má v intervalu  $I$  právě jedno řešení  $\bar{x}$ . To dostaneme MPI s libovolnou počáteční hodnotou  $x_0 \in I$ . Odhad chyby:

$$|\bar{x} - x_i| \leq \frac{q}{1 - q} |x_i - x_{i-1}|.$$

**Důkaz.**

- Existence řešení:

$\varphi$  zobrazuje  $I$  do  $I$

$\psi(x) = \varphi(x) - x$  je  $\psi$  v  $a$  nezáporná a v  $b$  nekladná; je spojitá, a tedy má v  $I$  nulový bod; ten je řešením rovnice  $\varphi(x) = x$ .

- Jednoznačnost řešení: Předpokládejme další řešení  $\bar{\bar{x}} \in I$ . Pak

$$|\bar{x} - \bar{\bar{x}}| = |\varphi(\bar{x}) - \varphi(\bar{\bar{x}})| \leq q \cdot |\bar{x} - \bar{\bar{x}}| \implies \bar{\bar{x}} = \bar{x}$$

- Konvergence MPI k řešení:

$$|\bar{x} - x_i| = |\varphi(\bar{x}) - \varphi(x_{i-1})| \leq q \cdot |\bar{x} - x_{i-1}| \leq \dots \leq q^i \cdot |\bar{x} - x_0| \rightarrow 0.$$

- Odhad chyby:

$$\begin{aligned} |\bar{x} - x_i| &\leq q \cdot |\bar{x} - x_{i-1}| = q \cdot |(\bar{x} - x_i) + (x_i - x_{i-1})| \\ &\leq q \cdot |\bar{x} - x_i| + q \cdot |x_i - x_{i-1}|, \end{aligned}$$

$$|\bar{x} - x_i| \leq \frac{q}{1 - q} |x_i - x_{i-1}|.$$

$\square$

Koeficient kontrakce  $q$  je horním odhadem

$$L(1) = \lim_{i \rightarrow \infty} \frac{|\bar{x} - x_i|}{|\bar{x} - x_{i-1}|} < 1,$$

což je kritérium konvergence použité u předchozích metod.

### 4.9.3 Optimalizace MPI

Jak rovnici  $f(x) = 0$  převést na ekvivalentní tvar  $\varphi(x) = x$  takový, že MPI rychle konverguje?  
Možné řešení:

$$\varphi(x) = x + \lambda f(x),$$

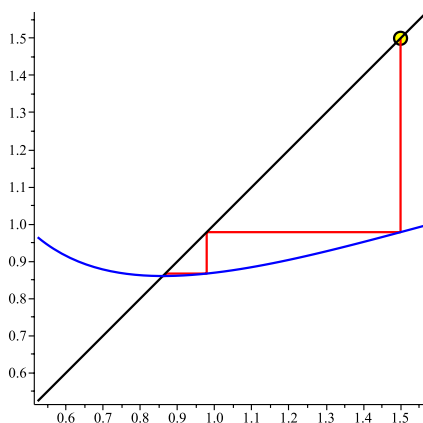
kde  $\lambda \neq 0$  a

$$\varphi'(x) = 1 + \lambda f'(x)$$

je malá.

**Příklad 4.2 (pokračování Příkladu 4.1)**  $f'(x) = 2 + \cotg^2 x \in \langle 2, 3 \rangle \implies \lambda \in \langle -\frac{1}{2}, -\frac{1}{3} \rangle$ ,  
 $-1/f'(0.86) \approx -0.365 \implies \lambda = -0.365$ ,

$$\varphi(x) = 0.635x + 0.365 \cotg x.$$



$$x_{i+1} = 0.635x_i + 0.365 \cotg x_i, \quad x_0 = 1.5$$

konverguje monotónně a rychle

### 4.9.4 Řád metody prosté iterace

**Věta 4.10** *Nechť MPI konverguje k  $\bar{x}$ . Nechť  $p$  je nejmenší přirozené číslo, pro které  $\varphi^{(p)}(\bar{x}) \neq 0$ , a  $\varphi^{(p)}$  je spojitá v nějakém okolí bodu  $\bar{x}$ . Pak řád metody je  $p$ .*

**Důkaz.** Taylorův rozvoj funkce  $\varphi$  se středem  $\bar{x}$  vyhodnotíme v  $x_{i-1}$ :

$$\begin{aligned} \underbrace{\varphi(x_{i-1})}_{x_i} &= \underbrace{\varphi(\bar{x})}_{\bar{x}} + \frac{1}{p!} (x_{i-1} - \bar{x})^p \varphi^{(p)}(\xi_{i-1}), \quad \text{kde } \xi_{i-1} \in I(\bar{x}, x_{i-1}), \\ \frac{\ln |\bar{x} - x_i|}{\ln |\bar{x} - x_{i-1}|} &= \frac{\ln \left| \frac{1}{p!} (x_{i-1} - \bar{x})^p \varphi^{(p)}(\xi_{i-1}) \right|}{\ln |\bar{x} - x_{i-1}|} = \\ &= \frac{p \ln |\bar{x} - x_{i-1}|}{\ln |\bar{x} - x_{i-1}|} - \underbrace{\frac{\ln p!}{\ln |\bar{x} - x_{i-1}|}}_{\rightarrow 0} + \underbrace{\frac{\ln |\varphi^{(p)}(\xi_{i-1})|}{\ln |\bar{x} - x_{i-1}|}}_{\rightarrow \varphi^{(p)}(\bar{x}) \neq 0} \rightarrow p. \end{aligned}$$

□

**Poznámka 4.2** *Nejčastěji je  $\varphi'(\bar{x}) \neq 0$ , takže MPI je řádu 1. Nemusí to však být vždy, např. Newtonova metoda je speciálním případem MPI.*

### Zrychlení konvergence MPI

**Nápad:** V každém kroku zvolíme jiný koeficient  $\lambda_i$  tak, aby  $\varphi'(x_i) = 0$ , tj.

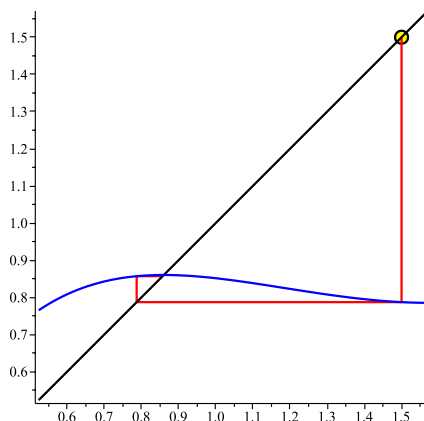
$$\lambda_i = -\frac{1}{f'(x_i)},$$

Dostaneme

$$x_{i+1} = x_i + \lambda_i f(x_i) = x_i - \frac{f(x_i)}{f'(x_i)},$$

což je Newtonova metoda (jako speciální případ MPI); ta je (obvykle) řádu 2, zatímco MPI (obvykle) řádu 1.

## Newtonova metoda jako speciální případ MPI



$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}, \quad x_0 = 1.5$$

konverguje nemonotónně a rychle

### 4.9.5 Kritéria pro výběr metody řešení rovnic

- **jednobodové**, např. MPI (která je v jistém smyslu univerzální jednobodovou metodou), Newtonova metoda,
- **dvoubodové**, např. bisekce, regula falsi, metoda sečen,
- **vícebodové**.

Z programátorského hlediska:

- **nevyžadující derivaci**, např. bisekce, regula falsi, metoda sečen a *obvykle* MPI (záleží na zvoleném iteračním vzorci),
- **vyžadující znalost první derivace**, např. Newtonova,
- **vyžadující znalost vyšších derivací**.

Podle konvergence dělíme metody řešení rovnic na

- **vždy konvergentní**, např. bisekce a regula falsi,
- **ostatní**, např. Newtonova, metoda sečen, MPI.

## 4.10 Podobné úlohy

### 4.10.1 Hledání násobných kořenů

- V okolí kořene *sudé násobnosti* funkce nemění znaménko, takže nelze použít metody bisekce a regula falsi.
- Metoda sečen a Newtonova metoda jsou sice použitelné pro hledání násobných kořenů, ale jejich konvergence je pak prvního řádu.
- V metodě prosté iterace záleží pouze na použitém iteračním vzorci, nikoli na násobnosti kořene původní rovnice.

**1. metoda:** Najdeme (všechny) kořeny funkce  $f'$  a vyzkoušíme, zda některý z nich je kořenem funkce  $f$ . Tam, kde má  $f$  kořen sudé násobnosti, má  $f'$  kořen liché násobnosti a mění znaménko.

### Hledání násobných kořenů

**2. metoda:** Uvažujme funkci  $h(x) = \frac{f(x)}{f'(x)}$  (kde „odstraníme odstranitelné nespojitosti“).

**Tvrzení 4.2** *Nechť  $\bar{x}$  je  $k$ -násobný kořen funkce  $f$ , v jehož okolí má  $f$  spojitou derivaci řádu  $k$ . Pak  $\bar{x}$  je jednoduchým kořenem funkce  $h = f/f'$ .*

**Důkaz.** Definice  $k$ -násobného kořene říká, že  $f^{(j)}(\bar{x}) = 0$  pro  $j < k$  a  $f^{(k)}(\bar{x}) \neq 0$ . Opakovaným užitím l'Hospitalova pravidla odvodíme nenulovou limitu

$$\lim_{x \rightarrow \bar{x}} \frac{f(x)}{(x - \bar{x})^k} = \lim_{x \rightarrow \bar{x}} \frac{f'(x)}{k(x - \bar{x})^{k-1}} = \dots = \lim_{x \rightarrow \bar{x}} \frac{f^{(k)}(x)}{k!} \neq 0.$$

Tedy podíl prvních dvou výrazů je definován a je jednotkový,

$$\lim_{x \rightarrow \bar{x}} \frac{f(x)}{(x - \bar{x})^k} \cdot \frac{k(x - \bar{x})^{k-1}}{f'(x)} = 1,$$

tím dostáváme pro funkci  $h$  limitu

$$\lim_{x \rightarrow \bar{x}} \frac{h(x)}{x - \bar{x}} = \frac{1}{k} \neq 0.$$

V poslední limitě konverguje jmenovatel k nule, musí k ní tedy konvergovat i čitatel, takže  $\lim_{x \rightarrow \bar{x}} h(x) = 0$  a  $\bar{x}$  je kořenem funkce  $h$ . Nenulová je podle l'Hospitalova pravidla též  $\lim_{x \rightarrow \bar{x}} h'(x)$ , takže  $\bar{x}$  je jednoduchý kořen funkce  $h$ .  $\square$

Pokud funkce  $f$  má pouze kořeny konečné násobnosti, pak funkce  $h$  má tytéž kořeny, ale jednoduché (není však spojitá).

#### 4.10.2 Řešení algebraických rovnic neboli hledání kořenů polynomů

Speciální případ rovnice  $f(x) = 0$ , kde  $f$  je polynom.

**Věta 4.11 (Odhad polohy kořenů polynomu)** *Všechny (komplexní) kořeny rovnice*

$$\sum_{i=0}^n a_i x^i = 0$$

*mají absolutní hodnotu nejvýše*

$$1 + \frac{\max(|a_0|, \dots, |a_{n-1}|)}{|a_n|}.$$

#### 4.10.3 Řešení rovnic v komplexním oboru

Metoda bisekce a metoda regula falsi jsou závislé na úplném uspořádání reálných čísel  $\implies$  ve větší dimenzi nepoužitelné.

Metoda sečen a Newtonova metoda jsou použitelné pro komplexní kořeny.

Pro nalezení komplexních kořenů může být nutný počáteční odhad s nenulovou imaginární částí.

#### 4.10.4 Řešení soustav rovnic

Newtonova metoda má i zobecnění pro soustavy nelineárních rovnic; pak místo derivace pracujeme s jacobíánem a místo dělení jej potřebujeme invertovat, čímž se jednak zvyšuje složitost výpočtu, jednak vznikají problémy s body, v nichž je jacobíán singulární. Podmínky konvergence jsou opět složitější než v reálném případě.

Metoda prosté iterace je použitelná i v prostorech větší (konečné) dimenze. Zajištění kontraktivity použitého zobrazení může být problém.

Kvůli obtížím se zajištěním konvergence se pro řešení soustav rovnic často používají metody založené na jiných principech než v jednodimenzionálním případě.

## Dodatek: Přehled značení

Popis je zjednodušený a nemusí být přesný, podrobnosti jsou v textu. Značení použité jen lokálně zde není uvedeno.

### Značení specifické pro tuto kapitolu

$f$  ... funkce, jejíž kořeny hledáme; řešíme rovnici  $f(x) = 0$

$\langle a_0, b_0 \rangle$  ... počáteční interval, ve kterém hledáme kořeny (výsledek separace kořenů)

$\bar{x}$  ... přesné řešení;  $f(\bar{x}) = 0$

$x_0, x_1, x_2, \dots$  ... posloupnost odhadů řešení

$\varepsilon$  ... požadovaná přesnost řešení (v argumentu,  $|x_i - \bar{x}| \leq \varepsilon$ )

$\delta$  ... požadovaná přesnost funkční hodnoty ( $|f(x_i)| \leq \delta$ )

$\eta$  ... hodnota pro podmínku ukončení podle rozdílu posledních odhadů ( $|x_i - x_{i-1}| \leq \eta$ )

$q$  ... (převážně) koeficient kontrakce u kontraktivního zobrazení

$r$  ... řád metody

$\varphi$  ... iterační funkce v metodě prosté iterace s předpisem  $x_i = \varphi(x_{i-1})$

### Značení používané podobně v celém předmětu

$I(\dots)$  ... nejmenší interval obsahující čísla (body) v závorce, např.  $I(x_0, \dots, x_{n-1}) = [\min_i x_i, \max_i x_i]$

$M_j$  ... horní odhad absolutní hodnoty  $j$ -té derivace funkce,  $|f^{(j)}| \leq M_j$  na použitém intervalu

$m_j$  ... dolní odhad absolutní hodnoty  $j$ -té derivace funkce,  $|f^{(j)}| \geq m_j$  na použitém intervalu

## 5 NUMERICKÉ ŘEŠENÍ SOUSTAV LINEÁRNÍCH ROVNIC

### 5.1 Formulace úlohy a její obtíž

**Úloha:** Hledáme řešení soustavy  $n$  lineárních rovnic o  $n$  neznámých  $x_1, x_2, \dots, x_n$

$$\begin{aligned} a_{1,1} x_1 + a_{1,2} x_2 + \dots + a_{1,n} x_n &= b_1 \\ a_{2,1} x_1 + a_{2,2} x_2 + \dots + a_{2,n} x_n &= b_2 \\ &\vdots \\ a_{n,1} x_1 + a_{n,2} x_2 + \dots + a_{n,n} x_n &= b_n \end{aligned}$$

Maticový tvar:

$$\mathbf{A} \vec{x} = \vec{b},$$

kde  $\mathbf{A} = (a_{i,j})_{i,j=1,\dots,n}$  je (regulární) matice soustavy,

$\vec{b} = (b_1, b_2, \dots, b_n)^\top$  vektor pravých stran,

$\vec{x} = (x_1, x_2, \dots, x_n)^\top$  vektor neznámých.

**Cramerovo pravidlo** má velkou výpočetní složitost a numerické chyby.

#### 5.1.1 Druhy problémů

Matice soustavy:

- plné, ne příliš velké,
- řídké, často velmi velké (mj. u kubického splinu).

#### 5.1.2 Špatná podmíněnost

$$\vec{x} = \mathbf{A}^{-1} \vec{b}$$

Malá změna koeficientů soustavy nebo pravé strany může způsobit velkou změnu řešení.

Zpětné dosazení (nepřesného) řešení  $\vec{x}_c$  dá **reziduum řešení**:

$$\vec{r} = \vec{b} - \mathbf{A} \vec{x}_c,$$

Pokud matice  $\mathbf{A}^{-1}$  má velké prvky, může být reziduum  $\vec{r}$  malé, i když se vektor  $\vec{x}_c$  podstatně liší od přesného řešení  $\vec{x}$ .

$$\begin{aligned} \vec{r} &= \mathbf{A} \vec{x} - \mathbf{A} \vec{x}_c = \mathbf{A} (\vec{x} - \vec{x}_c), \\ \vec{x} - \vec{x}_c &= \mathbf{A}^{-1} \vec{r}. \end{aligned}$$

Jsou-li prvky matice  $\mathbf{A}^{-1}$  velké, může i malá složka vektoru  $\vec{r}$  způsobit velký rozdíl  $\vec{x} - \vec{x}_c$ .

Malé reziduum nezaručuje malou chybu řešení!

Takové soustavy nazýváme **špatně podmíněné**.

#### Příklad 5.1 *Soustava*

$$\begin{aligned} 2x + 6y &= 8 \\ 2x + 6.00001y &= 8.00001 \end{aligned}$$

má řešení  $x = 1, y = 1$ ;

minimální změna koeficientů na soustavu

$$\begin{aligned} 2x + 6y &= 8 \\ 2x + 5.99999y &= 8.00002 \end{aligned}$$

změní řešení na  $x = 10, y = -2$ .

Inverzní matice k oběma soustavám mají prvky řádově  $10^5$ , což ukazuje na jejich špatnou podmíněnost.

Rovnice v soustavách jsou „skoro lineárně závislé“.

### 5.1.3 Zdroje chyb

- nepřesnost koeficientů soustavy a pravé strany,
- zaokrouhlovací chyby při výpočtu,
- chyby metody – nekonečný proces je nahrazen konečným počtem kroků (u iteračních metod).

## 5.2 Přímé metody

Po konečném počtu kroků vedou (teoreticky) k přesnému řešení.

### 5.2.1 Gaussova eliminace (GEM)

Postupné úpravy matice soustavy pomocí ekvivalentních úprav (nemění řešení soustavy) na horní trojúhelníkovou matici, ze které lze zpětným dosazením snadno získat řešení.

**Rozšířená matice soustavy** má prvky

$$\begin{aligned}a_{i,j}^{(0)} &= a_{i,j}, & \text{pro } i = 1, 2, \dots, n, j = 1, 2, \dots, n; \\a_{i,n+1}^{(0)} &= b_i, & \text{pro } i = 1, 2, \dots, n.\end{aligned}$$

Soustavu

$$\begin{aligned}a_{1,1}^{(0)} x_1 + a_{1,2}^{(0)} x_2 + \dots + a_{1,n}^{(0)} x_n &= a_{1,n+1}^{(0)} \\a_{2,1}^{(0)} x_1 + a_{2,2}^{(0)} x_2 + \dots + a_{2,n}^{(0)} x_n &= a_{2,n+1}^{(0)} \\&\vdots \\a_{n,1}^{(0)} x_1 + a_{n,2}^{(0)} x_2 + \dots + a_{n,n}^{(0)} x_n &= a_{n,n+1}^{(0)}\end{aligned}$$

převědeme povolenými úpravami na tvar

$$\begin{aligned}a_{1,1}^{(0)} x_1 + a_{1,2}^{(0)} x_2 + \dots + a_{1,n}^{(0)} x_n &= a_{1,n+1}^{(0)} \\a_{2,2}^{(1)} x_2 + \dots + a_{2,n}^{(1)} x_n &= a_{2,n+1}^{(1)} \\&\vdots \\a_{n,n}^{(n-1)} x_n &= a_{n,n+1}^{(n-1)},\end{aligned}$$

ze kterého zpětnou substitucí vypočítáme vektor řešení.

Pokud vyjde na diagonále nulový prvek, stačí provést záměnu řádků (resp. sloupců – v tom případě musíme **zaměnit i odpovídající složky vektoru řešení!**).

To lze, pokud je matice soustavy regulární.

**Algoritmus 5.1** Pro  $k = 1, 2, \dots, n - 1$ , pro  $i = k + 1, k + 2, \dots, n$ ,  $j = k + 1, k + 2, \dots, n + 1$

$$a_{i,j}^{(k)} = a_{i,j}^{(k-1)} - \frac{a_{i,k}^{(k-1)}}{a_{k,k}^{(k-1)}} a_{k,j}^{(k-1)}.$$

Pokud po provedení **přímého chodu** je nějaký diagonální prvek  $a_{i,i}^{(i-1)} = 0$  (resp.  $|a_{i,i}^{(i-1)}| < \varepsilon$ ), matice soustavy je (resp. může být) singulární.

V opačném případě použijeme **zpětnou substituci**

$$x_i = \frac{1}{a_{i,i}^{(i-1)}} \left( a_{i,n+1}^{(i-1)} - \sum_{j=i+1}^n a_{i,j}^{(i-1)} x_j \right), \quad \text{pro } i = n, n - 1, \dots, 1.$$

### 5.2.2 Výběr hlavního prvku

Pokud číslo na diagonále je v absolutní hodnotě malé, jeho malá změna vyvolá velkou změnu výsledku při dělení a rostou zaokrouhlovací chyby.

Proto v každém kroku eliminace vybereme na diagonálu koeficient s co největší absolutní hodnotou = **hlavní prvek (pivot)**.

GEM s výběrem hlavního prvku

- **úplným** – vybíráme z  $(n - k)^2$  prvků zbylé čtvercové podmatice (výpočetně složité),
- **sloupcovým** – vybíráme v rámci sloupce a pouze vyměníme řádky,
- **řádkový** – vybíráme v rámci řádku a vyměníme sloupce (**i pořadí neznámých!**).



### 5.2.3 Gaussova-Jordanova redukce

GEM může pokračovat eliminací prvků nad diagonálou.

Diagonální prvky lze převést na jedničky.

Sloupec pravých stran je pak vektor řešení.

Pro jedno použití je větší složitost, ale vyplatí se, pokud máme mnoho úloh lišících se pouze pravou stranou (např. výpočet inverzní matice, kdy řešíme soustavu lineárních rovnic současně pro  $n$  pravých stran, kde vycházíme z jednotkové matice).

### 5.2.4 LU-rozklad

Označme

$$\mathbf{L}_1 = \begin{pmatrix} 1 & 0 & \dots & 0 \\ -\frac{a_{2,1}}{a_{1,1}} & 1 & 0 & \dots & 0 \\ -\frac{a_{3,1}}{a_{1,1}} & 0 & 1 & & \\ \vdots & \vdots & & \ddots & \\ -\frac{a_{n,1}}{a_{1,1}} & 0 & & & 1 & 0 \\ \frac{a_{n,1}}{a_{1,1}} & 0 & & \dots & 0 & 1 \end{pmatrix}$$

a vynásobme  $\mathbf{L}_1 \cdot \mathbf{A}$ . Dostaneme první přidruženou soustavu z GEM s vynulovaným prvním sloupcem pod diagonálou. Pokračujeme:

$$\mathbf{A}_0 = \mathbf{A}, \quad \mathbf{A}_{i+1} = \mathbf{L}_{i+1} \cdot \mathbf{A}_i \quad \text{pro } i = 0, 1, \dots, n-2,$$

kde

$$\mathbf{L}_{i+1} = \begin{pmatrix} 1 & 0 & & \dots & & 0 \\ 0 & 1 & & & & 0 \\ & & 0 & & & \vdots \\ & & -\frac{a_{i+2,i+1}^{(i)}}{a_{i+1,i+1}^{(i)}} & & & \\ & & & \ddots & & \\ \vdots & \vdots & \vdots & & & 1 & 0 \\ 0 & 0 & -\frac{a_{n,i+1}^{(i)}}{a_{i+1,i+1}^{(i)}} & \dots & 0 & 1 \end{pmatrix}.$$

Po provedení  $n-1$  maticových násobení máme

$$\mathbf{L}_{n-1} \cdot \mathbf{L}_{n-2} \dots \mathbf{L}_2 \cdot \mathbf{L}_1 \cdot \mathbf{A} = \mathbf{U},$$

kde matice  $\mathbf{U}$  je horní trojúhelníková (=výsledek přímého chodu GEM) a  $\bar{\mathbf{L}} = \mathbf{L}_{n-1} \cdot \mathbf{L}_{n-2} \dots \mathbf{L}_2 \cdot \mathbf{L}_1$  dolní trojúhelníková s jednotkami na diagonále. Inverzní matice  $\bar{\mathbf{L}}^{-1} = \mathbf{L}$  existuje a je rovněž dolní trojúhelníková s jednotkami na diagonále.

$$\begin{aligned} \bar{\mathbf{L}} \cdot \mathbf{A} &= \mathbf{U}, \\ \mathbf{A} &= \bar{\mathbf{L}}^{-1} \cdot \mathbf{U} = \mathbf{L} \cdot \mathbf{U}. \end{aligned}$$

Původní soustavu  $\mathbf{A} \vec{x} = \vec{b}$  nahradíme dvěma soustavami s trojúhelníkovými maticemi

$$\begin{aligned} \mathbf{L} \vec{y} &= \vec{b}, \\ \mathbf{U} \vec{x} &= \vec{y}, \end{aligned}$$

(neboť  $\mathbf{A} \vec{x} = \mathbf{L} \cdot \mathbf{U} \vec{x} = \mathbf{L} \vec{y} = \vec{b}$ ),

které řešíme zpětnou substitucí.

Rozepsáním součinu  $\mathbf{L} \cdot \mathbf{U}$  dostáváme

**Algoritmus 5.2** Pro  $r = 1, 2, \dots, n$

$$\begin{aligned}
 u_{i,r} &= a_{i,r} - \sum_{s=1}^{i-1} l_{i,s} a_{s,r} & \text{pro } i = 1, 2, \dots, r, \\
 l_{i,r} &= \frac{1}{u_{r,r}} \left( a_{i,r} - \sum_{s=1}^{r-1} l_{i,s} u_{s,r} \right) & \text{pro } i = r+1, r+2, \dots, n, \\
 y_i &= b_i - \sum_{s=1}^{i-1} l_{i,s} y_s & \text{pro } i = 1, 2, \dots, n, \\
 x_i &= \frac{1}{u_{i,i}} \left( y_i - \sum_{s=i+1}^n u_{i,s} x_s \right) & \text{pro } i = n, \dots, 2, 1.
 \end{aligned}$$

Potřebujeme všechny prvky  $u_{r,r} \neq 0$  (i během výpočtu) nenulové  $\implies$  výběr hlavního prvku (částečný); zmenší se tím i zaokrouhlovací chyby.

**Poznámka:** Celý algoritmus můžeme realizovat „na místě“, v jediné čtvercové matici. Pro překrývající se diagonální prvky použijeme  $u_{r,r}$ , protože na diagonále matice  $\mathbf{L}$  jsou jedničky, které nepočítáme ani neukládáme.

**Poznámka:** Tato metoda je zvláště vhodná pro řadu úloh lišících se pouze pravými stranami.

Lze použít i výpočet inverzní matice  $\mathbf{A}^{-1}$  (ovšem s větší složitostí).

### 5.2.5 Výpočet inverzní matice

$\mathbf{E}$  = jednotková matice

$\mathbf{A} \cdot \mathbf{A}^{-1} = \mathbf{E} \implies \mathbf{A}$  vynásobená  $j$ -tým sloupcem matice  $\mathbf{A}^{-1}$  je rovna  $j$ -tému sloupci jednotkové matice  $\mathbf{E}$ .

$\vec{x}_j = j$ -tý sloupec  $\mathbf{A}^{-1}$

$\vec{e}_j = j$ -tý sloupec  $\mathbf{E}$

$$\mathbf{A} \vec{x}_j = \vec{e}_j$$

Máme soustavu rovnic, kde  $\vec{x}_j$  je vektor neznámých.

Jednotlivé sloupce inverzní matice  $\mathbf{A}^{-1}$  dostaneme jako řešení soustavy pro různé pravé strany – sloupce matice  $\mathbf{E}$ .

Můžeme využít GEM pro jednu matici soustavy a několik pravých stran současně; stačí „prodloužit“ cyklus pro řádky rozšířené matice soustavy typu  $(n \times 2n)$ , tedy  $j = k+1, k+2, \dots, 2n$  (viz Gaussova-Jordanova redukce).

Použití LU-rozkladu:  $\mathbf{A} = \mathbf{L} \cdot \mathbf{U}$ :

$$\mathbf{A}^{-1} = (\mathbf{L} \cdot \mathbf{U})^{-1} = \mathbf{U}^{-1} \cdot \mathbf{L}^{-1}.$$

Výpočet  $\mathbf{U}^{-1}$  a  $\mathbf{L}^{-1}$  je snadný, inverzní matice k trojúhelníkové je opět trojúhelníková.

### 5.2.6 Výpočet determinantu

Podle definice pouze pro velmi malé řádky matic.

$n$	2	3	4	5	10	20	30
počet operací	4	18	96	600	36 288 000	$4.8 \cdot 10^{19}$	$7.9 \cdot 10^{33}$

GEM: po eliminaci jako součin prvků na diagonále:

$$\det \mathbf{A} = \pm a_{1,1}^{(0)} a_{2,2}^{(1)} a_{3,3}^{(2)} \dots a_{n,n}^{(n-1)}.$$

**POZOR!** Výměna řádků či sloupců (při výběru hlavního prvku) **mění znaménko** determinantu. (Stačí si pamatovat, zda počet výměn byl sudý nebo lichý.)

LU-rozklad  $\mathbf{A} = \mathbf{L} \cdot \mathbf{U}$ :

$$\det \mathbf{A} = \det (\mathbf{L} \cdot \mathbf{U}) = \det \mathbf{L} \cdot \det \mathbf{U} = u_{1,1} u_{2,2} u_{3,3} \dots u_{n,n}.$$

(Jedná se o trojúhelníkové matice a  $\mathbf{L}$  má navíc na diagonále jednotky.)

Opět **nutno ošetřit znaménko** při výměnách řádků nebo sloupců.

### 5.2.7 Zpřesnění výsledků pomocí rezidua

Přesné řešení  $\vec{x}$  můžeme vyjádřit pomocí nepřesného řešení  $\vec{x}_c$ :

$$\vec{x} = \vec{x}_c + \vec{\delta},$$

kde  $\vec{\delta} = (\delta_1, \delta_2, \dots, \delta_n)^T$  je **vektor oprav**.

$$\begin{aligned} \mathbf{A} \vec{x} &= \vec{b} \\ \mathbf{A} (\vec{x}_c + \vec{\delta}) &= \vec{b} \\ \mathbf{A} \vec{x}_c + \mathbf{A} \vec{\delta} &= \vec{b} \\ \mathbf{A} \vec{\delta} &= \vec{b} - \mathbf{A} \vec{x}_c = \vec{r} \end{aligned}$$

Vektor oprav je řešením stejné soustavy s pravou stranou nahrazenou reziduem  $\vec{r}$ .

Je-li reziduum podobně velké jako původní pravé strany, nemá smysl pokračovat.

Postup můžeme několikrát opakovat (je výhodné použít LU-rozklad) a získat další zpřesněná řešení

$$\vec{x}_c^{(1)}, \vec{x}_c^{(2)}, \vec{x}_c^{(3)}, \dots$$

## 5.3 Iterační metody

Snaží se konstruovat posloupnosti vektorů, konvergující k přesnému řešení soustavy.

### 5.3.1 Normy vektorů a matic

$\mathbb{R}^n$  ...  $n$ -rozměrný aritmetický vektorový prostor

$\mathbb{R}^{n,n}$  ... prostor čtvercových matic řádu  $n$

$\vec{o} \in \mathbb{R}^n$  ... nulový vektor

$\mathbf{O} \in \mathbb{R}^{n,n}$  ... nulová matice

$\mathbf{E} \in \mathbb{R}^{n,n}$  ... jednotková matice

**Definice 5.1** (Vektorová) **norma** je zobrazení  $\|\cdot\|_v: \mathbb{R}^n \rightarrow \mathbb{R}$  splňující

- $\|\vec{x}\|_v \geq 0$ , přičemž  $\|\vec{x}\|_v = 0 \Leftrightarrow \vec{x} = \vec{o}$ ,
- $\|c\vec{x}\|_v = |c| \|\vec{x}\|_v$ ,
- $\|\vec{x} + \vec{y}\|_v \leq \|\vec{x}\|_v + \|\vec{y}\|_v$  (trojúhelníková nerovnost).

„Vektorových norem je hodně:“

#### Příklad 5.2

- $\|\vec{x}\|_e = \sqrt{\sum_{i=1}^n x_i^2}$  *euklidovská*
- $\|\vec{x}\|_r = \max_{i=1, \dots, n} |x_i|$  *maximová, Čebyševova*
- $\|\vec{x}\|_s = \sum_{i=1}^n |x_i|$  *součtová, „manhattanská“*
- $\|\vec{x}\|_q = \left( \sum_{i=1}^n |x_i|^q \right)^{\frac{1}{q}}$ ,  $q \geq 1$  *společné zobecnění předchozích*

Co není norma: Pro  $q = 1/2$  bychom dostali

$$\|\vec{x}\|_{1/2} = \left( \sum_{i=1}^n \sqrt{|x_i|} \right)^2.$$

Porušuje trojúhelníkovou nerovnost:

$$\begin{aligned} (1, 1) &= (1, 0) + (0, 1), \\ \|(1, 1)\|_{1/2} &= 2^2 = 4 > \|(1, 0)\|_{1/2} + \|(0, 1)\|_{1/2} = 1 + 1 = 2. \end{aligned}$$

**Změna měřítka:** Je-li  $\|\cdot\|_v$  vektorová norma a  $r > 0$ , pak  $\|\vec{x}\|_u = r \|\vec{x}\|_v$  je také vektorová norma.

**Tvrzení 5.1** Norma je **konvexní** funkce.

**Důkaz.** Pro  $c \in \langle 0, 1 \rangle$ :

$$\|c\vec{x} + (1-c)\vec{y}\|_v \leq \|c\vec{x}\|_v + \|(1-c)\vec{y}\|_v = c\|\vec{x}\|_v + (1-c)\|\vec{y}\|_v$$

□

**Definice 5.2** Posloupnost vektorů  $\vec{x}^{(0)}, \vec{x}^{(1)}, \vec{x}^{(2)}, \dots$  **konverguje k vektoru**  $\vec{x} \in \mathbb{R}^n$ , jestliže

$$\lim_{k \rightarrow \infty} x_i^{(k)} = x_i \quad \text{pro } i = 1, 2, \dots, n.$$

**Věta 5.1** Posloupnost vektorů  $\vec{x}^{(0)}, \vec{x}^{(1)}, \vec{x}^{(2)}, \dots$  konverguje k vektoru  $\vec{x} \in \mathbb{R}^n$ , právě když

$$\lim_{k \rightarrow \infty} \|\vec{x}^{(k)} - \vec{x}\|_v = 0,$$

kde za normu můžeme zvolit libovolnou z výše uvedených vektorových norem.

(Konvergence nezáleží na volbě normy.)

Stejně můžeme definovat normy i pro matice; ty také tvoří vektorový prostor, ale má to háček:

Nechceme, aby „malý vektor“ násobený „malou maticí“ mohl dát „velký výsledek“.

Příklad (maximová norma  $\|\cdot\|_r$  vektorů i matic):

$$(1, 1) \cdot \begin{pmatrix} 0.9 & 0.9 \\ 0.9 & 0.9 \end{pmatrix} = (1.8, 1.8),$$

$$\|(1, 1)\|_r \cdot \left\| \begin{pmatrix} 0.9 & 0.9 \\ 0.9 & 0.9 \end{pmatrix} \right\|_r = 0.9 \cdot 1 = 0.9 < \|(1.8, 1.8)\|_r = 1.8.$$

Hůře pro „velkou“ dimenzi  $n \gg 1$ :

$$(1/2, \dots, 1/2) \cdot \begin{pmatrix} 1/2 & \dots & 1/2 \\ \vdots & \ddots & \vdots \\ 1/2 & \dots & 1/2 \end{pmatrix} = (n/4, \dots, n/4),$$

$$\|(1/2, \dots, 1/2)\|_r \cdot \left\| \begin{pmatrix} 1/2 & \dots & 1/2 \\ \vdots & \ddots & \vdots \\ 1/2 & \dots & 1/2 \end{pmatrix} \right\|_r = \frac{1}{2} \cdot \frac{1}{2} = \frac{1}{4} \ll \|(n/4, \dots, n/4)\|_r = \frac{n}{4}.$$

Ještě hůře pro  $k \times$  opakované násobení:

$$(1/2, \dots, 1/2) \cdot \begin{pmatrix} 1/2 & \dots & 1/2 \\ \vdots & \ddots & \vdots \\ 1/2 & \dots & 1/2 \end{pmatrix}^k = \left( \frac{n^k}{2^{k+1}}, \dots, \frac{n^k}{2^{k+1}} \right),$$

$$\|(1/2, \dots, 1/2)\|_r \cdot \left\| \begin{pmatrix} 1/2 & \dots & 1/2 \\ \vdots & \ddots & \vdots \\ 1/2 & \dots & 1/2 \end{pmatrix} \right\|_r^k = \frac{1}{2^{k+1}} \ll \left\| \left( \frac{n^k}{2^{k+1}}, \dots, \frac{n^k}{2^{k+1}} \right) \right\|_r = \frac{n^k}{2^{k+1}}.$$

Pro normu  $\|\cdot\|_M$  (čtvercových) matic a normu  $\|\cdot\|_v$  (sloupcových) vektorů budeme požadovat

$$\forall \vec{x} \in \mathbb{R}^n : \|\mathbf{B}\vec{x}\|_v \leq \|\mathbf{B}\|_M \|\vec{x}\|_v. \quad (11)$$

Taková norma  $\|\cdot\|_M$  se nazývá **souhlasná** s vektorovou normou  $\|\cdot\|_v$ .

**Věta 5.2** Ke každé (vektorové) normě  $\|\cdot\|_v$  lze najít souhlasnou normu matic  $\|\cdot\|_V$ . Stačí pro každou matici  $\mathbf{B} \in \mathbb{R}^{n,n}$  definovat  $\|\mathbf{B}\|_V$  jako **nejmenší** číslo, které splňuje (11). Takto definovaná funkce  $\|\cdot\|_V: \mathbb{R}^{n,n} \rightarrow \mathbb{R}$  je norma a nazývá se **operátorová norma indukovaná** (vektorovou) normou  $\|\cdot\|_v$ . Ekvivalentně ji lze zavést předpisem

$$\|\mathbf{B}\|_V = \sup_{\vec{x} \neq \vec{o}} \frac{\|\mathbf{B}\vec{x}\|_v}{\|\vec{x}\|_v} = \sup_{\|\vec{u}\|_v=1} \|\mathbf{B}\vec{u}\|_v, \quad \left(\vec{u} = \frac{\vec{x}}{\|\vec{x}\|_v}\right).$$

**Důsledek 5.1** Norma matic  $\|\cdot\|_M$  je souhlasná s vektorovou normou  $\|\cdot\|_v$ , právě když  $\|\cdot\|_M \geq \|\cdot\|_V$ , kde  $\|\cdot\|_V$  je operátorová norma indukovaná vektorovou normou  $\|\cdot\|_v$ .

Operátorová norma je **nejmenší souhlasná** maticová norma (pro danou vektorovou normu).

**Důsledek 5.2**  $\|\mathbf{E}\|_V = 1$ .

**Poznámka:** Zde máme prostor konečné dimenze, kde sup je max.

Pro  $k \times$  opakované násobení vektoru maticí chceme

$$\|\mathbf{B}^k \vec{x}\|_v \leq \|\mathbf{B}^k\|_M \|\vec{x}\|_v \leq \|\mathbf{B}\|_M^k \|\vec{x}\|_v,$$

proto budeme požadovat

$$\|\mathbf{B}^k\|_M \leq \|\mathbf{B}\|_M^k$$

a obecněji i pro násobení různých matic

$$\|\mathbf{B} \cdot \mathbf{C}\|_M \leq \|\mathbf{B}\|_M \|\mathbf{C}\|_M \quad (\text{Schwarzova nerovnost}).$$

Ne všechny „normy matic“ splňují Schwarzovu nerovnost:

**Příklad 5.3** (maximová norma  $\|\cdot\|_r$ )

$$\begin{aligned} & \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} = \begin{pmatrix} 2 & 2 \\ 2 & 2 \end{pmatrix}, \\ & \left\| \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \right\|_r \cdot \left\| \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \right\|_r = 1 \cdot 1 = 1 < \left\| \begin{pmatrix} 2 & 2 \\ 2 & 2 \end{pmatrix} \right\|_r = 2. \end{aligned}$$

**Definice 5.3** **Maticová norma** je zobrazení  $\|\cdot\|_M: \mathbb{R}^{n,n} \rightarrow \mathbb{R}$  splňující

- $\|\mathbf{B}\|_M \geq 0$ , přičemž  $\|\mathbf{B}\|_M = 0 \Leftrightarrow \mathbf{B} = \mathbf{O}$
- $\|c\mathbf{B}\|_M = |c| \|\mathbf{B}\|_M$ ,
- $\|\mathbf{B} + \mathbf{C}\|_M \leq \|\mathbf{B}\|_M + \|\mathbf{C}\|_M$  (trojúhelníková nerovnost),
- $\|\mathbf{B} \cdot \mathbf{C}\|_M \leq \|\mathbf{B}\|_M \|\mathbf{C}\|_M$  (Schwarzova nerovnost).

**Věta 5.3** Každá **operátorová norma** je **maticová norma**.

**Důkaz.** Netriviální je jen důkaz Schwarzovy nerovnosti:

Je-li  $\|\mathbf{B}\mathbf{C}\|_V > 0$ ,  $\vec{x} \neq \vec{o}$ , pak  $\vec{y} := \mathbf{C}\vec{x} \neq \vec{o}$ ,

$$\begin{aligned} \|\mathbf{B}\mathbf{C}\|_V &= \max_{\vec{x} \neq \vec{o}} \frac{\|\mathbf{B}\mathbf{C}\vec{x}\|_v}{\|\vec{x}\|_v} = \max_{\vec{x}; \mathbf{C}\vec{x} \neq \vec{o}} \left( \frac{\|\mathbf{B}\mathbf{C}\vec{x}\|_v}{\|\mathbf{C}\vec{x}\|_v} \frac{\|\mathbf{C}\vec{x}\|_v}{\|\vec{x}\|_v} \right) \\ &\leq \left( \max_{\vec{x}; \mathbf{C}\vec{x} \neq \vec{o}} \frac{\|\mathbf{B}\mathbf{C}\vec{x}\|_v}{\|\mathbf{C}\vec{x}\|_v} \right) \left( \max_{\vec{x} \neq \vec{o}} \frac{\|\mathbf{C}\vec{x}\|_v}{\|\vec{x}\|_v} \right) \\ &= \left( \max_{\vec{y} \in \mathbf{C}\mathbb{R}^n; \vec{y} \neq \vec{o}} \frac{\|\mathbf{B}\vec{y}\|_v}{\|\vec{y}\|_v} \right) \left( \max_{\vec{x} \neq \vec{o}} \frac{\|\mathbf{C}\vec{x}\|_v}{\|\vec{x}\|_v} \right) \\ &\leq \left( \max_{\vec{z} \neq \vec{o}} \frac{\|\mathbf{B}\vec{z}\|_v}{\|\vec{z}\|_v} \right) \left( \max_{\vec{x} \neq \vec{o}} \frac{\|\mathbf{C}\vec{x}\|_v}{\|\vec{x}\|_v} \right) \\ &= \|\mathbf{B}\|_V \cdot \|\mathbf{C}\|_V. \end{aligned}$$

□

**Příklad 5.4** Operátorová norma indukovaná součtovou normou vektorů:

$$\|\mathbf{B}\|_S = \sup_{\|\vec{u}\|_s=1} \|\mathbf{B}\vec{u}\|_s.$$

„Jednotková koule“ v normě  $\|\cdot\|_s$  je konvexním obalem svých vrcholů, tj. vektorů tvaru  $(\pm 1, 0, 0, \dots, 0)$ ,  $(0, \pm 1, 0, \dots, 0)$ ,  $\dots$ ,  $(0, 0, \dots, 0, \pm 1)$ .

Její obraz v lineárním zobrazení reprezentovaném maticí  $\mathbf{B}$  je konvexním obalem obrazů těchto vrcholů, což jsou  $\pm$  sloupce matice  $\mathbf{B}$ .

Maximum normy se nabývá v některém z těchto obrazů,

$$\begin{aligned} \|\mathbf{B}\|_S &= \max_j \|(b_{1,j}, \dots, b_{n,j})\|_s = \\ &= \max_j \sum_{i=1}^n |b_{i,j}| = \\ &= \|(\|(b_{1,1}, \dots, b_{n,1})\|_s, \|(b_{1,2}, \dots, b_{n,2})\|_s, \dots, \|(b_{1,n}, \dots, b_{n,n})\|_s)\|_r \end{aligned}$$

To je tzv. **sloupcová** maticová norma.

**Příklad 5.5** Operátorová norma indukovaná maximovou normou vektorů:

$$\|\mathbf{B}\|_R = \sup_{\|\vec{u}\|_r=1} \|\mathbf{B}\vec{u}\|_r.$$

„Jednotková koule“ v normě  $\|\cdot\|_r$  je hyperkrychle; je konvexním obalem svých vrcholů, tj. vektorů tvaru  $(\pm 1, \pm 1, \dots, \pm 1)$ .

Její obraz v lineárním zobrazení reprezentovaném maticí  $\mathbf{B}$  je konvexním obalem obrazů těchto vrcholů. Maximum normy nastává pro některý z nich, řekněme  $\vec{u}$ , a to pro  $k$ -tou složku obrazu,  $(\mathbf{B}\vec{u})_k = \sum_j b_{k,j} u_j$ ; ta je maximální, pokud členy  $b_{k,j} u_j$  mají pro všechna  $j$  stejná znaménka (bez újmy na obecnosti kladná), pak

$$(\mathbf{B}\vec{u})_k = \sum_j b_{k,j} u_j = \sum_j b_{k,j} \operatorname{sign}(b_{k,j}) = \sum_j |b_{k,j}|,$$

kde  $k$  bylo vybráno jako optimální ze všech  $i \in \{1, 2, \dots, n\}$ . Dostáváme operátorovou normu

$$\begin{aligned} \|\mathbf{B}\|_R &= \max_{i=1, \dots, n} \sum_{j=1}^n |b_{i,j}| = \\ &= \max_{i=1, \dots, n} \|(b_{i,1}, \dots, b_{i,n})\|_s = \\ &= \|(\|(b_{1,1}, \dots, b_{1,n})\|_s, \|(b_{2,1}, \dots, b_{2,n})\|_s, \dots, \|(b_{n,1}, \dots, b_{n,n})\|_s)\|_r \end{aligned}$$

To je tzv. **řádková** maticová norma.

**Příklad 5.6**

$$\|\mathbf{B}\|_F = \sqrt{\sum_{i=1}^n \sum_{j=1}^n b_{i,j}^2}$$

je maticová norma zvaná **euklidovská** nebo **Frobeniova**.

Nemůže to být operátorová norma, neboť pro jednotkovou matici  $\|E\|_F = \sqrt{n} \neq 1$ . **Není** to operátorová norma indukovaná euklidovskou normou vektorů,

$$\sup_{\|\vec{u}\|_e=1} \|\mathbf{B}\vec{u}\|_e,$$

pro tu neexistuje jednoduchý vzorec, protože je např. zdola omezená vlastními čísly matice  $\mathbf{B}$ .

Později ukážeme, že Frobeniova norma matic je **souhlasná** s euklidovskou normou vektorů.

**Příklad 5.7** Pro libovolnou maticovou normu  $\|\cdot\|_M$

$$\begin{aligned} \left\| \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}^k \right\|_M &= 2^k \left\| \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \right\|_M \rightarrow \infty \text{ pro } k \rightarrow \infty, \\ \left\| \begin{pmatrix} \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} \end{pmatrix}^k \right\|_M &= \frac{1}{2^k} \left\| \begin{pmatrix} \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} \end{pmatrix} \right\|_M \rightarrow 0 \text{ pro } k \rightarrow \infty. \end{aligned}$$

**Konvergence nekonečného součinu:** Dosadíme  $\mathbf{B} := \mathbf{B}^k$ :

$$\|\mathbf{B}^k \cdot \mathbf{C}\|_M \leq \|\mathbf{B}\|_M^k \|\mathbf{C}\|_M \implies \|\mathbf{B}^k \cdot \mathbf{C}\|_M \rightarrow 0 \text{ pro } \|\mathbf{B}\|_M < 1, \quad k \rightarrow \infty.$$

**Tvrzení 5.2** Operátorová norma je bezrozměrná, změna měřítka vektorové normy nemá vliv. Pokud vektorovou normu  $\|\cdot\|_v$  vynásobíme  $r > 0$ ,  $\|\vec{x}\|_u := r \|\vec{x}\|_v$ , operátorová norma se nezmění:

$$\max_{\vec{x} \neq \vec{0}} \frac{\|\mathbf{B} \vec{x}\|_u}{\|\vec{x}\|_u} = \max_{\vec{x} \neq \vec{0}} \frac{r \|\mathbf{B} \vec{x}\|_v}{r \|\vec{x}\|_v} = \max_{\vec{x} \neq \vec{0}} \frac{\|\mathbf{B} \vec{x}\|_v}{\|\vec{x}\|_v} = \|\mathbf{B}\|_V.$$

**Věta 5.4** Ke každé maticové normě  $\|\cdot\|_M$  existuje alespoň jedna souhlasná vektorová norma  $\|\cdot\|_v$ , a to  $\|\vec{x}\|_v = \|\mathbf{X}\|_M$ , kde

$$\mathbf{X} = \begin{pmatrix} x_1 & 0 & 0 & \dots & 0 \\ x_2 & 0 & 0 & \dots & 0 \\ \vdots & & & & \\ x_n & 0 & 0 & \dots & 0 \end{pmatrix}.$$

**Důkaz.**

$$\|\mathbf{B} \vec{x}\|_v = \left\| \begin{pmatrix} \sum_{j=1}^n b_{1,j} x_j & 0 & \dots & 0 \\ \sum_{j=1}^n b_{2,j} x_j & 0 & \dots & 0 \\ \vdots & & & \\ \sum_{j=1}^n b_{n,j} x_j & 0 & \dots & 0 \end{pmatrix} \right\|_M = \|\mathbf{B} \cdot \mathbf{X}\|_M \leq \|\mathbf{B}\|_M \|\mathbf{X}\|_M = \|\mathbf{B}\|_M \|\vec{x}\|_v.$$

□

**Důsledek 5.3**  $\|\cdot\|_F$  je souhlasná s  $\|\cdot\|_e$ .

**Důkaz.**

$$\left\| \begin{pmatrix} x_1 & 0 & 0 & \dots & 0 \\ x_2 & 0 & 0 & \dots & 0 \\ \vdots & & & & \\ x_n & 0 & 0 & \dots & 0 \end{pmatrix} \right\|_F^2 = \sum_{i=1}^n x_i^2 = \|\vec{x}\|_e^2.$$

□

**Definice 5.4** *Posloupnost matic*  $\mathbf{B}^{(0)} = (b_{i,j}^{(0)})_{i,j=1}^n$ ,  $\mathbf{B}^{(1)} = (b_{i,j}^{(1)})_{i,j=1}^n$ ,  $\mathbf{B}^{(2)} = (b_{i,j}^{(2)})_{i,j=1}^n, \dots$  **konverguje k matici**  $\mathbf{B} = (b_{i,j})_{i,j=1}^n$ , *jestliže*

$$\lim_{k \rightarrow \infty} b_{i,j}^{(k)} = b_{i,j} \quad \text{pro každé } i, j = 1, 2, \dots, n.$$

( $\mathbf{B}^{(k)}$  ...  $k$ -tý člen posloupnosti)

**Věta 5.5** Posloupnost matic  $\mathbf{B}^{(0)}, \mathbf{B}^{(1)}, \mathbf{B}^{(2)}, \dots$  konverguje k matici  $\mathbf{B}$ , právě když

$$\lim_{k \rightarrow \infty} \|\mathbf{B}^{(k)} - \mathbf{B}\|_M = 0,$$

pro **nějakou** maticovou normu  $\|\cdot\|_M$ .

(Konvergence nezáleží na volbě normy.)

**Definice 5.5** *Matice*  $\mathbf{B}$  je **konvergentní**, *jestliže posloupnost matic*  $\mathbf{B}, \mathbf{B}^2, \mathbf{B}^3, \mathbf{B}^4, \dots$  *konverguje k nulové matici*. V opačném případě řekneme, že matice  $\mathbf{B}$  je **divergentní**.

### 5.3.2 Vlastní čísla a spektrální poloměr

**Definice 5.6** Číslo  $\lambda \in \mathbb{C}$  je **vlastní (charakteristické) číslo matice**  $\mathbf{B} \in \mathbb{R}^{n,n}$ , též **vlastní číslo lineárního zobrazení**  $\vec{x} \mapsto \mathbf{B}\vec{x}$ , jestliže existuje nenulový vektor  $\vec{x} \in \mathbb{C}^n$  (**vlastní (charakteristický) vektor**) splňující  $\vec{x} \mapsto \lambda\vec{x}$ , tj.  $\mathbf{B}\vec{x} = \lambda\vec{x}$ .

**Věta 5.6** Číslo  $\lambda$  je vlastní číslo matice  $\mathbf{B}$ , právě když  $\det(\mathbf{B} - \lambda\mathbf{E}) = 0$ .

**Důkaz.** Je-li  $\lambda$  vlastní číslo matice  $\mathbf{B}$ , pak  $\exists \vec{x} \neq \vec{0} : \lambda\vec{x} = \mathbf{B}\vec{x}$ , tedy  $\mathbf{B}\vec{x} - \lambda\vec{x} = \vec{0}$ ,  $(\mathbf{B} - \lambda\mathbf{E})\vec{x} = \vec{0}$ .

Nechť  $\det(\mathbf{B} - \lambda\mathbf{E}) \neq 0$ , tedy matice  $\mathbf{B} - \lambda\mathbf{E}$  je regulární. Homogenní soustava rovnic s regulární maticí soustavy má pouze triviální řešení  $\implies$  spor.

Naopak z předpokladu  $\det(\mathbf{B} - \lambda\mathbf{E}) = 0$  přímo plyne, že matice  $\mathbf{B} - \lambda\mathbf{E}$  je singulární a  $\exists \vec{x} \neq \vec{0} : (\mathbf{B} - \lambda\mathbf{E})\vec{x} = \vec{0}$ . Tedy  $\mathbf{B}\vec{x} - \lambda\vec{x} = \vec{0}$ ,  $\lambda\vec{x} = \mathbf{B}\vec{x}$ .  $\square$

### 5.3.3 Výpočet vlastních čísel

Pro matice malých řádů:

- Vypočteme  $\det(\mathbf{B} - \lambda\mathbf{E})$ , což je polynom v proměnné  $\lambda$  stupně  $n$  (**charakteristický polynom matice**  $\mathbf{B}$ ).
- Vlastní čísla  $\lambda_i$  jsou kořeny charakteristického polynomu matice  $\mathbf{B}$  (rovnice  $\det(\mathbf{B} - \lambda\mathbf{E}) = 0$  je **charakteristická rovnice**).

**Poznámka:** Za vlastní čísla nadále považujeme všechny kořeny charakteristické rovnice včetně komplexních. (O těch bychom měli správně hovořit jen v komplexním vektorovém prostoru; v reálném není násobení komplexním číslem definováno.)

**Definice 5.7** **Spektrální poloměr** matice  $\mathbf{B} \in \mathbb{R}^{n,n}$  je číslo

$$\varrho(\mathbf{B}) = \max_{i=1,2,\dots,n} |\lambda_i|,$$

kde  $\lambda_i, i = 1, 2, \dots, n$ , jsou vlastní čísla matice  $\mathbf{B}$  (včetně komplexních).

**Věta 5.7** Každá maticová norma  $\|\cdot\|_M$  splňuje

$$\|\mathbf{B}\|_M \geq \varrho(\mathbf{B}).$$

**Důkaz.** (Jen pro případ, že vlastní číslo  $\lambda$  s největší absolutní hodnotou je reálné.)

Nechť  $\vec{y}$  je vlastní vektor příslušný  $\lambda$ ,  $\mathbf{B}\vec{y} = \lambda\vec{y}$ .

Maticová norma  $\|\cdot\|_M$  je souhlasná s nějakou vektorovou normou  $\|\cdot\|_v$ , tedy větší nebo rovna příslušné operátorové normě  $\|\cdot\|_V$ , indukované  $\|\cdot\|_v$ , pro kterou platí

$$\|\mathbf{B}\|_M \geq \|\mathbf{B}\|_V = \max_{\vec{x} \neq \vec{0}} \frac{\|\mathbf{B}\vec{x}\|_v}{\|\vec{x}\|_v} \geq \frac{\|\mathbf{B}\vec{y}\|_v}{\|\vec{y}\|_v} = \frac{|\lambda| \|\vec{y}\|_v}{\|\vec{y}\|_v} = |\lambda| = \varrho(\mathbf{B}).$$

$\square$

**Věta 5.8** Matice  $\mathbf{B}$  je konvergentní  $\iff \varrho(\mathbf{B}) < 1$ .

**Věta 5.9** **Postačující podmínka konvergentnosti matice**  $\mathbf{B}$ :

- $\|\mathbf{B}\|_M < 1$  pro **nějakou** maticovou normu, neboli
- lineární zobrazení  $\vec{x} \mapsto \mathbf{B}\vec{x}$ , reprezentované maticí  $\mathbf{B}$ , je kontraktivní (vzhledem k nějaké vektorové normě, souhlasné s maticovou normou  $\|\cdot\|_M$ ).

**Příklad 5.8** Matice

$$\mathbf{B} = \begin{pmatrix} 0.8 & 0.8 \\ 0.1 & 0.1 \end{pmatrix}$$

je konvergentní,  $\|\mathbf{B}\|_S = 0.9 < 1$ , ačkoli  $\|\mathbf{B}\|_R = 1.6 > 1$ ,  $\|\mathbf{B}\|_F = \sqrt{1.3} > 1$ .



### 5.3.4 Maticové iterační metody

Hledáme posloupnost vektorů  $\vec{x}^{(k)} \in \mathbb{R}^n$  splňující  $\vec{x}^{(k)} \rightarrow \vec{x}$ , tj.  $\|\vec{x}^{(k)} - \vec{x}\|_v \rightarrow 0$ .

Použijeme rekurentní vzorec

$$\vec{x}^{(k+1)} = F_k(\vec{x}^{(k)}, \vec{x}^{(k-1)}, \dots, \vec{x}^{(k-m)}).$$

Omezíme se na **lineární jednobodové stacionární maticové iterační metody**, tj.  $F_k$  nezávisí na  $k$ , závisí jen na  $\vec{x}^{(k)}$ , a to „lineárně“ (správně **afinně**),

$$\vec{x}^{(k+1)} = \mathbf{B} \vec{x}^{(k)} + \vec{c}.$$

Např.

$$\begin{aligned} \vec{b} &= \mathbf{A} \vec{x}, \\ \vec{x} + \vec{b} &= \vec{x} + \mathbf{A} \vec{x} = (\mathbf{E} + \mathbf{A}) \vec{x}, \\ \vec{x} &= (\mathbf{E} + \mathbf{A}) \vec{x} - \vec{b}, \\ \vec{x}^{(k+1)} &= \underbrace{(\mathbf{E} + \mathbf{A})}_{\mathbf{B}} \vec{x}^{(k)} - \underbrace{\vec{b}}_{\vec{c}}. \end{aligned}$$

**Vektor chyby:**

$$\begin{aligned} \vec{\varepsilon}^{(k)} &= \vec{x}^{(k)} - \vec{x} \quad (\text{předpokládáme } \neq \vec{0}) \\ \vec{x}^{(k)} &= \mathbf{B} \vec{x}^{(k-1)} + \vec{c} \\ \vec{x} &= \mathbf{B} \vec{x} + \vec{c} \\ \vec{x}^{(k)} - \vec{x} &= \mathbf{B} (\vec{x}^{(k-1)} - \vec{x}) = \mathbf{B}^2 (\vec{x}^{(k-2)} - \vec{x}) = \dots = \mathbf{B}^k (\vec{x}^{(0)} - \vec{x}) \\ \vec{\varepsilon}^{(k)} &= \mathbf{B} \vec{\varepsilon}^{(k-1)} = \dots = \mathbf{B}^k \vec{\varepsilon}^{(0)} \\ \lim_{k \rightarrow \infty} \mathbf{B}^k &= \mathbf{O} \quad \Rightarrow \quad \lim_{k \rightarrow \infty} \vec{\varepsilon}^{(k)} = \vec{0}. \end{aligned}$$

**Věta 5.10 Nutná a postačující podmínka konvergence iterační metody tvaru  $\vec{x}^{(k+1)} = \mathbf{B} \vec{x}^{(k)} + \vec{c}$  je  $\rho(\mathbf{B}) < 1$ . Pak pro vzdálenost od limity  $\vec{x}$  platí**

$$\|\vec{x}^{(k)} - \vec{x}\|_v \leq \frac{\rho(\mathbf{B})}{1 - \rho(\mathbf{B})} \|\vec{x}^{(k)} - \vec{x}^{(k-1)}\|_v.$$

**Věta 5.11 Postačující podmínka konvergence iterační metody tvaru  $\vec{x}^{(k+1)} = \mathbf{B} \vec{x}^{(k)} + \vec{c}$  je  $\|\mathbf{B}\|_M < 1$  (pro nějakou maticovou normu). Odhad chyby je**

$$\|\vec{x}^{(k)} - \vec{x}\|_v \leq \frac{\|\mathbf{B}\|_M}{1 - \|\mathbf{B}\|_M} \|\vec{x}^{(k)} - \vec{x}^{(k-1)}\|_v,$$

(normy matice a vektorů můžeme volit, ale musí být souhlasné).

(Počáteční odhad i vektorová norma mohou být libovolné.)

### 5.3.5 Jacobiova iterační metoda (JIM)

Vyjádříme matici  $\mathbf{A}$  ve tvaru

$$\mathbf{A} = \mathbf{D} + \mathbf{L} + \mathbf{U},$$

kde  $\mathbf{D}$  je diagonální,  $\mathbf{L}$  ostře dolní trojúhelníková a  $\mathbf{U}$  ostře horní trojúhelníková.

$$\begin{aligned} \mathbf{A} \vec{x} &= (\mathbf{D} + \mathbf{L} + \mathbf{U}) \vec{x} = \mathbf{D} \vec{x} + (\mathbf{L} + \mathbf{U}) \vec{x} = \vec{b}, \\ \mathbf{D} \vec{x} &= -(\mathbf{L} + \mathbf{U}) \vec{x} + \vec{b}, \\ \vec{x} &= \mathbf{D}^{-1} (-(\mathbf{L} + \mathbf{U}) \vec{x} + \vec{b}), \end{aligned}$$

volíme

$$\vec{x}^{(k+1)} = \underbrace{-\mathbf{D}^{-1}(\mathbf{L} + \mathbf{U})}_{\mathbf{B}_{JIM}} \vec{x}^{(k)} + \underbrace{\mathbf{D}^{-1} \vec{b}}_{\vec{c}_{JIM}}.$$

Předpokládáme, že hlavní diagonála matice  $\mathbf{A}$  neobsahuje žádný nulový prvek (toho lze dosáhnout výměnou řádků nebo sloupců).

Chceme, aby na diagonále byly „velké“ prvky.

Po složkách:

$$\begin{aligned} x_1 &= -\frac{1}{a_{1,1}} (a_{1,2} x_2 + a_{1,3} x_3 + \dots + a_{1,n} x_n) + \frac{b_1}{a_{1,1}} \\ x_2 &= -\frac{1}{a_{2,2}} (a_{2,1} x_1 + a_{2,3} x_3 + \dots + a_{2,n} x_n) + \frac{b_2}{a_{2,2}} \\ &\vdots \\ x_n &= -\frac{1}{a_{n,n}} (a_{n,1} x_1 + a_{n,2} x_2 + \dots + a_{n,n-1} x_{n-1}) + \frac{b_n}{a_{n,n}} \\ x_i^{(k+1)} &= -\frac{1}{a_{i,i}} \left( \sum_{j=1}^{i-1} a_{i,j} x_j^{(k)} + \sum_{j=i+1}^n a_{i,j} x_j^{(k)} \right) + \frac{b_i}{a_{i,i}}, \quad i = 1, 2, \dots, n. \end{aligned}$$

Podmínka ukončení:  $\|\vec{x}^{(k+1)} - \vec{x}^{(k)}\|_v < \varepsilon$ .

### 5.3.6 Gaussova-Seidelova iterační metoda (GSM)

Každá už vypočtená složka vektoru  $\vec{x}^{(k+1)}$  se ihned použije v dalším výpočtu.

$$x_i^{(k+1)} = -\frac{1}{a_{i,i}} \left( \sum_{j=1}^{i-1} a_{i,j} x_j^{(k+1)} + \sum_{j=i+1}^n a_{i,j} x_j^{(k)} \right) + \frac{b_i}{a_{i,i}}, \quad i = 1, 2, \dots, n.$$

Pro realizaci výpočtu stačí pouze jeden vektor  $\vec{x}$ .

Maticový tvar:

$$\begin{aligned} (\mathbf{D} + \mathbf{L}) \vec{x} &= -\mathbf{U} \vec{x} + \vec{b}, \\ \vec{x} &= -(\mathbf{D} + \mathbf{L})^{-1} (\mathbf{U} \vec{x} - \vec{b}), \\ \vec{x}^{(k+1)} &= \underbrace{-(\mathbf{D} + \mathbf{L})^{-1} \mathbf{U}}_{\mathbf{B}_{GSM}} \vec{x}^{(k)} + \underbrace{(\mathbf{D} + \mathbf{L})^{-1} \vec{b}}_{\vec{c}_{GSM}}. \end{aligned}$$

Může se stát, že konverguje pouze jedna z těchto metod (nebo žádná).

**Věta 5.12** [Num. Recipes] Pokud JIM i GSM konvergují, pak  $0 \leq \rho(\mathbf{B}_{GSM}) = \rho(\mathbf{B}_{JIM})^2 \leq \rho(\mathbf{B}_{JIM}) < 1$ , takže GSM konverguje rychleji.

(Můžeme očekávat, že bude stačit asi 2× méně kroků.)

Problém: určení vlastních čísel matice.

**Definice 5.8** Matice  $\mathbf{A} \in \mathbb{R}^{n,n}$  se nazývá **ostře diagonálně dominantní**, jestliže

$$|a_{i,i}| > \sum_{j=1, j \neq i}^n |a_{i,j}| \quad \text{pro každé } i = 1, 2, \dots, n.$$

**Věta 5.13** Nechť matice  $\mathbf{A} \in \mathbb{R}^{n,n}$  je ostře diagonálně dominantní, pak JIM i GSM konverguje pro libovolnou počáteční iteraci.

**Důkaz.** Pro JIM:  $\mathbf{B}_{\text{JIM}}$  má prvky

$$b_{i,j} = \begin{cases} -\frac{a_{i,j}}{a_{i,i}} & i \neq j, \\ 0 & i = j, \end{cases}$$

$$\|\mathbf{B}\|_R = \max_{i=1,\dots,n} \sum_{j=1}^n |b_{i,j}| = \max_{i=1,\dots,n} \sum_{j=1, j \neq i}^n \frac{|a_{i,j}|}{|a_{i,i}|} = \max_{i=1,\dots,n} \frac{1}{|a_{i,i}|} \sum_{j=1, j \neq i}^n |a_{i,j}| < 1.$$

□

**Poznámka:** Pro transponovanou matici dostaneme jinou podmínku, ale stejně užitečnou, neboť zajistí  $\|\mathbf{B}\|_S < 1$ .

**Poznámka:** Ostře diagonálně dominantní matice vychází např. u soustav lineárních rovnic, kterými se řeší osvětlení scény za předpokladu lambertovských povrchů. (Soustavy jsou extrémně velké, mnohdy se ani nevejdou do paměti.) Navíc při změně osvětlení se změní jen pravá strana.

**Definice 5.9** Matice  $\mathbf{A} \in \mathbb{R}^{n,n}$  se nazývá **pozitivně definitní**, jestliže pro každý nenulový vektor  $\vec{x} \in \mathbb{R}^n$  platí

$$\vec{x}^T \mathbf{A} \vec{x} > 0.$$

**Věta 5.14** Nechť matice  $\mathbf{A} \in \mathbb{R}^{n,n}$  je symetrická a pozitivně definitní, pak GSM konverguje.

**Poznámka:** Symetrická a pozitivně definitní je např. matice soustavy normálních rovnic u metody nejmenších čtverců.

**Poznámka:** Matice soustavy rovnic pro koeficienty kubického splinu je ostře diagonálně dominantní, navíc řídká, takže složitost jedné iterace je úměrná  $n$ . Při rovnoměrném rozdělení uzlových bodů je navíc symetrická, což dovoluje řešení s téměř lineární složitostí [Spielman, Teng].

### 5.3.7 Superrelaxační metoda (SOR – Successive OverRelaxation method)

Konvergenci urychlí jakákoliv modifikace, která vede ke zmenšení spektrálního poloměru matice  $\mathbf{B}$ .

$$\begin{aligned} \mathbf{D} \vec{x} &= \mathbf{D} \vec{x} + \omega \overbrace{(-\mathbf{A} \vec{x} + \vec{b})}^{\vec{\sigma}} = \\ &= \mathbf{D} \vec{x} + \omega \left( (-\mathbf{L} - \mathbf{D} - \mathbf{U}) \vec{x} + \vec{b} \right), \\ \mathbf{D} \vec{x} + \omega \mathbf{L} \vec{x} &= (1 - \omega) \mathbf{D} \vec{x} - \omega \mathbf{U} \vec{x} + \omega \vec{b}, \quad (\text{GSM: } \omega := 1) \\ (\mathbf{D} + \omega \mathbf{L}) \vec{x} &= [(1 - \omega) \mathbf{D} - \omega \mathbf{U}] \vec{x} + \omega \vec{b}, \\ \vec{x} &= (\mathbf{D} + \omega \mathbf{L})^{-1} \left( [(1 - \omega) \mathbf{D} - \omega \mathbf{U}] \vec{x} + \omega \vec{b} \right) = \\ &= (\mathbf{D} + \omega \mathbf{L})^{-1} [(1 - \omega) \mathbf{D} - \omega \mathbf{U}] \vec{x} + \omega (\mathbf{D} + \omega \mathbf{L})^{-1} \vec{b}, \\ \vec{x}^{(k+1)} &= \underbrace{(\mathbf{D} + \omega \mathbf{L})^{-1} [(1 - \omega) \mathbf{D} - \omega \mathbf{U}]}_{\mathbf{B}_\omega} \vec{x}^{(k)} + \underbrace{\omega (\mathbf{D} + \omega \mathbf{L})^{-1} \vec{b}}_{\vec{c}_\omega}, \end{aligned}$$

kde  $\omega$  je **relaxační faktor**. (Pro  $\omega = 1$  dostáváme GSM.) Po složkách:

$$x_i^{(k+1)} = \frac{\omega}{a_{i,i}} \left( -\sum_{j=1}^{i-1} a_{i,j} x_j^{(k+1)} - \sum_{j=i+1}^n a_{i,j} x_j^{(k)} + b_j \right) + (1 - \omega) x_i^{(k)}$$

**Věta 5.15** Metoda SOR konverguje pro libovolnou počáteční iteraci právě tehdy, když  $\rho(\mathbf{B}_\omega) < 1$ .

**Věta 5.16** (Ostrowského) Nechť  $\mathbf{A}$  je symetrická matice s kladnými prvky na diagonále. Pak platí  $\rho(\mathbf{B}_\omega) < 1$  právě tehdy, když matice  $\mathbf{A}$  je pozitivně definitní a  $0 < \omega < 2$ .

**Poznámka:** Často je lepší relaxační faktor nadhodnotit.

## 5.4 Jaký postup volit?

**Poznámka:** Složitost řešení soustavy  $n$  lineárních rovnic o  $n$  neznámých s jedním vektorem pravých stran pomocí GEM a dalších přímých metod je úměrná  $n^3$ . Je-li matice soustavy trojúhelníková (speciálně při zpětném chodu GEM a v řešení soustavy při známém LU-rozkladu), vychází složitost úměrná  $n^2$ . Rovněž vynásobení vektoru pravých stran známou inverzní maticí má složitost úměrnou  $n^2$ . Je-li matice soustavy diagonální, je složitost řešení úměrná  $n$ .

U iteračních metod má v obecném případě jeden krok složitost úměrnou  $n^2$ . Počet kroků závisí na požadované přesnosti a rychlosti konvergence (a ta na spektrálním poloměru iterační matice).

**Úloha:**

$$\mathbf{A} \vec{x} = \mathbf{B}$$

**Poznámka:** Matice  $\mathbf{B}$  zde reprezentuje pravé strany soustavy, nikoli iterační matici.

- $\mathbf{A}$  plná (málo nulových prvků),  $\mathbf{B}$  málo sloupců: GEM nebo její modifikace
- $\mathbf{A}$  plná,  $\mathbf{B}$  mnoho sloupců: výpočet  $\mathbf{A}^{-1}$  a řešení maticovým násobením nebo LU-rozklad
- $\mathbf{A}$  řídká (mnoho nulových prvků) nepravidelně: GEM s výběrem hlavního prvku
- $\mathbf{A}$  řídká pravidelně: iterační metody JIM, GSM, SOR

## Dodatek: Přehled značení

Popis je zjednodušený a nemusí být přesný, podrobnosti jsou v textu. Značení použité jen lokálně zde není uvedeno.

### Značení specifické pro tuto kapitolu

$n$  ... počet rovnic i neznámých

$\mathbb{R}^n$  ...  $n$ -rozměrný aritmetický vektorový prostor

$\mathbb{R}^{n,n}$  ... prostor čtvercových matic řádu  $n$

$\vec{0} \in \mathbb{R}^n$  ... nulový vektor

$\mathbf{O} \in \mathbb{R}^{n,n}$  ... nulová matice

$\mathbf{E} \in \mathbb{R}^{n,n}$  ... jednotková matice

$\mathbf{A}$  ... matice soustavy

$\vec{b}$  ... vektor pravých stran

$\vec{x}$  ... vektor řešení

$\vec{x}$  ... správné řešení

$\vec{x}^{(k)}$ ,  $k = 1, 2, \dots$  ... posloupnost přibližných řešení

$\vec{x}_c$  ... jakékoli přibližné řešení

$\vec{r}$  ... residuum přibližného řešení  $\vec{x}_c$

$\mathbf{L}, \mathbf{U}$  ... matice z LU-rozkladu (dolní a horní trojúhelníková)

$\mathbf{D}, \mathbf{L}, \mathbf{U}$  ... matice v iteračních metodách (diagonální, ostře dolní a ostře horní trojúhelníková)

$\mathbf{B}, \mathbf{B}_{\text{JIM}}, \mathbf{B}_{\text{GSM}}, \mathbf{B}_\omega$  ... iterační matice (obecná, Jacobiova, Gaussova-Seidelova, superrelaxační)

$\lambda_i$ ,  $i = 1, 2, \dots$  ... vlastní čísla iterační matice

$\varrho(\cdot)$  ... spektrální poloměr

$\vec{c}, \vec{c}_{\text{JIM}}, \vec{c}_{\text{GSM}}, \vec{c}_\omega$  ... konstantní vektor v iteračních metodách (obecná, Jacobiova, Gaussova-Seidelova, superrelaxační)

Vektorové normy:

$\|\cdot\|_v$  ... obecná

$\|\cdot\|_e$  ... euklidovská

$\|\cdot\|_r$  ... maximová, Čebyševova

$\|\cdot\|_s$  ... součtová, „manhattanská“

$\|\cdot\|_q$  ... společné zobecnění pro  $q \geq 1$

Maticové normy:

$\|\cdot\|_M$  ... obecná

$\|\cdot\|_F$  ... euklidovská, Frobeniova

$\|\cdot\|_R$  ... řádková

$\|\cdot\|_S$  ... sloupcová

### Značení používané podobně v celém předmětu

$\mathbb{R}$  ... množina všech reálných čísel

$\mathbb{C}$  ... množina všech komplexních čísel

## Literatura

- [Navara, Němeček] Navara, M., Němeček, A.: *Numerické metody*. ČVUT, Praha, dotisk 2005.
- [Knuth] Knuth, D.E.: *Fundamental Algorithms*. Vol. 1 of *The Art of Computer Programming*, 3rd ed., Addison-Wesley, Reading, MA, 1997.
- [Num. Recipes] Press, W.H., Teukolsky, S.A., Vetterling, W.T., Flannery, B.P.: *Numerical Recipes (The Art of Scientific Computing)*. 3rd edition, Cambridge University Press, Cambridge, 2007. <http://www.nrbook.com/a/bookcpdf.php>
- [Handbook Lin. Alg.] Hogben, L. (ed.): *Handbook of Linear Algebra*. Chapman & Hall/CRC, Boca Raton/London/New York, 2007.
- [Spielman, Teng] Spielman, D.A., Teng, S.H.: Nearly-Linear Time Algorithms for Preconditioning and Solving Symmetric, Diagonally Dominant Linear Systems. [arXiv:cs/0607105v5](https://arxiv.org/abs/cs/0607105v5), 2112. DOI 10.48550/arXiv.cs/0607105