Analýza maticových výpočtů

Petr Tichý

8. prosince 2025

Obsah

I Úplný problém nejmenších čtverců **5**

1 Formulace a řešení pomocí SVD **6**
1.1 Lineární aproximační problém 7
1.2 Úplný problém nejmenších čtverců (TLS) . 11
1.3 Předpoklady 12
1.4 Řešení pomocí SVD 15
1.5 Technická realizace 19
1.6 Podmínky existence řešení 24

2 Další metody a zobecnění **26**

2.1 Optimalizační TLS problém 27
2.2 Normální rovnice pro TLS 28
2.3 Scaled total least-squares 29
2.4 Negenerické řešení 30

II Metoda sdružených gradientů 32

3 Odvození CG 33
3.1 Kvadratický functionál 34
3.2 Optimum na afinním prostoru 35
3.3 A-skalární součin a norma 37
3.4 Optimální vlastnost 41
3.5 Výpočet A-ortogonální báze 42
3.6 Algoritmus metody CG 44
3.7 Praktické otázky 48

4 Konvergence a předpodmínění 50
4.1 CG a Krylovovy prostory 51
4.2 Konvergence CG 52
4.3 Předpodmínění 55

III Krylovovské metody 59

5 Projektivní metody 60
5.1 Projektivní metody 61
5.2 Krylovovské metody 63
5.3 Výpočet ortogonální báze 65
5.4 Metody nad ortogonální bází 67

5.5 Metody pro $A = A^*$ 68
5.6 Metody pro obecnou A 70
5.7 Neortogonální báze 74

6 Faber-Manteuffelova věta 76
6.1 Problém krátkých rekurencí 77
6.2 Nutné a postačující podmínky 78
6.3 Kdy je A normální stupně s? 79
6.4 Nestandardní skalární součin 80

IV Maticové funkce 82

7 Definice maticové funkce 83
7.1 Maticové polynomy 86
7.2 Definice $f(A)$ (Taylor) 87
7.3 Hermiteova interpolace 91
7.4 Definice $f(A)$ (Hermite) 93
7.5 Ekvivalence definic 94
7.6 Definice $f(A)$ (Cauchy) 95

8 Výpočet a aproximace 96
8.1 $f(A)$ a $f(A)v$ v aplikacích 97
8.2 Využití spektrální informace 98
8.3 Polynomiální a racionální $\approx$ 102
8.4 Maticové iterace 106
8.5 Odhad normy $f(A)$ 107
8.6 Aproximace $f(A)v$ 108

V FFT **112**

9 Rychlá Fourierova transformace **113**
9.1 Spojitá transformace 114
9.2 Diskrétní transformace 115
9.3 Aproximace periodické f 117
9.4 Algoritmus FFT 120

10 Maticové výpočty a FFT **123**
10.1 Cirkulantní matice 124
10.2 Toeplitz a Hankel matice 125
10.3 Struktura DFT matice 126
10.4 Fast Poisson solver 127

VI Analýza výpočtů **130**

11 Zpětná stabilita **131**
11.1 Reprezentace čísel v počítači 132
11.2 Počítačová aritmetika 134
11.3 Zpětná stabilita algoritmu 135
11.4 Výpočet skalárního součinu 137
11.5 Přesnost výpočtu 142

12 Analýza výpočtů **144**
12.1 Householderovy reflexe 145
12.2 Iterační metody 148
12.3 Finite precision CG 149
12.4 Lanczosova metoda 151

Část I

Úplný problém nejmenších čtverců

1

Formulace a řešení pomocí SVD

1.1 Lineární aproximační problém

Uvažujme problém

$$\underbrace{A}_{\text{model}} \quad \underbrace{x}_{\text{neznámá}} \quad \approx \quad \underbrace{b}_{\text{měření}}$$

kde $A \in \mathbb{R}^{n \times m}$, $b \in \mathbb{R}^n$.

- $n > m \rightarrow$ přeurčená úloha
  - řešení nemusí existovat
- $n < m \rightarrow$ nedourčená úloha
  - volné parametry, řešíme např. $\min \|z\|$: $Az = b$.

Pro zjednodušení výkladu $\rightarrow n \geq m$.

- Jak interpretovat $\approx$? Závisí na problému.

Problém nejmenších čtverců

V (přesných) časech t_i naměříme (přibližné) hodnoty b_i, $i = 1, \ldots, n$. Předpokládáme lineární závislost,

$$\alpha\, t = b.$$

Hledáme α tak, aby

$$\alpha\, t_i \approx b_i, \quad i = 1, \ldots, n,$$

maticově

$$\underbrace{\begin{bmatrix} t_1 \\ \vdots \\ t_n \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} \alpha \end{bmatrix}}_{x} \approx \underbrace{\begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix}}_{b}.$$

Mohu řešit ve smyslu nejmenších čtverců

$$\min_{\alpha} \sum_{i=1}^{n} (b_i - \alpha t_i)^2 = \min_{x} \|b - Ax\|^2.$$

Ekvivalentně hledáme nejmenší f tak, aby byl

$$Ax = b + f$$

kompatibilní. x^{LS} je pak řešením opraveného systému.

Předpoklad → model je přesný, hledáme opravu pravé strany. Obecně však i model bývá nepřesný.

Opakování LS

$A \in \mathbb{R}^{n\times m}$, $b \in \mathbb{R}^n$, $n \geq m$, $b \notin \mathcal{R}(A)$, problém

$$\min_{x\in\mathbb{R}^m} \| b - Ax \|^2.$$

Platí

- x je řeší LS $\Leftrightarrow$ $Ax = b|_{\mathcal{R}(A)}$.
- x je řeší LS $\Leftrightarrow$ $A^*Ax = A^*b$.
- Je-li $\text{rank}(A) = m$, pak je řešení jednoznačné.
- $\exists!$ řešení LS minimální v normě: $Ax = b|_{\mathcal{R}(A)}$ a

$$x \perp \mathcal{N}(A), \text{ tj. } x \in \mathcal{R}(A^*).$$

Ekvivalentně, hledáme nejmenší opravu

$$\min_{f\in\mathbb{R}^m} \{\| f \|^2 : b + f \in \mathcal{R}(A)\}.$$

Taková oprava vždy existuje a je určena jednoznažně,

$$f = b|_{\mathcal{R}(A)} - b.$$

Ortogonální regrese

- V (přibližných) časech t_i naměříme (přibližné) hodnoty b_i, $i = 1, \ldots, n$, lineární závislost dat.
- Nechť je dána nějaká přímka, označme

  $$(t_i + \delta_i, b_i + f_i)$$

  nejbližší bod na přímce k (t_i, f_i).
- Kvadrát vzdálenosti je dán

  $$\delta_i^2 + f_i^2.$$

- Chceme určit přímku tak, aby byl součet kvadrátů vzdáleností daných bodů od přímky, tj.

  $$\sum_{i=1}^{n} \delta_i^2 + \sum_{i=1}^{n} f_i^2,$$

  minimální.
- Označme $E = [\delta_1, \ldots, \delta_n]^T$ a $f = [f_1, \ldots, f_n]^T$.
- Body $(t_i + \delta_i, b_i + f_i)$ leží na přímce, tedy systém

  $$(A + E)x = b + f$$

  je kompatibilní.
- Chceme řešit úlohu

  $$\min_{f,E} \|[f, E]\|_F : \ (A + E)\, x = b + f.$$

1.2 Úplný problém nejmenších čtverců (TLS)

- Uvažujme lineární aproximační problém

$$Ax \approx b, \quad b \in \mathbb{R}^n, \quad A \in \mathbb{R}^{n \times m}, \quad n \geq m.$$

  Předpoklad chyby v b (pozorování) i v A (model).

- Hledáme co nejmenší změny $E \in \mathbb{R}^{n \times m}$ a $f \in \mathbb{R}^n$ tak, že je systém

$$(A + E)\, x = b + f$$

  kompatibilní. Velikost změn měříme pomocí

$$\|[f, E]\|_F .$$

- Úplný problém nejmenších čtverců:

$$\min_{f,E} \|[f, E]\|_F : \ (A + E)\, x = b + f,$$

  Řešení (pokud existuje) označme x^{TLS}.

- TLS (Total least squares), ortogonální regrese.

1.3 Předpoklady

- Je-li $Ax = b$ kompatilní, $b \in \mathcal{R}(A)$, pak $E = 0$, $f = 0$ a TLS formulace není potřeba $\rightarrow$ řešíme $Ax = b$.
- V dalším předpoklad

$$b \notin \mathcal{R}(A).$$

- Jsou-li sloupce A lineárně závislé, model obsahuje redundantní informaci $\rightarrow$ původní úlohu lze přeformulovat (například vybereme jen lineárně nezávislé sloupce a položíme $x_i = 0$ pro nevybrané sloupce). Úlohu lze formulovat „lépe“.
- V dalším předpoklad

$$\text{rank}(A) = m.$$

- Následující lemma a věta ukazují, že pokud $b \notin \mathcal{R}(A)$ a $\text{rank}(A) < m$, pak TLS nemá řešení.

Lemma 1.1 (TLS a neplná sloupcová hodnost)**.** *Nechť jsou dány* $A \in \mathbb{R}^{n\times m}$, $n \geq m$, $b \in \mathbb{R}^n$, $b \neq 0$, *a nechť* $\operatorname{rank}(A) < m$. *Potom* $\forall \varepsilon > 0\ \exists\, [f, E] \in \mathbb{R}^{n\times(m+1)}$:

$$\|[f, E]\|_F = \varepsilon\|b\| \quad a \quad (b + f) \in \mathcal{R}(A + E)\,.$$

Důkaz. (dobrovolný)

- Položme $f = 0$ a uvažujme singulární rozklad A,

$$A = USV^T, \quad U \in \mathbb{R}^{n\times n},\ S \in \mathbb{R}^{n\times m},\ V \in \mathbb{R}^{m\times m}.$$

- $\operatorname{rank}(A) < m \Rightarrow \exists j$: j-tý sloupec S je nulový.
- Nechť $V = [v_1, \ldots, v_m]$ a položme $E = \varepsilon b v_j^T$, platí

$$\|[f, E]\|_F = \|E\|_F = \varepsilon\|b\|.$$

- Nechť $e_j \in \mathbb{R}^m$ je j-tý jednotkový vektor, pak

$$\begin{aligned} U^T(A + E)Ve_j &= U^T(USV^T + \varepsilon b v_j^T)v_j \\ &= Se_j + \varepsilon U^T b = \varepsilon U^T b. \end{aligned}$$

- b je tedy násobkem j-sloupce $(A + E)V$, platí

$$b \in \mathcal{R}((A + E)V) = \mathcal{R}(A + E).$$

- $\|[f, E]\|_F = \varepsilon\|b\|$ a $(A+E)x = b+f$ je kompatibilní. □

Věta 1.2. *Pokud $b \notin \mathcal{R}(A)$ a $\operatorname{rank}(A) < m$, pak TLS problém nemá řešení.*

Důkaz. Předpokládejme, že TLS má řešení a označme δ minimální $\|[f, E]\|_F$ vedoucí na kompatibilní systém.

- Z $b \notin \mathcal{R}(A)$ plyne, $\delta > 0$.
- Z předchozího lemmatu $\exists \varepsilon > 0 : \|[f_\varepsilon, E_\varepsilon]\|_F = \varepsilon \|b\| < \delta$ a $(A + E_\varepsilon)x = b + f_\varepsilon$ je kompatibilní.
- Spor s minimalitou δ. □

Příklad 1.3. Uvažujme TLS problém

$$\begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \approx \begin{bmatrix} 1 \\ 1 \end{bmatrix}.$$

Definujme

$$f_\varepsilon = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad \text{a} \quad E_\varepsilon = \begin{bmatrix} 0 & 0 \\ 0 & \varepsilon \end{bmatrix}.$$

Potom je perturbovaný systém kompatibilní,

$$\begin{bmatrix} 1 & 0 \\ 0 & \varepsilon \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \qquad x_\varepsilon = \begin{bmatrix} 1 \\ 1/\varepsilon \end{bmatrix}, .$$

Vidíme

$$\|[f_\varepsilon, E_\varepsilon]\|_F \to 0, \qquad \|x_\varepsilon\| \to \infty,$$

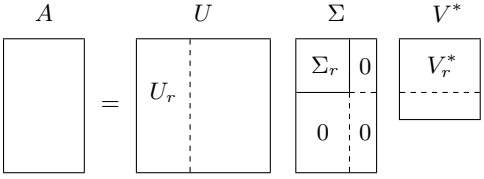
minimální oprava neexistuje $\Rightarrow$ TLS řešení neexistuje.

TLS - vyrazně komplikovanější než LS
(analýza i výpočet - pomocí SVD)

1.4 Řešení pomocí SVD

Opakování singulárního rozkladu (SVD)

$A \in \mathbb{R}^{n\times m}$, $\operatorname{rank}(A) = r$



Zápisy SVD

$$A = \underbrace{U\Sigma V^*}_{\text{plný}} = \underbrace{U_r\,\Sigma_r\,V_r^*}_{\text{ekonomický}} = \underbrace{\sum_{j=1}^{r} \sigma_j\, u_j\, v_j^*}_{\text{dyadický}},$$

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0.$$

Úloha aproximace matice

Pro $A \in \mathbb{R}^{n\times m}$ a přirozené $k < r$ najdi nejlepší aproximaci A hodnosti k:

$$\min_{\substack{X\in\mathbb{R}^{n\times m}\\ \mathrm{rank}(X)=k}} \|A - X\|_\star \,,$$

kde $\star = 2$ nebo $\star = F$.

Věta 1.4 (Eckart, Young, Mirsky)**.** *$A \in \mathbb{R}^{n\times m}$ hodnosti r,*

$$A = \sum_{j=1}^{r} \sigma_j \, u_j \, v_j^* \,.$$

Potom je pro $k < r$ matice

$$A^{(k)} \equiv \sum_{j=1}^{k} \sigma_j \, u_j \, v_j^*$$

nejlepší aproximací A hodnosti k a platí

$$\begin{aligned} \|A - A^{(k)}\|_2 &= \sigma_{k+1}, \\ \|A - A^{(k)}\|_F &= \sqrt{\sigma_{k+1}^2 + \cdots + \sigma_r^2}. \end{aligned}$$

Minimální oprava snižující hodnost

- Dána $B \in \mathbb{R}^{n\times(m+1)}$ hodnosti $m+1$. Hledáme minimální opravu Δ tak, aby

$$B + \Delta = X \quad \text{a} \quad \operatorname{rank}(X) \leq m.$$

  Velikost Δ měříme pomocí $\|\Delta\|_F$.

- Řešíme

$$\min_{\substack{X\in\mathbb{R}^{n\times m}\\ \operatorname{rank}(X)\leq m}} \|B - X\|_F \;=\; \min_{\substack{X\in\mathbb{R}^{n\times m}\\ \operatorname{rank}(X)= m}} \|B - X\|_F$$

- Minimální oprava snižující hodnost B

$$B = \underbrace{\sigma_1\, u_1\, v_1^* + \cdots + \sigma_m\, u_m\, v_m^*}_{\text{nejlepší rank-}m\text{ aproximace}} + \underbrace{\sigma_{m+1}\, u_{m+1}\, v_{m+1}^*}_{-\Delta}.$$

Řešení TLS pomocí SVD

Předpoklad $b \notin \mathcal{R}(A)$ a $\operatorname{rank}(A) = m$, řešíme

$$\min_{f,E} \|[f,E]\|_F : \ (A+E)\,x = b + f.$$

Platí

$$(A+E)\,x = b+f \quad \Leftrightarrow \quad \left(\underbrace{[f,E]}_{\text{oprava}} + \underbrace{[b,A]}_{\text{data}} \right) \begin{bmatrix} -1 \\ x \end{bmatrix} = 0\,,$$

$\Rightarrow$ oprava $[f,E]$ musí snižovat hodnost $[b,A]$.

Dva kroky k řešení

1. Hledáme minimální opravu $[f,E]$ tak, aby byl nulový prostor $\mathcal{N}([f,E]+[b,A])$ netriviální.
2. Pokud $\exists z \in \mathcal{N}([f,E]+[b,A]) : \ z_1 \neq 0$, pak

$$-\frac{1}{z_1} z = - \begin{bmatrix} 1 \\ \frac{z_2}{z_1} \\ \vdots \\ \frac{z_{m+1}}{z_1} \end{bmatrix} = \begin{bmatrix} -1 \\ x^{\mathrm{TLS}} \end{bmatrix},$$

tj.

$$x^{\mathrm{TLS}} = -\frac{1}{z_1} \begin{bmatrix} z_2 \\ \vdots \\ z_{m+1} \end{bmatrix}.$$

1.5 Technická realizace

Minimální oprava $[f, E]$ snižující hodnost $[b, A]$:

$$[b, A] = \underbrace{\sigma_1 u_1 v_1^* + \cdots + \sigma_m u_m v_m^*}_{\text{nejlepší rank-}m\text{ aproximace}} + \underbrace{\sigma_{m+1} u_{m+1} v_{m+1}^*}_{-[f,E]}$$

Jednoduché σ_{m+1}

$$\sigma_1 \geq \cdots \geq \sigma_m > \sigma_{m+1}$$

- Matice nejmenší normy snižující hodnost $[b, A]$ je

  $$[f, E] = -\sigma_{m+1} u_{m+1} v_{m+1}^*$$

  a je jednoznačná, $\|[f, E]\|_F = \sigma_{m+1}$.

- Nulový prostor

  $$\mathcal{N}([b, A] + [f, E]) = \operatorname{span}\{v_{m+1}\}.$$

- Je-li $e_1^T v_{m+1} \neq 0$, pak

  $$v_{m+1} = \begin{bmatrix} \nu \\ w \end{bmatrix} \quad \Rightarrow \quad x^{\mathrm{TLS}} = -\frac{1}{\nu} w.$$

Interpretace následující věty: Pokud řešení TLS existuje, najdeme ho pomocí kroků 1 a 2.

Věta 1.5. *$A \in \mathbb{R}^{n \times m}$, $\operatorname{rank}(A) = m$, $b \notin \mathcal{R}(A)$. Uvažujme SVD matice $[b, A]$, nechť $\sigma_m > \sigma_{m+1}$ a nechť $e_1^T v_{m+1} = 0$. Potom TLS řešení neexistuje.*

Důkaz. (dobrovolný) Velikost opravy $\|[f, E]\|_F = \sigma_{m+1}$ nevede na kompatibilní systém. Pro $0 < \varepsilon < 1$ položme

$$v = \varepsilon v_k + \sqrt{1-\varepsilon^2}\, v_{m+1},$$

kde v_k splňuje $e_1^T v_k \neq 0 \Rightarrow e_1^T v \neq 0$, a definujme

$$\begin{aligned}[f_\varepsilon, E_\varepsilon] &\equiv -[b, A]vv^* \\ &= -\underbrace{\left(\varepsilon\sigma_k u_k + \sqrt{1-\varepsilon^2}\sigma_{m+1}u_{m+1}\right)}_{u} v^*.\end{aligned}$$

Stopa matice $[f_\varepsilon, E_\varepsilon]^*[f_\varepsilon, E_\varepsilon] = \|u\|^2 vv^*$ dává

$$\|[f_\varepsilon, E_\varepsilon]\|_F = \|u\| = \sqrt{\varepsilon^2\sigma_k^2 + (1-\varepsilon^2)\sigma_{m+1}^2}.$$

Zároveň platí

$$[b, A] + [f_\varepsilon, E_\varepsilon] = [b, A](I - vv^*),$$

tj.

$$[b + f_\varepsilon, A + E_\varepsilon]v = 0.$$

Jelikož je $e_1^T v \neq 0$, $\left|e_1^T v\right| < \varepsilon$, $\|v\| = 1$, platí

$$b + f_\varepsilon \in \mathcal{R}(A + E_\varepsilon)$$

a opravený systém je kompatibilní, řešení

$$\begin{bmatrix} -1 \\ x_\varepsilon \end{bmatrix} = -\frac{v}{\varepsilon\left(e_1^T v_k\right)}.$$

Očividně, pro $\varepsilon \to 0$ je

$$\|x_\varepsilon\| \to \infty, \quad \|[f_\varepsilon, E_\varepsilon]\|_F \to \sigma_{m+1}.$$

Viz též [14, s. 68]. □

Násobné σ_{m+1}

$$\sigma_1 \geq \cdots \geq \sigma_\ell > \sigma_{\ell+1} = \cdots = \sigma_{m+1}, \quad 1 \leq \ell < m.$$

Definujme

$$\mathcal{V}_{\min} \equiv \operatorname{span}\{v_{\ell+1}, \ldots, v_{m+1}\}.$$

- Je-li $e_1^T v = 0 \ \forall v \in \mathcal{V}_{\min}$, lze dokázat, že řešení neexistuje.
- Jinak $\exists v \in \mathcal{V}_{\min} : e_1^T v \neq 0$ a lze jej použít ke konstrukci x^{TLS}
  - existuje nekonečně mnoho řešení
  - existuje právě jedno minimální v normě
- Hledáme $v \in \mathcal{V}_{\min}$ tak, že příslušné TLS řešení má minimální normu.
- $v,\ w \in \mathcal{V}_{\min}$, $\|v\| = \|w\| = 1$, $v_1 \neq 0 \neq w_1$, pak

$$|w_1| < |v_1| \ \Leftrightarrow \ \left\| \frac{1}{v_1} \begin{bmatrix} v_2 \\ \vdots \\ v_{m+1} \end{bmatrix} \right\| < \left\| \frac{1}{w_1} \begin{bmatrix} w_2 \\ \vdots \\ w_{m+1} \end{bmatrix} \right\|.$$

- Hledáme $v \in \mathcal{V}_{\min}$: $\|v\| = 1$ a $|e_1^T v|$ je maximální.

Algoritmus

- Připomeňme Householderovy reflexe
  - ortogonální, hermitovské
$$H_q = I - 2qq^*$$
  - nemění normu vektoru, nulování složek w
$$H_q w = \|w\| e_1 \quad \Leftrightarrow \quad w^* H_q = \|w\| e_1^T$$
    pomocí
$$q = \frac{w - \|w\| e_1}{\|w - \|w\| e_1\|}.$$
  - k vynulování lze použít i vektor $-\|w\| e_1$.
- Označme
$$V_{\min} = [v_{\ell+1}, \ldots, v_{m+1}].$$
- Je-li H libovolná reflexe, pak $V_{\min} H$ nemění normy řádků, ortonormální sloupce $V_{\min} H$ leží ve $\mathcal{V}_{\min}$.
- Označme první řádek $V_{\min}$ jako w^* a určeme H tak, aby nulovala složky w. Potom má vektor
$$v = (V_{\min} H)\, e_1$$
  maximální první složku (v absolutní hodnotě).

Věta 1.6. *$A \in \mathbb{R}^{n\times m}$, $\operatorname{rank}(A) = m$, $b \notin \mathcal{R}(A)$. Má-li TLS problém řešení, pak je řešení minimální v normě jednoznačné.*

Důkaz. (dobrovolný)

• Existují-li dvě TLS řešení minimální v normě, pak $\exists$ dva lineárně nezávislé $v, w \in \mathcal{V}_{\min}$:

$$\|v\| = 1 = \|w\|, \quad |e_1^T v| = |e_1^T w|.$$

• Uvažujme matici $[v, \widetilde{w}]$, jejíž sloupce tvoří ortonormální bázi $\operatorname{span}\{v, w\}$,

$$\widetilde{w} = \frac{w - (v^* w)\, v}{\|w - (v^* w)\, v\|}.$$

• Z lineární nezávislosti plyne $|v^* w| < 1$ a první složka $\widetilde{w}$ je tudíž nenulová.

• Pomocí předchozího (aplikací HH reflexe zprava na $[v, \widetilde{w}]$) umíme sestrojit vektor $z \in \operatorname{span}\{v, w\}$ mající první složku větší než $|e_1^T v|$. □

1.6 Podmínky existence řešení

Uvažujme ekonomický SVD, $r = m + 1$,

$$[b, A] = U_r \Sigma_r V_r^* \quad \Leftrightarrow \quad \begin{bmatrix} b^* \\ A^* \end{bmatrix} = V_r \Sigma_r U_r^*.$$

Násobením U_r zprava dostaneme

$$\begin{bmatrix} b^* \\ A^* \end{bmatrix} U_r = V_r \Sigma_r.$$

Pro nejmenší singulární číslo σ_{m+1} je

$$\begin{bmatrix} b^* \\ A^* \end{bmatrix} [u_{\ell+1}, \ldots, u_{m+1}] = \sigma_{m+1} [v_{\ell+1}, \ldots, v_{m+1}].$$

První složky $v_{\ell+1}, \ldots, v_{m+1}$ jsou nulové $\Leftrightarrow$

$$b \perp \operatorname{span}\{u_{\ell+1}, \ldots, u_{m+1}\} \equiv \mathcal{U}_{\min}.$$

Nutná a postačující podmínka

Věta 1.7 (Existence TLS řešení)**.** *$A \in \mathbb{R}^{n \times m}$,* $\operatorname{rank}(A) = m$*, $b \notin \mathcal{R}(A)$. Potom TLS řešení existuje* $\Leftrightarrow$

$$b \not\perp \mathcal{U}_{\min}([b, A]).$$

Důkaz. Pokud $b \not\perp \mathcal{U}_{\min}$, pak je některá z prvních složek $v_{\ell+1}, \ldots, v_{m+1}$ nenulová a řešení lze sestrojit dle předchozího. Pokud je $b \perp \mathcal{U}_{\min}$, pak lze podobně jako v případě jednonásobného σ_{m+1} dokázat neexistenci nejmenší opravy vedoucí na kompatibilní systém. □

Golub–Van Loanova podmínka (postačující)

Věta 1.8 (Golub, Van Loan)**.** *Dáno $A \in \mathbb{R}^{n\times m}$, $b \in \mathbb{R}^n$, $n \geq m$. Nechť $b \notin \mathcal{R}(A)$ a $\operatorname{rank}(A) = m$. Pokud platí*

$$\sigma_{\min}(A) > \sigma_{\min}([b, A]), \tag{1.1}$$

pak je $\sigma_{\min}([b, A])$ jednonásobné a $e_1^T v_{m+1} \neq 0$, tj. existuje právě jedno TLS řešení.

Důkaz. $\forall v \in \mathcal{V}_{min}$ je vlastním vektorem $[b, A]^*[b, A]$,

$$\begin{bmatrix} b^*b & b^*A \\ A^*b & A^*A \end{bmatrix} v = \sigma_{m+1}^2 v.$$

Kdyby $e_1^T v = 0$, pak je σ_{m+1} singulárním číslem A, což je ve sporu s (1.1). Libovolný $v \in \mathcal{V}_{min}$ má tedy první složku nenulovou.

Pokud $\dim(\mathcal{V}_{min}) > 1$, pak $\exists v \in \mathcal{V}_{min} : e_1^T v = 0$. □

2

Další metody a zobecnění

2.1 Optimalizační TLS problém

Je-li splněna Golub–Van Loanova podmínka (1.1), je možné řešit TLS pomocí optimalizačních metod. Důležité v případech, kdy nelze použít SVD.

Připomeňme, že pokud je S hermitovská, pak platí

$$\lambda_{\min}(S) = \min_{v \neq 0} \frac{v^* S v}{v^* v}.$$

Je-li splněna podmínka (1.1), pak

$$\begin{aligned} \|[f, E]\|_F^2 &= \sigma_{\min}^2([b, A]) = \lambda_{\min}([b, A]^T [b, A]) \\ &= \min_{x \in \mathbb{R}^m} \frac{\begin{bmatrix} -1 \\ x \end{bmatrix}^* [b, A]^* [b, A] \begin{bmatrix} -1 \\ x \end{bmatrix}}{\begin{bmatrix} -1 \\ x \end{bmatrix}^* \begin{bmatrix} -1 \\ x \end{bmatrix}} \\ &= \min_{x \in \mathbb{R}^m} \frac{\|b - Ax\|^2}{1 + \|x\|^2}. \end{aligned}$$

K řešení lze použít metody pro minimalizaci funkcionálu

$$\begin{aligned} f(x) &= \frac{\|b - Ax\|^2}{1 + \|x\|^2}, \\ \nabla f(x) &= \frac{2A^T(Ax - b)}{1 + \|x\|^2} - \frac{2\|b - Ax\|^2}{(1 + \|x\|^2)^2}\, x\,. \end{aligned}$$

2.2 Normální rovnice pro TLS

Nechť existuje TLS řešení. Pak je vektor

$$\begin{bmatrix} -1 \\ x^{TLS} \end{bmatrix}$$

nenulovým násobkem nějakého $v \in \mathcal{V}_{min}$ a splňuje

$$\begin{bmatrix} b^*b & b^*A \\ A^*b & A^*A \end{bmatrix} \begin{bmatrix} -1 \\ x^{TLS} \end{bmatrix} = \sigma_{m+1}^2 \begin{bmatrix} -1 \\ x^{TLS} \end{bmatrix}.$$

Platí tedy

$$-A^*b + A^*Ax^{TLS} = \sigma_{m+1}^2 x^{TLS}$$

neboli

$$\left(A^*A - \sigma_{m+1}^2 I\right) x^{TLS} = A^*b.$$

Je-li splněna Golub–Van Loanova podmínka (1.1), pak je $A^*A - \sigma_{m+1}^2$ regulární a soustava výše má jednoznačné řešení. Problémy s podmíněností.

2.3 Scaled total least-squares

- Chyby jen v A - data least-squares (DLS)

$$\min_{E} \|E\|_F : \ (A+E)\,x = b.$$

- Chyby v A i b různé - scaled TLS pro $\gamma > 0$:

$$\min_{f,E} \|[\gamma f, E]\|_F : \ (A+E)\,x = b+f.$$

- $\gamma \to 0$ upřednostňuji perturbace v b, vede na LS

$$\min_{f,\widetilde{E}} \left\|[f, \widetilde{E}]\right\|_F : \ (A+\gamma\widetilde{E})\,x = b+f.$$

- $\gamma \to \infty$ upřednostňuji E, vede na DLS,

$$\min_{\widetilde{f},E} \left\|[\widetilde{f}, E]\right\|_F : \ (A+E)\,x = b+\widetilde{f}/\gamma.$$

Řešení

$$\left(\underbrace{[\widetilde{f}, E]}_{\text{oprava}} + \underbrace{[\gamma b, A]}_{\text{data}}\right) \begin{bmatrix} -\gamma^{-1} \\ x \end{bmatrix} = 0\,.$$

2.4 Negenerické řešení

Náhradní přístup (i pro $\text{rank}(A) < m$), neexistuje-li řešení

$$\sigma_k > \underbrace{\sigma_{k+1} = \cdots = \sigma_\ell}_{\text{zkusím analogicky}} > \underbrace{\sigma_{\ell+1} = \cdots = \sigma_{m+1}}_{\text{nelze použít}}.$$

Nevede-li minimální korekce k TLS řešení ($e_1^T v = 0$ $\forall v \in \mathcal{V}_{min}$), lze zkusit hledat korekci vzhledem k druhému nejmenšímu singulárnímu číslu σ_ℓ ve tvaru

$$\sigma_\ell u v^*,$$

kde v leží v pravém singulárním prostoru příslušnému k σ_ℓ a u je korespondující levý singulární vektor

$$[b, A]v = \sigma_\ell u.$$

Nyní ale řešíme TLS s omezením

$$\min_{E,f} \|[f, E]\|_F : \ (A + E)\, x = b + f,$$

za podmínky

$$[f, E]v = 0 \ \forall v \in \mathcal{V}_{min}.$$

Algoritmus

- vstup $[b, A] = U_r \Sigma_r V_r^*$ ekonomický, $r = \operatorname{rank}([b, A])$
- $k = r$
  1. Urči index ℓ tak, že
     $$\sigma_\ell > \sigma_{\ell+1} = \cdots = \sigma_k$$
     Pokud takové ℓ neexistuje, definuj $\ell = 0$.
  2. Najdi pravý singulární prostor k σ_k a sestav
     $$V = [v_{\ell+1}, \ldots, v_k].$$
  3. Pokud $e_1^T V \neq 0$, najdi řešení s min. normou
     $$VH = \begin{bmatrix} \nu & 0^T \\ z & X \end{bmatrix}, \quad x^{\mathrm{NG}} = -\frac{1}{\nu} z.$$
  4. Pokud $e_1^T V = 0$, polož $k = \ell$ a jdi na 1.
- výstup x^{NG} je x^{TLS} nebo negenerické

Není potřeba celé SVD, stačí triplety od nejmenších singulárních čísel. Vždy vrací výstup, náročnost $\mathcal{O}(n^3)$.

Část II

Metoda sdružených gradientů

3 Odvození CG

- Systém lineárních rovnic

$$Ax = b,$$

  $A \in \mathbb{R}^{n \times n}$ je symetrická, pozitivně definitní (SPD).

- Do roku 1952 jen dva typy metod
  - přímé (Gaussova eliminace)
  - stacionární.
- Hledala se ideální metoda, která
  - poskytuje aproximaci v každé iteraci,
  - nalezne řešení v konečném počtu kroků.

3.1 Kvadratický functionál

- Data úlohy: $A \in \mathbb{R}^{n\times n}$ SPD a $b \in \mathbb{R}^n$.
- A a b definují **kvadratický** funkcionál $f : \mathbb{R}^n \mapsto \mathbb{R}$

$$f(x) \equiv \frac{1}{2} x^T A x - x^T b.$$

- Pomocí kalkulu funkcí více proměnných,

$$\nabla f(x) = Ax - b, \quad \nabla^2 f(x) = A.$$

- **Hessova matice** je SPD a f je ryze konvexní.
- Jediného (lokálního i globálního) minima nabývá f ve stacionárnim bodě x_*,

$$0 = \nabla f(x_*) = Ax_* - b \quad \Leftrightarrow \quad Ax_* = b.$$

- Systém lineárních rovnic lze identifikovat s podmínkou stacionárního bodu $\nabla f(x) = 0$.
- Numerickou metodu pro řešení $Ax = b$ lze odvodit jako optimalizační metodu pro minimalizaci $f(x)$.

3.2 Optimum na afinním prostoru

Lemma 3.1 (minimalizace přes afinní prostor)**.** *Uvažujme SPD $A \in \mathbb{R}^{n\times n}$ a $b \in \mathbb{R}^n$. Nechť je dán afinní prostor*

$$v + \mathcal{S}, \quad v \in \mathbb{R}^n, \quad \mathcal{S} \subseteq \mathbb{R}^n,$$

kde $\mathcal{S}$ dimenze k je reprezentovaný bází $\{p_0, \ldots, p_{k-1}\}$, $1 \le k \le n$. Potom lze jednoznačný minimizér funkcionálu f přes $v + \mathcal{S}$ vyjádřit ve tvaru

$$v + P\alpha_*, \quad P = [p_0, \ldots, p_{k-1}] \in \mathbb{R}^{n\times k},$$

kde α_* je řešením soustavy lineárních rovnic

$$P^T A P \alpha = P^T (b - Av).$$

Důkaz. Minimizér ve tvaru $v + P\alpha$ určíme minimalizací

$$\phi(\alpha) = f(v + P\alpha), \quad \alpha \in \mathbb{R}^k.$$

$\phi(\alpha)$ je opět kvadratický funkcionál

$$\begin{aligned}\phi(\alpha) &= \frac{1}{2}(v + P\alpha)^T A(v + P\alpha) - (v + P\alpha)^T b \\ &= \frac{1}{2}\alpha^T P^T A P \alpha + (P\alpha)^T (Av - b) + \frac{1}{2} v^T A v - v^T b\end{aligned}$$

s gradientem a SPD Hessovou maticí

$$\nabla\phi(\alpha) = P^T A P \alpha + P^T (Av - b), \quad \nabla^2\phi(\alpha) = P^T A P.$$

$\exists!$ optimální α_*, které je řešením $\nabla\phi(\alpha) = 0$. □

Speciální případy

- $\dim(\mathcal{S}) = 1$, $\mathcal{S} = \text{span}\{p\}$, minimizér

$$v + \alpha_* p\,, \quad \alpha_* = \frac{p^T(b - Av)}{p^T A p}\,.$$

- $\dim(\mathcal{S}) = n$, $\mathcal{S} = \text{span}\{p_0, \ldots, p_{n-1}\}$, minimizér

$$\begin{aligned} v + P\alpha_* &= v + P\left(P^T A P\right)^{-1} P^T(b - Av) \\ &= A^{-1}b = x_*. \end{aligned}$$

Pozorování

Je-li $P^T A P$ diagonální, minimizér lze spočíst pomocí

$$v + \sum_{i=0}^{k-1} \alpha_{*,i}, p_i, \quad \alpha_{*,i} = \frac{p_i^T(b - Av)}{p_i^T A p_i}.$$

3.3 A-skalární součin a norma

- $A \in \mathbb{R}^{n \times n}$ symetrická, pozitivně definitní (SPD).
- Pomocí A lze definovat skalární součin a normu

$$\langle x, y \rangle_A \equiv y^T A x, \quad \|x\|_A \equiv \langle x, x \rangle_A^{1/2}.$$

- Je skalární součin a norma dle definice.
- Označme $\|\cdot\|$ Euklidovskou normu vektoru. Platí

$$\|x\|_A = \|A^{1/2} x\|,$$

  kde $A^{1/2} = U \Lambda^{1/2} U^T$ (použij spektrální rozklad).
- Množina vektorů $\|x\|_A = 1$, je **elipsoid** s osami danými vlastními vektory A a délkou poloos $\lambda_i^{-1/2}$. Nakresli 2D obrázek.
- $\langle x, y \rangle_A = 0 \rightarrow x$ a y jsou A-ortogonální (sdružené)
- $P^T A P$ diagonální znamená, že báze $\{p_0, \ldots, p_{k-1}\}$ je A-ortogonální.

Myšlenka metody sdružených směrů

- Dána počáteční aproximace x_0.
- Generuj A-ortogonální báze prostorů $\mathcal{S}_k$ dimenze k,

$$\mathcal{S}_1 \subset \mathcal{S}_2 \subset \cdots \subset \mathcal{S}_k \subset \mathcal{S}_{k+1} \ldots$$

  a počítej minimizéry $x_k \in x_0 + \mathcal{S}_k$.

- Předpokládejme

$$\begin{aligned} \mathcal{S}_k &= \operatorname{span}\{p_0, \ldots, p_{k-1}\}, \\ \mathcal{S}_{k+1} &= \operatorname{span}\{p_0, \ldots, p_{k-1}, p_k\}. \end{aligned}$$

- Označíme-li $r_0 = b - Ax_0$, platí dle předchozího

$$\begin{aligned} x_{k+1} &= x_0 + \sum_{i=0}^{k} \left(\frac{p_i^T r_0}{p_i^T A p_i} \right) p_i \\ &= x_k + \left(\frac{p_k^T r_0}{p_k^T A p_k} \right) p_k. \end{aligned}$$

- Tak získáme řešení x_* nejvýše po n iteracích.

Věta 3.2 (o metodě sdružených směrů)**.** *$A \in \mathbb{R}^{n\times n}$ SPD, $b \in \mathbb{R}^n$, $x_0 \in \mathbb{R}^n$ počáteční přiblížení. Nechť $\{p_i\}_{i=0}^{n-1}$ tvoří A-ortogonální bázi $\mathbb{R}^n$ a označme*

$$\mathcal{S}_k \equiv \operatorname{span}\{p_0, \ldots, p_{k-1}\}.$$

Potom lze minimizéry x_k přes $x_0 + \mathcal{S}_k$ počítat rekurencí

$$x_{j+1} = x_j + \gamma_j p_j, \quad \gamma_j = \frac{p_j^T r_j}{p_j^T A p_j}, \tag{3.1}$$

kde $r_j = b - Ax_j$.

Důkaz. Z předchozího lze minimizéry počítat rekurencí

$$x_{j+1} = x_j + \left(\frac{p_j^T r_0}{p_j^T A p_j}\right) p_j.$$

Stačí tedy ukázat, že $p_j^T r_0 = p_j^T r_j$.

Uvědomme si, že $x_j - x_0 \in \mathcal{S}_j$ a $p_j \perp_A \mathcal{S}_j$. Platí

$$\begin{aligned} p_j^T r_j &= p_j^T (b - Ax_j + Ax_0 - Ax_0) \\ &= p_j^T (b - Ax_0) + p_j^T A(x_j - x_0) \\ &= p_j^T r_0. \end{aligned}$$

□

Pozorování

- Volbou γ_j provádíme 1D minimalizaci na přímce

$$x_j + \alpha p_j,$$

  viz speciální případ $k = 1$.

- Vztah pro reziduum, $x_{k+1} = x_k + \gamma_k\, p_k \Rightarrow$

$$r_{k+1} = r_k - \gamma_k\, A\, p_k$$

  a díky volbě γ_k platí

$$p_k^T r_{k+1} = p_k^T r_k - \gamma_k\, p_k^T A p_k = 0.$$

3.4 Optimální vlastnost

Metoda bude generovat aproximace x_k s vlastností

$$f(x_k) = \min_{x \in x_0 + \mathcal{S}_k} f(x).$$

Platí

$$\begin{aligned} f(x) &= \frac{1}{2} x^T A x - x^T b \\ &= \frac{1}{2}(x_* - x)^T A (x_* - x) - \frac{1}{2} x_*^T A x_* \\ &= \frac{1}{2}\|x_* - x\|_A^2 - \frac{1}{2}\|x_*\|_A^2 \\ &= \frac{1}{2}\|x_* - x\|_A^2 + f(x_*)\,. \end{aligned}$$

Nutně tedy

$$\|x_* - x_k\|_A = \min_{x \in x_0 + \mathcal{S}_k} \|x_* - x\|_A.$$

Z optimality a z

$$\underbrace{x_k - x_0}_{\in \mathcal{S}_k} + x_* - x_k \;=\; \underbrace{x_* - x_0}_{\text{daný vektor}}$$

plyne ortogonalita (obrázek)

$$x_* - x_k \perp_A \mathcal{S}_k \quad \Leftrightarrow \quad r_k \perp \mathcal{S}_k.$$

3.5 Výpočet A-ortogonální báze

- Jak generovat A-ortogonální bázi?
- Iterace k: Máme sdružené směry $p_0, \ldots, p_k \neq 0$, aproximace $x_0, \ldots, x_k$ a rezidua $r_0, \ldots, r_k \neq 0$. Umíme sestrojit x_{k+1} a r_{k+1}. Jak rozšiřovat prostor směrových vektorů, jak volit p_{k+1}?
- $r_{k+1} = 0 \Rightarrow x_{k+1}$ je řešení. Nechť $r_{k+1} \neq 0$.
- Z předchozího,

  $$r_{k+1} \perp \mathcal{S}_{k+1} = \operatorname{span}\{p_0, \ldots, p_k\}\,.$$

  a $r_{k+1}, p_0, \ldots, p_k$ jsou tudíž lineárně nezávislé.
- K rozšíření použijeme reziduum r_{k+1}. Volme

  $$p_{k+1} = r_{k+1} - \sum_{j=0}^{k} c_{k,j}\, p_j\,,$$

  koeficienty $c_{k,j}$ určíme tak, aby $\langle p_{k+1}, p_j \rangle_A = 0$, tj.

  $$c_{k,j} = \frac{\langle r_{k+1}, p_j \rangle_A}{\langle p_j, p_j \rangle_A}\,, \qquad j = 0, \ldots, k\,.$$

  Z lineární nezávislosti $r_{k+1}, p_0, \ldots, p_k \Rightarrow p_{k+1} \neq 0$.
- Vše dobře definováno, $p_j \neq 0 \Rightarrow \langle p_j, p_j \rangle_A > 0$.
- Start $p_0 = r_0 = b - Ax_0$.

Algorithm 3.1 Prototyp metody sdružených gradientů

input A, b, x_0
$r_0 = b - Ax_0$
$p_0 = r_0$
for $k = 0, 1, 2, \ldots$ **do**
  $\gamma_k = \frac{p_k^T r_k}{p_k^T A\, p_k}$
  $x_{k+1} = x_k + \gamma_k p_k$
  $r_{k+1} = r_k - \gamma_k A\, p_k$
  $c_{k,j} = \frac{\langle r_{k+1}, p_j \rangle_A}{\langle p_j, p_j \rangle_A}, \quad j = 0, \ldots, k$
  $p_{k+1} = r_{k+1} - \sum_{j=0}^{k} c_{k,j}\, p_j$
end for

- Z předchozího, nalezne řešení nejvýše po n iteracích.
- Lze zapsat efektivněji?

3.6 Algoritmus metody CG

- Prostory $\mathcal{S}_i$ jsou vnořené $\mathcal{S}_1 \subset \cdots \subset \mathcal{S}_{k+1}$.
- Z konstrukce, $j = 0, \ldots, k$,

$$r_j \in \text{span}\{p_0, \ldots, p_j\} = \mathcal{S}_{j+1},$$

$$r_{k+1} \perp \mathcal{S}_{k+1} \;\Rightarrow\; r_{k+1} \perp r_j,$$

rezidua jsou vzájemně ortogonální. Pak

$$r_0, \ldots, r_j \in \text{span}\{p_0, \ldots, p_j\} = \mathcal{S}_{j+1}$$

implikuje $\text{span}\{r_0, \ldots, r_j\} = \mathcal{S}_{j+1}$.

- Platí $j = 0, \ldots, k$

$$\underbrace{r_{j+1}}_{\in \mathcal{S}_{j+2}} = \underbrace{r_j}_{\in \mathcal{S}_{j+1}} - \gamma_j\, Ap_j \;\Rightarrow\; Ap_j \in \mathcal{S}_{j+2}.$$

$r_{k+1} \perp \mathcal{S}_{k+1} \Rightarrow r_{k+1} \perp Ap_j$, $j = 0, \ldots, k-1$,

$$c_{k,j} = \frac{r_{k+1}^T A p_j}{p_j^T A p_j} = 0, \quad j < k$$

a

$$p_{k+1} \;=\; r_{k+1} - c_{k,k}\, p_k\,.$$

- Z $r_k \perp p_{k-1}$ plyne

$$p_k^T r_k = r_k^T p_k = r_k^T (r_k + c_{k-1,k-1} p_{k-1}) = \|r_k\|^2,$$

a proto

$$\gamma_k = \frac{p_k^T r_k}{p_k^T A p_k} = \frac{r_k^T r_k}{p_k^T A p_k}.$$

- Označme

$$\delta_{k+1} \equiv -c_{k,k} = -\frac{r_{k+1}^T A p_k}{p_k^T A p_k}.$$

Vyjádři $-Ap_k$ z $r_{k+1} = r_k - \gamma_k A p_k$ a dosaď,

$$\begin{aligned} \delta_{k+1} &= \frac{r_{k+1}^T \frac{(r_{k+1} - r_k)}{\gamma_k}}{p_k^T A p_k} = \frac{r_{k+1}^T (r_{k+1} - r_k)}{\gamma_k p_k^T A p_k} \\ &= \frac{\|r_{k+1}\|^2}{\|r_k\|^2}. \end{aligned}$$

Algorithm 3.2 Metoda sdružených gradientů (CG)

input A, b, x_0
$r_0 = b - Ax_0$
$p_0 = r_0$
for $k = 0, 1, 2, \ldots$ **do**
  $\gamma_k = \frac{r_k^T r_k}{p_k^T A\, p_k}$
  $x_{k+1} = x_k + \gamma_k p_k$
  $r_{k+1} = r_k - \gamma_k A\, p_k$
  $\delta_{k+1} = \frac{r_{k+1}^T r_{k+1}}{r_k^T r_k}$
  $p_{k+1} = r_{k+1} + \delta_{k+1} p_k$
end for

Věta 3.3 (o metodě sdružených gradientů)**.** *$A \in \mathbb{R}^{n\times n}$ SPD, $b \in \mathbb{R}^n$, $x_0 \in \mathbb{R}^n$ počáteční přiblížení. Algoritmus 3.2 nalezne řešení v m-té iteraci, tj. $\|r_m\| = 0$, kde $m \le n$. Vektory r_i a p_i jsou pro $0 \le i < m$ nenulové a platí*

$$r_i^T r_j = 0 = p_i^T A p_j, \qquad 0 \le i < j < m.$$

Označíme-li jako v předchozím $\mathcal{S}_k = \operatorname{span}\{p_0, \ldots, p_{k-1}\}$ pro $k \ge 1$, je aproximace x_k optimální ve smyslu

$$\|x_* - x_k\|_A = \min_{x \in x_0 + \mathcal{S}_k} \|x_* - x\|_A.$$

Dále platí

$$x_k \in x_0 + \mathcal{S}_k, \quad r_k \perp \mathcal{S}_k.$$

Důkaz. Věta je srhrnutím předchozích výsledků. □

Další vlastnosti

- A-norma chyby je monotónně klesající, platí

$$\|x - x_k\|_A^2 - \|x - x_{k+1}\|_A^2 = \gamma_k \|r_k\|^2.$$

- 2-norma chyby je monotónně klesající.
- Pro $x_0 = 0$ je $\|x_k\|$ ostře rostoucí.

3.7 Praktické otázky

- Paměťové nároky?
- Není-li x_0 k dispozici, volíme $x_0 = 0$.
- Kdy zastavit iterace?
- Jak zrychlit konvergenci?
- Chování v počítačové aritmetice?

Zastavovací kritéria

Tolerance $\tau > 0$. Pro zastavení lze využít různé veličiny.

- Relativní norma rezidua

$$\frac{\|r_k\|}{\|b\|} \leq \tau$$

  reprezentuje relativní velikost změny pravé strany,

$$Ax_k = b - r_k = b + \Delta b\,.$$

- Relativní zpětná chyba

$$\frac{\|r_k\|}{\|b\| + \|A\|\|x_k\|} \leq \tau$$

  udává relativní velikosti nejmenších změn ΔA a Δb:

$$(A + \Delta A)x_k = b + \Delta b\,.$$

- Normy chyby (A-norma, Euklidovská), například

$$\|x_* - x_k\|_A \leq \tau, \quad \frac{\|x_* - x_k\|_A}{\|x_*\|_A} \leq \tau.$$

  Existují techniky jak normy chyb odhadovat.

Výběr kritéria i tolerance → závisí na řešeném problému.

4

Konvergence a předpodmínění

4.1 CG a Krylovovy prostory

Z algoritmu a odvození plyne

$$\mathcal{S}_{k+1} = \text{span}\{ \underbrace{p_0, \ldots, p_k}_{A-\text{ortogonální}} \} = \text{span}\{ \underbrace{r_0, \ldots, r_k}_{\text{ortogonální}} \}.$$

Lze formálně popsat pomocí Krylovova podprostoru?

$$\mathcal{K}_{k+1}(A, v) \equiv \text{span}\{v, Av, A^2v, \ldots, A^kv\}.$$

Lemma 4.1 (CG a Krylovovy podprostory)**.** *Po k iteracích metody sdružených gradientů s $r_j \neq 0$, $j = 0, \ldots, k$, platí*

$$\mathcal{S}_{k+1} = \mathcal{K}_{k+1}(A, r_0).$$

Důkaz. Indukcí, pro $k = 0$ platí.

Nechť $\mathcal{S}_k = \mathcal{K}_k(A, r_0)$. Pak z

$$\begin{aligned} r_k &= r_{k-1} - \gamma_{k-1} A\, p_{k-1} \\ p_k &= r_k + \delta_k p_{k-1} \end{aligned}$$

plyne $r_k \in \mathcal{K}_{k+1}(A, r_0)$, $p_k \in \mathcal{K}_{k+1}(A, r_0)$ a proto

$$\mathcal{S}_{k+1} \subseteq \mathcal{K}_{k+1}(A, r_0).$$

Z lineární nezávislosti $r_0, \ldots, r_k$ plyne rovnost. □

4.2 Konvergence CG

Chyba pomocí polynomů

- Vektory lze zapsat pomocí polynomu

$$v \in \mathcal{K}_k(A, w) \quad \Rightarrow \quad v = q(A)w\,,$$

kde q je polynom stupně nejvýše $k-1$.

- Zde $x - x_0 \in \mathcal{K}_k(A, r_0)$ a $r_0 = A(x_* - x_0) \Rightarrow$

$$x - x_0 \;=\; q(A)r_0 = Aq(A)(x_* - x_0)$$

a

$$\begin{aligned} x_* - x &= (x_* - x_0) - (x - x_0) \\ &= I(x_* - x_0) - Aq(A)(x_* - x_0) \\ &= p(A)(x_* - x_0)\,, \end{aligned}$$

kde $p(z) \equiv 1 - z\,q(z)$ je stupně nejvýše k a $p(0) = 1$.

- Označme

$$\pi_k \equiv \{p \in \mathcal{P}_k :\ \deg p \le k,\ p(0) = 1\},$$

pak

$$\begin{aligned} \|x_* - x_k\|_A &= \min_{x \in x_0 + \mathcal{K}_k} \|x_* - x\|_A \\ &= \min_{p \in \pi_k} \|p(A)(x_* - x_0)\|_A. \end{aligned}$$

Jaký aproximační problém řeší CG?

- Spektrální rozklad $A = Q\Lambda Q^T$, definujme

$$\omega = [\omega_1, \ldots, \omega_n]^T \equiv Q^T r_0 \,.$$

- Využijeme-li $\|v\|_A = \|A^{1/2}v\|$,

$$\begin{aligned} \|p(A)(x_* - x_0)\|_A^2 &= \|p(A)A^{1/2}(x_* - x_0)\|^2 \\ &= \|p(A)A^{-1/2}r_0\|^2 \\ &= \|p(\Lambda)\Lambda^{-1/2}\omega\|^2 \\ &= \sum_{j=1}^{n} \frac{\omega_j^2}{\lambda_j}\, p(\lambda_j)^2 \,. \end{aligned}$$

- Dosazením

$$\|x_* - x_k\|_A = \min_{p \in \pi_k} \left(\sum_{j=1}^{n} \frac{\omega_j^2}{\lambda_j}\, p(\lambda_j)^2 \right)^{1/2} .$$

- Konvergence CG závisí na
  - velikosti projekcí r_0 do báze vlastních vektorů,
  - rozložení vlastních čísel matice A.
- Jak odhadnout veličinu na pravé straně?

Odhad rychlosti konvergence

- Pomocí $\|v\|_A = \|A^{1/2}v\|$ dostaneme

$$\|p(A)(x_* - x_0)\|_A \quad \leq \quad \|p(A)\| \, \|x_* - x_0\|_A$$

a

$$\begin{aligned} \frac{\|x_* - x_k\|_A}{\|x_* - x_0\|_A} &\leq \min_{p \in \pi_k} \|p(A)\| \\ &= \min_{p \in \pi_k} \max_{i=1,\ldots,n} |p(\lambda_i)| \\ &\leq \min_{p \in \pi_k} \max_{\lambda \in [\lambda_1, \lambda_n]} |p(\lambda)|. \end{aligned}$$

- Nejprve jsme oddělili vliv b.
- Optimální polynomy se adaptují na spektrum.
- Problém na intervalu $\rightarrow$ Čebyševovy polynomy (posunuté, škálované), jejich dosazením dostaneme

$$\min_{p \in \pi_k} \max_{\lambda \in [\lambda_1, \lambda_n]} |p(\lambda)| \leq 2 \left(\frac{\sqrt{\kappa(A)} - 1}{\sqrt{\kappa(A)} + 1} \right)^k .$$

- $\kappa(A)$ malé $\Rightarrow$ CG konverguje rychle.
- $\kappa(A)$ velké $\not\Rightarrow$ CG konverguje pomalu.
- Vliv rozložení λ_i na konvergenci může být značný.

4.3 Předpodmínění

- Kdy je CG konvergence rychlá?
  - Malé číslo podmíněnosti.
  - Vhodné rozložení vlastních čísel.
  - r_0 má málo nenulových komponent.
- Předpodmínění je modifikace (transformace) soustavy za účelem zrychlení konvergence.
- Matice modifikované soustavy by měla být SPD.
- Nechť M je SPD a nechť jsou lineární systémy

  $$Mx = c$$

  rychle řešitelné $\rightarrow \mathcal{O}(n)$ operací.
- Příklady: M diagonální, M dána jako součin dolní a horní trojúhelníkové matice.
- Nechť $M = CC^T$, C regulární. Štěpení potřeba jen pro odvození.
- $Ax = b$ lze transformovat pomocí M:

  $$(\underbrace{C^{-1}AC^{-T}}_{\widehat{A}})(\underbrace{C^T x}_{\widehat{x}}) \;=\; \underbrace{C^{-1}b}_{\widehat{b}}\,.$$

  $\widehat{A}$ je SPD matice, vztah mezi řešeními $x = C^{-T}\widehat{x}$.

Algorithm 4.1 CG pro $\hat{A}\hat{x} = \hat{b}$

input $\hat{A}$, $\hat{b}$, $\hat{x}_0 = C^T x_0$
$\hat{r}_0 = \hat{b} - \hat{A}\hat{x}_0$
$\hat{p}_0 = \hat{r}_0$
for $k = 0, 1, 2, \ldots$ **do**
  $\hat{\gamma}_k = \frac{\hat{r}_k^T \hat{r}_k}{\hat{p}_k^T \hat{A}\, \hat{p}_k}$
  $\hat{x}_{k+1} = \hat{x}_k + \hat{\gamma}_k \hat{p}_k$
  $\hat{r}_{k+1} = \hat{r}_k - \hat{\gamma}_k \hat{A}\, \hat{p}_k$
  $\hat{\delta}_{k+1} = \frac{\hat{r}_{k+1}^T \hat{r}_{k+1}}{\hat{r}_k^T \hat{r}_k}$
  $\hat{p}_{k+1} = \hat{r}_{k+1} + \hat{\delta}_{k+1} \hat{p}_k$
end for

- CG na $\widehat{A}\widehat{x} = \widehat{b} \rightarrow$ aproximace řešení $\widehat{x}_k$.
- Definujme aproximace $x = C^{-T}\widehat{x}$ jako

$$x_k \equiv C^{-T}\widehat{x}_k.$$

- Reziduum k x_k je

$$r_k \equiv b - Ax_k = C(\widehat{b} - \widehat{A}\widehat{x}_k) = C\widehat{r}_k.$$

- Položme

$$p_k \equiv C^{-T}\widehat{p}_k \quad \text{a} \quad z_k \equiv C^{-T}C^{-1}r_k,$$

  přepišme CG pro $\widehat{A}\widehat{x} = \widehat{b}$ pomocí těchto vektorů a $M = CC^T$.

Algorithm 4.2 PCG

input A, b, x_0, M
$r_0 = b - Ax_0$
$z_0 = M^{-1}r_0$
$p_0 = z_0$
for $k = 0, 1, 2, \ldots$ **do**
  $\hat{\gamma}_k = \frac{z_k^T r_k}{p_k^T A p_k}$
  $x_{k+1} = x_k + \hat{\gamma}_k p_k$
  $r_{k+1} = r_k - \hat{\gamma}_k A p_k$
  $z_{k+1} = M^{-1} r_{k+1}$
  $\hat{\delta}_{k+1} = \frac{z_{k+1}^T r_{k+1}}{z_k^T r_k}$
  $p_{k+1} = z_{k+1} + \hat{\delta}_{k+1} p_k$
end for

- Nepočítáme inverzi M, platí

$$z_k = M^{-1} r_k \quad \Leftrightarrow \quad M z_k = r_k \,,$$

  tj. v každé iteraci řešíme soustavu s maticí M.

- Předpodmínění $\rightarrow$ kombinace přímých metod (řešíme soustavy s M) a iteračních metod (CG).

- Soustavu s M je možné řešit i iteračně (vnější a vnitřní iterace).

Vlastnosti PCG

- Rovnost

$$\|\hat{x} - \hat{x}_k\|_{\hat{A}}^2 \quad = \quad \|x - x_k\|_A^2,$$

  plyne rozepsáním chyby na levé straně pomocí

$$(C^T(x - x_k))^T C^{-1} A C^{-T} (C^T(x - x_k)).$$

- Důsledek: x_k minimalizuje $\|x - x_k\|_A$ přes

$$x_0 + \mathcal{K}_k(M^{-1}A, M^{-1}r_0).$$

- Z vlastností CG pro $\widehat{A}\widehat{x} = \widehat{b}$ plyne pro $i \neq j$

$$\begin{aligned} 0 &= \widehat{r}_i^T \widehat{r}_j = r_i^T M^{-1} r_j = z_i^T M z_j, \\ 0 &= \widehat{p}_i^T \widehat{A} \widehat{p}_j = p_i^T A p_j. \end{aligned}$$

Část III

Krylovovské metody

5

Projektivní metody

- $Ax = b$, $A \in \mathbb{R}^{n \times n}$ symetrická, poz. def., $b \in \mathbb{R}^n$.
- CG: x_k jednoznačně určeno podmínkami

$$x_k \in x_0 + \mathcal{S}_k, \quad r_k \perp \mathcal{S}_k,$$

$$\mathcal{S}_k = \mathcal{K}_k(A, v).$$

- Projekce problému na podprostor s menší dimenzí.
- Lze zobecnit pro obecnou regulární A?

5.1 Projektivní metody

Idea

- Dáno $A \in \mathbb{R}^{n\times n}$, $b \in \mathbb{R}^n$, $x_0 \in \mathbb{R}^n$, problém
$$Ax = b.$$
- Idea Petrov-Galerkin, diskretizace variačních úloh, hledáme
$$x_k \in x_0 + \mathcal{S}_k, \quad r_k \perp \mathcal{C}_k,$$
kde $\mathcal{S}_k$ je „search space“ a $\mathcal{C}_k$ je „constraint space“,
$$\dim \mathcal{S}_k = k = \dim \mathcal{C}_k.$$
- Rostou-li dimenze a existují-li x_k, je pro $k = n$
$$r_n \perp \mathcal{C}_n \quad \Rightarrow \quad r_n = 0.$$

Realizace

- S_k báze $\mathcal{S}_k$, C_k báze $\mathcal{C}_k$,
$$\begin{aligned} x_k &= x_0 + S_k y_k, \\ r_k &= r_0 - AS_k y_k. \end{aligned}$$
- Kolmost $r_k \perp \mathcal{C}_k$ lze zapsat ve tvaru
$$C_k^T (r_0 - AS_k y_k) = 0$$
neboli
$$\left(C_k^T AS_k\right) y_k = C_k^T r_0.$$
- Je-li $C_k^T AS_k$ regulární, je y_k dobře definováno.

Dvě základní volby

- $\mathcal{C}_k = \mathcal{S}_k$ a A **symetrická pozitivně definitní** $\Rightarrow$

$$S_k^T A S_k \quad \text{je regulární}$$

a

$$x_k \in x_0 + \mathcal{S}_k, \quad (x - x_k)^T A S_k = 0$$

implikuje minimální A-normu chyby.

- $\mathcal{C}_k = A\mathcal{S}_k$ a A **obecná regulární** $\Rightarrow$

$$S_k^T \left(A^T A\right) S_k \quad \text{je regulární}$$

a

$$r_k \in r_0 + A\mathcal{S}_k, \quad r_k \perp A\mathcal{S}_k$$

implikuje minimální normu rezidua.

5.2 Krylovovské metody

Na *základní* krylovovské metody lze nahlížet jako na projektivní metody s volbou $\mathcal{S}_k \equiv \mathcal{K}_k(A, r_0)$.

Krylovovy podprostory

- Maximální dimenzi, jež mohou $\mathcal{K}_k(A, v)$ dosáhnout,

  $$d(A, v) \equiv \min\{k : \dim \mathcal{K}_k(A, v) = \dim \mathcal{K}_{k+1}(A, v)\},$$

  nazveme stupněm v vzhledem k A. Zkráceně d.

- $\mathcal{K}_d(A, v)$ je invariantní vzhledem k násobení A

  $$x \in \mathcal{K}_d(A, v) \Rightarrow Ax \in \mathcal{K}_d(A, v)\,.$$

- Důsledek

  $$\mathcal{K}_1(A, v) \subset \mathcal{K}_2(A, v) \subset \cdots \subset \mathcal{K}_d(A, v) = \ldots.$$

- Označme $d(A)$ stupeň minimálního polynomu.

- Platí

  1. $d \equiv d(A, v) \leq d(A) \leq n$,
  2. $\dim(\mathcal{K}_k(A, v)) = k$, $k \leq d$,
  3. $\dim(\mathcal{K}_k(A, v)) = d$, $k \geq d$.

Základní krylovovské metody

$$\mathcal{S}_k = \mathcal{K}_k(A, r_0)$$

- $r_k = b - Ax_k$ definuje vztah mezi x_k a r_k,

  $$x_k \in x_0 + \mathcal{K}_k(A, r_0), \quad r_k \in r_0 + A\mathcal{K}_k(A, r_0)$$

  a $r_k \in \mathcal{K}_{k+1}(A, r_0)$.

- Volba $\mathcal{C}_k$
  - Je x_k dobře definováno?
  - Optimalita x_k?
  - Lze počítat krátkou rekurencí?

- $\mathcal{C}_k = \mathcal{K}_k$: Metody s **ortogonálními rezidui**

  $$r_k \perp \mathcal{K}_k(A, r_0)$$

  nemusí být pro regulární A dobře definovány.

- $\mathcal{C}_k = A\mathcal{K}_k$: Metody s **minimálními rezidui**

  $$r_k \perp A\mathcal{K}_k(A, r_0)$$

  vždy dobře definovány.

- $\mathcal{C}_k \equiv \mathcal{K}_k(A^T, s_0)$, s_0 pomocný vektor. x_k lze počítat krátkými rekurencemi, ne vždy dobře definováno.

- Praktická realizace? Báze? Krátké rekurence?

5.3 Výpočet ortogonální báze

Arnodiho algoritmus

Algorithm 5.1 Arnoldiho algoritmus (MGS verze)

1: **input** A, v
2: $v_1 = \frac{v}{\|v\|}$
3: **for** $k = 1, 2, \ldots$ **do**
4: $\quad w = Av_k$
5: $\quad$ **for** $i = 1 : k$ **do**
6: $\quad\quad h_{i,k} = \langle w, v_i \rangle$
7: $\quad\quad w = w - h_{i,k}\, v_i$
8: $\quad$ **end for**
9: $\quad h_{k+1,k} = \|w\|$
10: $\quad v_{k+1} = \frac{w}{h_{k+1,k}}$
11: **end for**

- Označíme-li $V_k \equiv [v_1, \ldots, v_k]$ a

$$H_k = \begin{bmatrix} h_{1,1} & h_{1,2} & \ldots & h_{1,k} \\ h_{2,1} & h_{2,2} & \ldots & h_{2,k} \\ & \ddots & \ddots & \vdots \\ & & h_{k,k-1} & h_{k,k} \end{bmatrix},$$

  platí

$$AV_k = V_k H_k + h_{k+1,k} v_{k+1} e_k^T, \quad V_k^T A V_k = H_k.$$

- Počítá ortogonální bázi $\mathcal{K}_k(A, v)$ dlouhou rekurencí.

Lanczosův algoritmus

Algorithm 5.2 Lanczosův algoritmus

1: **input** A, v
2: $v_0 = 0$, $\beta_1 = 0$
3: $v_1 = v/\|v\|$
4: **for** $k = 1, 2, \ldots$ **do**
5: $\quad w = Av_k - \beta_k v_{k-1}$
6: $\quad \alpha_k = v_k^T w$
7: $\quad w = w - \alpha_k v_k$
8: $\quad \beta_{k+1} = \|w\|$
9: $\quad v_{k+1} = w/\beta_{k+1}$
10: **end for**

- $A = A^T$, pak je $H_k = H_k^T$

$$H_k = \begin{bmatrix} \alpha_1 & \beta_2 & & & \\ \beta_2 & \alpha_2 & \beta_3 & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & \beta_k \\ & & & \beta_k & \alpha_k \end{bmatrix} \equiv T_k$$

a platí

$$AV_k = V_k T_k + \beta_{k+1} v_{k+1} e_k^T.$$

- Počítá ortogonální bázi $\mathcal{K}_k(A, v)$ krátkou rekurencí.

5.4 Metody nad ortogonální bází

- Prototyp metody nad ortogonální bází:
  - použij Arnoldi či Lanczos
  - uvažuj $x_k = x_0 + V_k y_k$
  - z určujících podmínek spočti y_k
- Výpočet **ortogonální** báze $\rightarrow$ Arnoldiho, Lanczos

$$AV_k = V_k H_k + h_{k+1,k} v_{k+1} e_k^T = V_{k+1} \underline{H}_k$$

- Vyjádříme-li $x_k = x_0 + V_k y_k$, pak

$$\begin{aligned} r_k &= r_0 - AV_k y_k \\ &= \|r_0\| v_1 - V_{k+1} \underline{H}_k y_k \\ &= V_{k+1} \left(\|r_0\| e_1 - \underline{H}_k y_k \right). \end{aligned}$$

- Metody s **ortogonálními rezidui**

$$\begin{aligned} r_k \perp \mathcal{K}_k(A, r_0) \quad &\Leftrightarrow \quad V_k^T (r_0 - AV_k y_k) = 0 \\ &\Leftrightarrow \quad \|r_0\| e_1 = H_k y_k \end{aligned}$$

- Metody s **minimálními rezidui**

$$\begin{aligned} \|r_k\| &= \min_{y \in \mathbb{R}^k} \|V_{k+1} \left(\|r_0\| e_1 - \underline{H}_k y \right)\| \\ &= \min_{y \in \mathbb{R}^k} \| \|r_0\| e_1 - \underline{H}_k y \|. \end{aligned}$$

5.5 Metody pro $A = A^*$

Použijeme **Lanczosův algoritmus** (krátké rekurence)

$$AV_k = V_k T_k + \beta_{k+1} v_{k+1} e_k^T = V_{k+1} \underline{T}_k.$$

Metoda s ortogonálními rezidui

$$\begin{aligned} \|r_0\| e_1 &= T_k y_k^{(G)}, \\ x_k^{(G)} &= x_0 + V_k y_k^{(G)}. \end{aligned}$$

- Je-li A SPD, pak je i T_k SPD a $x_k^{(G)} = x_k^{(CG)}$.
- Obecně T_k může být singulární, $x_k^{(G)}$ nemusí existovat, ale musí existovat v každém druhém kroku.
- Je možné počítat $x_k^{(G)}$ krátkou rekurencí? ANO.
- Zvol vhodný rozklad T_k a využij *tridiagonální* struktury, lze levně updatovat! Cholesky → **cgLanczos** nebo QR → **SYMMLQ**.
- Lze ukázat
$$x_{k+1}^{(G)} = x_k^{(G)} + \alpha_k w_k$$
a w_k lze updatovat dvoučlennou rekurencí.
- Pro indefinitní A není obecně optimální.

Metoda s minimálními rezidui

- Pro indefinitní A lepší použít metodu s minimálními rezidui, vždy dobře definována.
- Myšlenka: hledáme $x_k = x_0 + V_k y_k$ tak, aby

$$\begin{aligned} \|r_k\| &= \min_{y \in \mathbb{R}^k} \|V_{k+1} \left(\|r_0\| e_1 - \underline{T}_k y\right)\| \\ &= \min_{y \in \mathbb{R}^k} \left\|\|r_0\| e_1 - \underline{T}_k y\right\| . \end{aligned}$$

- Řešíme problém nejmenších čtverců

$$\begin{bmatrix} \alpha_1 & \beta_2 & & & \\ \beta_2 & \alpha_2 & \beta_3 & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & \beta_k \\ & & & \beta_k & \alpha_k \\ & & & & \beta_{k+1} \end{bmatrix} y_k \approx \begin{bmatrix} \|r_0\| \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} .$$

- $\underline{T}_k$ má vždy lineárně nezávislé sloupce, $\beta_j \neq 0$.
- Efektivní realizace $\rightarrow$ **MINRES** (krátké rekurence)
- Optimální $\rightarrow$ minimalizuje normu rezidua.

5.6 Metody pro obecnou A

Použijeme **Arnoldiho algoritmus** (dlouhé rekurence)

$$AV_k = V_k H_k + h_{k+1,k} v_{k+1} e_k^T = V_{k+1} \underline{H}_k,$$

paměťově náročné.

Metoda s ortogonálními rezidui

- FOM: Full Orthogonalization Method,

$$x_k^{(FOM)} = x_0 + V_k y_k^{(FOM)}.$$

- Podmínky $r_k \perp \mathcal{C}_k = \mathcal{K}_k(A, r_0)$ vedou na

$$\|r_0\| e_1 \quad = \quad H_k y_k^{(FOM)},$$

- H_k může být **singulární** ve více iteracích po sobě.
- Pro $d = d(A, r_0)$ je $AV_d = V_d H_d$. Tudíž, je-li A regulární, pak je i H_d regulární. Zároveň platí

$$r_d \in \mathcal{K}_{d+1}, \quad r_d \perp \mathcal{K}_d = \mathcal{K}_{d+1}$$

a nutně $r_d = 0 \Rightarrow x_d$ je řešení.

Metoda s minimálními rezidui

- GMRES: Generalized minimal residual method,

$$x_k = x_0 + V_k y_k.$$

- Podmínky $r_k \perp \mathcal{C}_k = A\mathcal{K}_k(A, r_0)$ vedou na

$$\|r_k\| \quad = \quad \min_{y \in \mathbb{R}^k} \left\| \|r_0\| e_1 - \underline{H}_k y \right\|,$$

  LS problém s $\underline{H}_k$, lineárně nezávislé sloupce $\forall k \leq d$.

- Givens-QR, eliminujeme podiagonálu

$$\underline{H}_k = Q_k \underline{R}_k = Q_k \begin{bmatrix} R_k \\ 0 \end{bmatrix},$$

  R_k je horní trojúhelníková (plný trojúhelník).

- Pak

$$\underline{H}_k y \approx \|r_0\| e_1 \quad \Leftrightarrow \quad \underline{R}_k y \approx \|r_0\| Q_k^T e_1 \equiv c$$

  a y_k je řešením soustavy $c_{1:k} = R_k y_k$.

- Nalezne řešení po právě d iteracích. Z

$$r_d \in \mathcal{K}_{d+1}, \quad r_d \perp A\mathcal{K}_d = \mathcal{K}_{d+1}$$

  plyne $r_d = 0$. Pokud $r_k = 0$ pro $k < d$, pak $r_0 \in A\mathcal{K}_k$ což implikuje $\mathcal{K}_k = \mathcal{K}_{k+1}$, spor s definicí d.

- Paměťově náročná metoda. Restarty $\rightarrow$ úspora paměti, ale zpomalení konvergence.

Analýza konvergence GMRES

- Podobně jako u CG platí

$$r_k = p_k(A)r_0, \quad p_k \in \pi_k,$$

a z podmínky minimální normy rezidua

$$\|r_k\| = \min_{p\in\pi_k} \|p(A)r_0\| \leq \|r_0\| \min_{p\in\pi_k} \|p(A)\|.$$

- Jordanův rozklad $A = XJX^{-1}$,

$$\begin{aligned} \min_{p\in\pi_k} \|p(A)\| &= \min_{p\in\pi_k} \|Xp(J)X^{-1}\| \\ &\leq \kappa(X) \min_{p\in\pi_k} \|p(J)\| \end{aligned}$$

a pro A diagonalizovatelnou

$$\|r_k\| \leq \|r_0\|\kappa(X) \min_{p\in\pi_k} \max_i |p(\lambda_i)|.$$

- $\kappa(X) = 1 \Leftrightarrow A$ je normální. Obecně $\kappa(X)$ může být obrovské $\rightarrow$ odhady nepoužitelné.
- Konvergence GMRES $\rightarrow$ otevřený problém.

Spektrum nemusí souviset s konvergencí!

Lze pozorovat libovolnou konvergenci GMRES pro libovolné spektrum, viz následující věta.

Věta 5.1 (Greenbaum, Pták, Strakoš)**.** *Nechť je dána posloupnost čísel*

$$f_0 \geq f_1 \geq \cdots \geq f_{n-1} > 0$$

a nenulová $\lambda_i \in \mathbb{C}$*,* $i = 1, \ldots, n$*. Pak existuje* $A \in \mathbb{C}^{n\times n}$ *s vlastními čísly* λ_i *a* $b \in \mathbb{C}^n$*,* $\|b\| = f_0$*, tak, že rezidua metody GMRES aplikované na* $Ax = b$ *s* $x_0 = 0$ *splňují*

$$\|r_k\| = f_k, \qquad k = 1, \ldots, n-1.$$

Důkaz. [6]. □

5.7 Neortogonální báze

GMRES optimální, ale paměťově náročná (Arnoldi).

Lze použít **krátké rekurence** na úkor **optimality**?

Metoda bikonjugovaných gradientů

- Řeší současně

$$Ax = b \quad \text{a} \quad A^T y = c.$$

- Dáno A, b, c, x_0, y_0. Pokud chceme řešit jen $Ax = b$, volíme $y_0 = 0$ a $c = b$ nebo i c náhodně.
- Definujme $r_0 = b - Ax_0$, $s_0 = c - A^T y_0$.
- Hledáme aproximace

$$x_k \in x_0 + \mathcal{K}_k(A, r_0), \quad y_k \in y_0 + \mathcal{K}_k(A^T, s_0)$$

  tak, že

$$r_k \perp \mathcal{K}_k(A^T, s_0), \quad s_k \perp \mathcal{K}_k(A, r_0).$$

- Rostou-li dimenze a existují-li aproximace $\rightarrow$ řešení v konečném počtu iterací.
- Lze počítat následujícím algoritmem.

Algorithm 5.3 BiCG

input A, A^T, b, c, x_0, y_0
$r_0 = p_0 = b - Ax_0$
$s_0 = q_0 = c - A^T y_0$
for $k = 0, 1, \ldots$ **do**
  $\alpha_k = \frac{s_k^T r_k}{q_k^T A p_k}$
  $x_{k+1} = x_k + \alpha_k p_k$
  $r_{k+1} = r_k - \alpha_k A p_k$
  $y_{k+1} = y_k + \alpha_k q_k$
  $s_{k+1} = s_k - \alpha_k A^T q_k$
  $\beta_{k+1} = \frac{s_{k+1}^T r_{k+1}}{s_k^T r_k}$
  $p_{k+1} = r_{k+1} + \beta_{k+1} p_k$
  $q_{k+1} = s_{k+1} + \beta_{k+1} q_k$
end for

- $r_k^T s_k = 0$ nebo $q_k^T A p_k = 0 \rightarrow$ nutné ukončit algoritmus, buď máme jedno z řešení ($r_k = 0$ nebo $s_k = 0$) nebo breakdown.
- Za předpokladu $r_k^T s_k \neq 0$ a $q_k^T A p_k \neq 0$:
  - r_i a s_j tvoří biortogonální posloupnost
$$r_i^T s_j = 0 = p_i^T A q_j, \quad i \neq j,$$
  - r_i LN a $\text{span}\{r_0, \ldots, r_{k-1}\} = \mathcal{K}_k(A, r_0)$,
  - s_i LN a $\text{span}\{s_0, \ldots, s_{k-1}\} = \mathcal{K}_k(A^T, s_0)$.

6

Faber-Manteuffelova věta

$$Ax = b$$

- Pro jaké třídy matic existuje **optimální** krylovovská metoda s **krátkými** rekurencemi?
- „optimální“ $\rightarrow$ minimalizuje *nějakou* normu chyby přes $x_0 + \mathcal{K}_k(A, r_0)$
- Redukce na otázku: Kdy je možné počítat **ortonormální** bázi $\mathcal{K}_k(A, r_0)$ pomocí **krátkých** rekurencí?
- Souhrnný článek Liesen - Strakoš [12].

6.1 Problém krátkých rekurencí

Kdy lze počítat pro $k = 0, \ldots, d-1$ ortogonální bázi krátkou rekurencí typu

$$h_{k+1,k} v_{k+1} = A v_k - \sum_{i=1}^{k} h_{i,k} v_i ?$$

Poslední spočtený je v_d, $v_{d+1} = 0$ mě nezajímá.

$$A\,V_{d-1} \quad = \quad V_d \quad \overbrace{\begin{bmatrix} \bullet & \cdots & \bullet & & \\ \bullet & \ddots & & \ddots & \\ & \ddots & \ddots & & \bullet \\ & & \ddots & \ddots & \vdots \\ & & & \ddots & \bullet \\ & & & & \bullet \end{bmatrix}}^{s+1} \Bigg\} d-1$$

Definice 6.1 (optimální rekurence)**.** Řekneme, že A umožňuje optimální $(s+2)$-člennou rekurenci ($s \geq 0$), pokud

- $\forall\ v$ je $H_{d,d-1}$ nejvýše $(s+2)$-pásová a
- $\exists v$: $H_{d,d-1}$ je $(s+2)$-pásová.

6.2 Nutné a postačující podmínky

Definice 6.2. $A \in \mathbb{R}^{n\times n}$ je normální stupně s, zkráceně normal(s), pokud existuje polynom p stupně s takový, že platí $A^* = p(A)$, a zároveň p je polynom nejmenšího stupně s touto vlastností.

Věta 6.3 (Faber a Manteuffel, [3])**.** *Nechť $A \in \mathbb{R}^{n\times n}$ a s je přirozené takové, že $s+2 < d(A)$. A umožňuje optimální $(s+2)$-člennou rekurenci $\Leftrightarrow$ A je normal(s).*

Důkaz. "$\Leftarrow$" Nechť $A^* = p(A)$, p stupně s.

Uvědomme si

$$v_i \in \mathcal{K}_i(A,v) \quad \Rightarrow \quad p(A)v_i \in \mathcal{K}_{i+s}(A,v).$$

Prvky matice $H_{d,d-1}$

$$h_{i,j} = \langle Av_j, v_i\rangle = \langle v_j, A^*v_i\rangle = \langle v_j, \underbrace{p(A)v_i}_{\mathcal{K}_{i+s}(A,v)}\rangle.$$

Pro $d > j > i+s$ platí

$$v_j \perp \mathcal{K}_{i+s}(A,v) \quad \Rightarrow \quad h_{i,j} = 0.$$

Celkem, j nejvýše $d-1$, tj. $h_{i,j} = 0$ pro

$$i = 1,\ldots,d-s-2, \quad j = i+s+1,\ldots,d-1.$$

Opačná implikace netriviální, viz [3] nebo [2]. □

6.3 Kdy je A normální stupně s?

- Je normální

$$A^*A = p(A)A = Ap(A) = AA^*.$$

- Použij spektrální rozklad, platí

$$A^* = p(A) \quad \Leftrightarrow \quad \overline{\lambda}_i = p(\lambda_i).$$

  Kolik kořenů může mít harmonický pol. $\overline{z} - p(z)$?

- Lze ukázat: A je normální stupně $1 \Leftrightarrow A$ je normální a její vlastní čísla leží na přímce.

- Lze ukázat: A je normální stupně $s > 1 \Leftrightarrow A$ je normální a $d(A) \leq 3s - 2$.

- Jediný praktický příklad $s = 1$, buď tříčlenná nebo plná rekurence.

- Více detailů v [11] a [13].

6.4 Nestandardní skalární součin

Definice 6.4. Nechť $A \in \mathbb{R}^{n\times n}$ a $B \in \mathbb{R}^{n\times n}$, kde B je symetrická pozitivně definitní. Definujme matici

$$A^+ = B^{-1}A^TB,$$

kterou nazveme *B-adjungovanou* maticí k A. *Řekneme, že A je B-normal(s)* pokud $\exists$ polynom p stupně s tak, že

$$A^+ = p(A)$$

a zároveň p je nejmenšího stupně s touto vlastností.

- B umožňuje změnu báze, A je B-normal(s) $\Leftrightarrow \hat{A} \equiv B^{1/2}AB^{-1/2}$ je normal(s):

$$B^{-1}A^TB = p_s(A) \quad \Leftrightarrow \quad \hat{A}^T = p_s(\hat{A}).$$

- Faber-Manteuffelovu větu lze aplikovat na $\hat{A}$ a ortogonální $\hat{v}_i$ lze generovat krátkou rekurencí

$$h_{k+1,k}\hat{v}_{k+1} = \hat{A}\hat{v}_k - \sum_{i=k-s}^{k} h_{i,k}\hat{v}_i.$$

- Definujme $v_i = B^{-1/2}\hat{v}_i$. Potom

$$h_{k+1,k}v_{k+1} = Av_k - \sum_{i=k-s}^{k} h_{i,k}v_i,$$

$0 = \hat{v}_j^T\hat{v}_i = v_j^TBv_i$, a v_i jsou B-ortogonální.

Věta 6.5 (Faber a Manteuffel, [3])**.** *Nechť jsou dány matice $A \in \mathbb{R}^{n\times n}$ a $B \in \mathbb{R}^{n\times n}$, kde B je symetrická pozitivně definitní. Nechť s je přirozené takové, že $s+2 < d(A)$. A umožňuje pro dané B optimální $(s+2)$-člennou rekurenci $\Leftrightarrow$ je A je B-normal(s).*

Část IV

Maticové funkce

7

Definice maticové funkce

- Dána čtvercová matice $A \in \mathbb{C}^{n\times n}$.
- Funkce na A: $\det(A)$, $\mathrm{trace}(A)$, $\kappa(A)$, $\|A\|$
- Maticové funkce: Mějme skalární funkci

  $$f : \mathbb{C} \to \mathbb{C},$$

  chceme ji rozšířit na matice,

  $$f : A \in \mathbb{C}^{n\times n} \mapsto f(A) \in \mathbb{C}^{n\times n}.$$

- Příklady: inverze, racionální funkce, polynom.

Značení

$A \in \mathbb{C}^{n\times n}$ je podobná J v Jordanově kanonickém tvaru,

$$A = XJX^{-1},$$

kde X je regulární. Označme

$$J = \begin{bmatrix} J_1 & & & \\ & J_2 & & \\ & & \ddots & \\ & & & J_k \end{bmatrix}, \quad J_j = \begin{bmatrix} \lambda_j & 1 & & \\ & \lambda_j & \ddots & \\ & & \ddots & 1 \\ & & & \lambda_j \end{bmatrix},$$

kde

- k je počet bloků,
- $J_j \in \mathbb{C}^{n_j\times n_j}$, $j = 1, \ldots, k$, Jordanův blok,
- λ_j vlastní číslo j-tého Jordanova bloku,
- $\mu_1, \ldots, \mu_\ell$ navzájem různá vlastní čísla A,
- m_i velikost maximálního Jordanova bloku k μ_i.

Minimální polynom

Minimální polynom matice A

$$\psi_A(z) = (z - \mu_1)^{m_1} \dots (z - \mu_\ell)^{m_\ell}$$

je stupně

$$d \equiv \sum_{i=1}^{\ell} m_i \,.$$

Lemma 7.1. *Nechť $A \in \mathbb{C}^{n\times n}$ a nechť $p(A) = 0$, kde $p \in \mathcal{P}_j$ a $j > d$. Pak ψ_A dělí p.*

Důkaz. Podle věty o dělení polynomů [9, p. 128] je

$$p(z) = \psi_A(z)s(z) + r(z), \quad \deg(r) < \deg(\psi_A),$$

r a s jsou jednoznačné. Chceme ukázat, že r je nulový.

Z předpokladu $p(A) = 0$ plyne $r(A) = 0$, neboť

$$0 = p(A) = \psi_A(A)s(A) + r(A) = r(A).$$

Jelikož je $\psi_A(A)$ polynom minimálního stupně anihilující A, tak z podmínek $r(A) = 0$ a $\deg(r) < \deg(\psi_A)$ plyne $r(z) \equiv 0$. □

Důsledek: $\psi_A(z)$ dělí $p(z) \Rightarrow p(\lambda_i) = 0$.

7.1 Maticové polynomy

Věta 7.2. *Nechť $A \in \mathbb{C}^{n\times n}$ a nechť p a q jsou polynomy. Potom platí*

$$\underbrace{p(A) = q(A)}_{\substack{\text{rovnost maticových}\\ \text{polynomů}}} \quad \Leftrightarrow \quad \underbrace{p^{(i)}(\mu_j) = q^{(i)}(\mu_j)}_{\substack{\text{rovnost polynomů}\\ \text{na spektru}}},$$

kde $j = 1, \dots, \ell$ a $i = 0, \dots, m_j - 1$.

Důkaz. Označme $r(z) = p(z) - q(z)$ rozdíl polynomů.

„$\Rightarrow$“ Je-li $r(A) = 0$ pak dle Lemma 7.1 platí $r(z) = \psi_A(z)s(z)$ a každé μ_j je tudíž nejméně m_j násobným kořenem $r(z)$. Z toho plyne, že

$$r^{(i)}(\mu_j) = 0, \quad i = 0, \dots, m_j - 1.$$

„$\Leftarrow$“ Pokud platí $r^{(i)}(\mu_j) = 0$, $i = 0, \dots, m_j - 1$, pak je každé μ_j nejméně m_j násobným kořenem $r(z)$. Polynom ψ_A tudíž dělí r beze zbytku, což implikuje

$$r(A) = 0 \;\Rightarrow\; p(A) = q(A).$$

□

Vidíme, že matice $p(A)$, kde p je polynom, je určena hodnotami a derivacemi p na spektru,

$$p^{(i)}(\mu_j)$$

kde $j = 1, \dots, \ell$ a $i = 0, \dots, m_j - 1$. Zobecnění na funkce?

7.2 Definice $f(A)$ (Taylor)

Motivace

- Nechť A je **diagonalizovatelná** a p je polynom,

$$A = X\Lambda X^{-1}, \quad p(z) = \sum_{i=0}^{k} \alpha_i z^i.$$

Pak

$$\begin{aligned} p(A) &= \sum_{i=0}^{k} \alpha_i A^i = X \left(\sum_{i=0}^{k} \alpha_i \Lambda^i \right) X^{-1} \\ &= X \begin{bmatrix} p(\lambda_1) & & \\ & \ddots & \\ & & p(\lambda_n) \end{bmatrix} X^{-1}. \end{aligned}$$

- Pro $A \in \mathbb{C}^{n\times n}$, $A = XJX^{-1}$, se nabízí definovat $f(A)$ pomocí

$$Xf(J)X^{-1} = X \begin{bmatrix} f(J_1) & & \\ & \ddots & \\ & & f(J_k) \end{bmatrix} X^{-1}.$$

- Jak definovat $f(B)$, kde B je Jordanův blok?

$f(B)$ pro Jordanův blok

Nechť

$$B = \begin{bmatrix} \lambda & 1 & & \\ & \lambda & \ddots & \\ & & \ddots & 1 \\ & & & \lambda \end{bmatrix} \in \mathbb{C}^{m\times m}$$

a nechť f je analytická (holomorfní) na otevřeném okolí λ. Pak na tomto okolí

$$f(z) = \sum_{i=0}^{\infty} \frac{f^{(i)}(\lambda)}{i!}(z-\lambda)^i.$$

Dosaďme B do Taylorovy řady,

$$\sum_{i=0}^{\infty} \frac{f^{(i)}(\lambda)}{i!}(B-\lambda I)^i \quad = \quad \sum_{i=0}^{m-1} \frac{f^{(i)}(\lambda)}{i!} \begin{bmatrix} 0 & 1 & & \\ & 0 & \ddots & \\ & & \ddots & 1 \\ & & & 0 \end{bmatrix}^i .$$

Definujme

$$f(B) \equiv \begin{bmatrix} f(\lambda) & \frac{f^{(1)}(\lambda)}{1!} & \cdots & \frac{f^{(m-1)}(\lambda)}{(m-1)!} \\ & f(\lambda) & \ddots & \vdots \\ & & \ddots & \frac{f^{(1)}(\lambda)}{1!} \\ & & & f(\lambda) \end{bmatrix}.$$

Poznámka. $f : \mathbb{R} \to \mathbb{R}$ je analytická na otevřené množině $D \subseteq \mathbb{R}$, je-li $f \in C^\infty(D)$ a její Taylorova řada konverguje $\forall x \in D$ (bodově). $f : \mathbb{C} \to \mathbb{C}$ je analytická (holomorfní) na otevřené $D \Leftrightarrow f \in C^\infty(D)$. Tato ekvivalence nemusí platit pro reálné funkce.

Definice (Taylor)

Definice 7.3. Nechť $A \in \mathbb{C}^{n\times n}$ a nechť $f^{(i)}(\mu_j)$ je definováno pro $j = 1, \ldots, \ell$ a $i = 0, \ldots, m_j - 1$. Pak maticovou funkci $f(A)$ definujeme jako

$$f(A) \equiv X f(J) X^{-1},$$

kde

$$f(J) \equiv \begin{bmatrix} f(J_1) & & \\ & \ddots & \\ & & f(J_k) \end{bmatrix}$$

a

$$f(J_j) \equiv \begin{bmatrix} f(\lambda_j) & \frac{f^{(1)}(\lambda_j)}{1!} & \cdots & \frac{f^{(n_j-1)}(\lambda_j)}{(n_j-1)!} \\ & f(\lambda_j) & \ddots & \vdots \\ & & \ddots & \frac{f^{(1)}(\lambda_j)}{1!} \\ & & & f(\lambda_j) \end{bmatrix}.$$

Poznámka 7.4. Pro fixní Jordanův blok J_j s vlastním číslem λ_j o velikosti n_j jsme nekonečný Taylorův rozvoj usekli a nahradili f polynomem

$$p_j(z) = \sum_{i=0}^{n_j-1} \frac{f^{(i)}(\lambda_j)}{i!}(z - \lambda_j)^i.$$

Tento polynom splňuje

$$p_j^{(i)}(\lambda_j) = f^{(i)}(\lambda_j), \quad i = 0, \ldots, n_j - 1$$

a je jednoznačně určeným **Hermiteových** interpolačním polynomem splňujícím výše uvedené podmínky.

7.3 Hermiteova interpolace

Úloha (Hermiteova interpolace)**.** Nechť jsou dány obecně komplexní hodnoty

$$f_{i,j}, \qquad i = 0, \dots, n, \quad j = 0, \dots, \alpha_i - 1,$$

kde $\alpha_i \in \mathbb{N}$, $\alpha \geq 1$, a označme

$$m \equiv \sum_{i=0}^{n} \alpha_i - 1.$$

Nechť dále $x_i \in \mathbb{C}$ jsou navzájem různé body. Hledáme polynom $h \in \mathcal{P}_m$, kde $\mathcal{P}_m$ označuje prostor polynomů stupně nejvýše m, takový, že

$$h^{(j)}(x_i) = f_{i,j}, \qquad i = 0, \dots, n, \quad j = 0, \dots, \alpha_i - 1.$$

Máme $m + 1$ rovnic pro $m + 1$ neznámých.

Věta 7.5. $\exists!\ h \in \mathcal{P}_m$ *řešící úlohu Hermiteovy interpolace.*

Důkaz. Hledejme $h \in \mathcal{P}_m$ ve tvaru

$$h(x) = \sum_{k=0}^{m} h_k \phi_k(x),$$

kde $\phi_0(x), \ldots, \phi_m(x)$ je zvolená báze prostoru $\mathcal{P}_m$.

Interpolační podmínky zapišme ve tvaru

$$\sum_{k=0}^{m} \phi_k^{(j)}(x_i) h_k = f_{i,j},$$

$i = 0, \ldots, n$, $j = 0, \ldots, \alpha_i - 1$.

Jde o systém $m+1$ lineárních rovnic o $m+1$ neznámých $h_0, \ldots, h_m$, s čtvercovou maticí mající prvky

$$\phi_k^{(j)}(x_i).$$

Tato matice je regulární. Pokud singulární, pak má systém s nulovou pravou stranou netriviální řešení, tj. existuje $p \in \mathcal{P}_m$, $p \neq 0$, který má $m+1$ kořenů (plyne z interpolačních podmínek) $\rightarrow$ spor. $\square$

7.4 Definice $f(A)$ (Hermite)

Definice 7.6. Nechť $A \in \mathbb{C}^{n\times n}$ a nechť $f^{(i)}(\mu_j)$ je definováno pro $j = 1, \ldots, \ell$ a $i = 0, \ldots, m_j - 1$. Pak maticovou funkci $f(A)$ definujeme jako

$$f(A) \equiv h(A),$$

kde $h \in \mathcal{P}_m$ je jednoznačně určený Hermiteův interpolační polynom splňující interpolační podmínky

$$h^{(i)}(\mu_j) = f^{(i)}(\mu_j),$$

$j = 1, \ldots, \ell$, $i = 0, \ldots, m_j - 1$.

7.5 Ekvivalence definic

Obě definice mají stejné předpoklady na hladkost f a za daných předpokladů jsou $f(A)$ dobře definovány.

Označme $f_1(A)$ a $f_2(A)$ matice definované pomocí Taylora (Definice 7.3) a Hermitea (Definice 7.6). Jelikož

$$f_1(A) = X f_1(J) X^{-1}, \quad f_2(A) = X h(J) X^{-1},$$

stačí ukázat $f_1(J) = h(J)$.

Pro každý Jordanův blok J_j platí

$$\begin{aligned} f_1(J_j) &= \begin{bmatrix} f(\lambda_j) & \frac{f^{(1)}(\lambda_j)}{1!} & \dots & \frac{f^{(n_j-1)}(\lambda_j)}{(n_j-1)!} \\ & f(\lambda_j) & \ddots & \vdots \\ & & \ddots & \frac{f^{(1)}(\lambda_j)}{1!} \\ & & & f(\lambda_j) \end{bmatrix} \\ &= \begin{bmatrix} h(\lambda_j) & \frac{h^{(1)}(\lambda_j)}{1!} & \dots & \frac{h^{(n_j-1)}(\lambda_j)}{(n_j-1)!} \\ & h(\lambda_j) & \ddots & \vdots \\ & & \ddots & \frac{h^{(1)}(\lambda_j)}{1!} \\ & & & h(\lambda_j) \end{bmatrix} = h(J_j). \end{aligned}$$

Nutně tedy platí

$$f_1(J) = h(J)$$

a tím je ekvivalence ukázána.

7.6 Definice $f(A)$ (Cauchy)

Definice pomocí Cauchyho integrálu.

Definice 7.7. Nechť f je analytická na otevřené množině Ω obsahující spektrum matice $A \in \mathbb{C}^{n\times n}$. Definujme

$$f(A) \equiv \frac{1}{2\pi \mathbf{i}} \int_{\Gamma} f(z)(zI - A)^{-1} \mathrm{dz},$$

kde Γ je jednoduchá uzavřená po částech hladká kladně orientovaná křivka v Ω, která striktně obklopuje všechna vlastní čísla A.

- Zobecnění Cauchyho integrální formule na matice.
- Za předpokladu, že f je analytická na Ω, všechny definice ekvivalentní, viz [10, str. 247, Theorem 6.2.28].
- Rezolventa matice

$$(zI - A)^{-1}.$$

8

Výpočet a aproximace

8.1 $f(A)$ a $f(A)v$ v aplikacích

- Solomonovy rovnice (2D nukleární magnetická rezonance)
$$\frac{\partial M}{\partial t} = -RM, \quad M(0) = I,$$
R je daná symetrická diagonálně dominantní matice. Pak
$$M(t) = e^{-Rt}.$$

- Lineární diferenciální rovnice
$$\frac{\partial y}{\partial t} = Ay, \quad y(0) = c,$$
kde $c \in \mathbb{C}^n$, $A \in \mathbb{C}^{n\times n}$. Pak
$$y(t) = e^{At}c.$$

- Jak počítat $f(A)$? Dle definice?
$$f(A) = Xf(J)X^{-1}.$$
Výpočetně náročné, může být numericky nestabilní.

8.2 Využití spektrální informace

Normální matice (spektrální rozklad)

- Nechť A diagonalizovatelná

$$A = X\Lambda X^{-1} \quad \Rightarrow \quad f(A) = X f(\Lambda) X^{-1}.$$

- Výpočet nákladný, QR algoritmus spočte vlastní čísla, pak pomocí inverzních iterací vlastní vektory.

- Stabilita výpočtu, pro jednoduchost X a Λ přesné, chyba jen ve vyhodnocení f, spočtu

$$\widehat{f(A)} = X(f(\Lambda) + E)X^{-1} = f(A) + \underbrace{XEX^{-1}}_{\substack{\text{chyba} \\ \text{výpočtu}}}.$$

- Tedy

$$\|f(A) - \widehat{f(A)}\| \leq \kappa(X)\|E\|.$$

- Vhodné pro matice s malým $\kappa(X) \to$ normální, tj. hermitovské, unitární atd.

Obecné matice (Schurův rozklad)

- Potřebujeme unitární podobnostní transfomaci $\rightarrow$ Schurův rozklad,

$$A = QRQ^*,$$

  R horní $\triangle$ s $\lambda_1, \ldots, \lambda_n$ na diagonále, $Q^*Q = I$.

- R lze spočíst zpětně stabilním QR algoritmem a

$$f(A) = Qf(R)Q^*.$$

- Zřejmé $\rightarrow$ $f(R)$ je horní $\triangle$ s $f(\lambda_i)$ na diagonále.

- Jak spočíst $F \equiv f(R)$? Platí

$$FR = RF,$$

  známe R, diagonálu F.

Věta 8.1. *Nechť $R \in \mathbb{C}^{n\times n}$ je horní trojúhelníková s navzájem různými vlastními čísly λ_i na diagonále a nechť funkce f je definována na λ_i, $i = 1, \ldots, n$. Označme*

$$R = [r_{i,j}]_{i,j=1}^{n} \quad a \quad f(R) = F = [f_{i,j}]_{i,j=1}^{n},$$

kde $f_{i,j} = 0$ pro $i > j$ a $f_{i,i} = f(\lambda_i)$. Pak pro $i < j$ platí

$$f_{i,j} = r_{i,j}\frac{f_{i,i} - f_{j,j}}{\lambda_i - \lambda_j} + \sum_{k=i+1}^{j-1} \frac{f_{i,k}r_{k,j} - r_{i,k}f_{k,j}}{\lambda_i - \lambda_j}.$$

Důkaz. Srovnáním prvků na pozici (i, j) z $FR = RF$

$$\sum_{k=i}^{j} r_{i,k}f_{k,j} = \sum_{k=i}^{j} f_{i,k}r_{k,j},$$

dostáváme

$$f_{i,j}(r_{i,i}-r_{j,j}) = r_{i,j}(f_{i,i}-f_{j,j}) + \sum_{k=i+1}^{j-1} (f_{i,k}r_{k,j} - r_{i,k}f_{k,j}).$$

Je-li $\lambda_i = r_{i,i} \neq r_{j,j} = \lambda_j$, pak lze vydělit $(r_{i,i} - r_{j,j})$. □

- F lze dopočítávat po sloupcích v pořadí

  $$(1,2) \to (2,3), (1,3) \to (3,4), (2,4), (1,4) \to \ldots.$$

- V obecném případě je nutné uvažovat bloky s navzájem různými λ_i, viz [8, Chapter 9].

- Výpočet $f(A)$ přes Schurův rozklad (Schur-Parlett algoritmus) je stabilní, ale drahý $O(n^3)$.

Příklad 8.2. Dáno R a f,

$$R = \begin{bmatrix} \lambda_1 & r_{1,2} \\ & \lambda_2 \end{bmatrix}, \quad F = \begin{bmatrix} f(\lambda_1) & \bullet \\ & f(\lambda_2) \end{bmatrix}.$$

Je-li $r_{1,2} = 0$, pak vše jasné. Nechť $r_{1,2} \neq 0$.

- Je-li $\lambda_1 \neq \lambda_2$, pak $FR = RF$ dává

$$f_{1,2} = r_{1,2} \frac{f_{1,1} - f_{2,2}}{\lambda_1 - \lambda_2}.$$

- Je-li $\lambda_1 = \lambda_2$ pak $f_{1,2} = r_{1,2} f'(\lambda_1)$, neboť

$$R = \begin{bmatrix} \lambda & r \\ & \lambda \end{bmatrix} = \begin{bmatrix} 1 & \\ & r^{-1} \end{bmatrix} \begin{bmatrix} \lambda & 1 \\ & \lambda \end{bmatrix} \begin{bmatrix} 1 & \\ & r \end{bmatrix}$$

a

$$\begin{aligned} f(R) &= \begin{bmatrix} 1 & \\ & r^{-1} \end{bmatrix} f\left(\begin{bmatrix} \lambda & 1 \\ & \lambda \end{bmatrix} \right) \begin{bmatrix} 1 & \\ & r \end{bmatrix} \\ &= \begin{bmatrix} 1 & \\ & r^{-1} \end{bmatrix} \begin{bmatrix} f(\lambda) & f'(\lambda) \\ & f(\lambda) \end{bmatrix} \begin{bmatrix} 1 & \\ & r \end{bmatrix} \\ &= \begin{bmatrix} f(\lambda) & r f'(\lambda) \\ & f(\lambda) \end{bmatrix}. \end{aligned}$$

8.3 Polynomiální a racionální ≈

Využití Taylorova rozvoje

Nechť f analytická,

$$f(z) = \sum_{i=0}^{\infty} a_i (z-\alpha)^i, \quad a_i \equiv \frac{f^{(i)}(\alpha)}{i!},$$

α je střed. Nemusí konvergovat $\forall z$, jen pro z :

$$|z-\alpha| < r,$$

r nazýváme poloměr konvergence. Kdy platí

$$\underbrace{f(A)}_{\substack{\text{žije na}\\ \text{spektru}}} = \underbrace{\sum_{i=0}^{\infty} a_i (A-\alpha I)^i ?}_{\text{konverguje?}}$$

Věta 8.3. *Nechť f je analytická na okolí bodu $\alpha \in \mathbb{C}$ s poloměrem $r > 0$. Je-li*

$$|\mu_j - \alpha| < r, \quad j = 1, \ldots, \ell,$$

pak

$$f(A) = \sum_{i=0}^{\infty} a_i (A-\alpha I)^i.$$

Důkaz. (dobrovolný) Pro $\lambda \in \mathbb{C}$, $|\lambda - \alpha| < r$, označme

$$f_N(\lambda) \equiv \sum_{i=0}^{N} a_i (\lambda - \alpha)^i.$$

Větu stačí dokázat pro Jordanův blok B velikosti m s vlastním číslem λ. Uvědomme si, že

$$B - \alpha I = (\lambda - \alpha)\, I + E,$$

E je Jordanův blok s 0 na diagonále, $E^m = 0$. Platí

$$\begin{aligned}
\sum_{j=0}^{N} a_j \left((\lambda - \alpha)\, I + E\right)^j &= \sum_{j=0}^{N} a_j \sum_{i=0}^{j} \binom{j}{i} (\lambda - \alpha)^{j-i} E^i \\
&= \sum_{i=0}^{N} \sum_{j=i}^{N} a_j \binom{j}{i} (\lambda - \alpha)^{j-i} E^i \\
&= \sum_{i=0}^{N} \frac{E^i}{i!} \sum_{j=i}^{N} a_j \left[\frac{j!(\lambda - \alpha)^{j-i}}{(j-i)!}\right] \\
&= \sum_{i=0}^{N} \frac{E^i}{i!} \sum_{j=i}^{N} a_j \left[\frac{d^i}{d\lambda^i}(\lambda - \alpha)^j\right] \\
&= \sum_{i=0}^{N} \frac{E^i}{i!} \left[\frac{d^i}{d\lambda^i} \sum_{j=i}^{N} a_j (\lambda - \alpha)^j\right] \\
&= \sum_{i=0}^{N} \frac{E^i}{i!} \frac{d^i}{d\lambda^i} f_N(\lambda)
\end{aligned}$$

a pro $N \geq m-1$ je

$$\sum_{j=0}^{N} a_j \left(B - \alpha I\right)^j = \sum_{i=0}^{m-1} \frac{E^i}{i!} \frac{d^i}{d\lambda^i} f_N(\lambda).$$

Pro $N \to \infty$ pak dostáváme

$$\lim_{N\to\infty} \frac{d^i}{d\lambda^i} f_N(\lambda) = f^{(i)}(\lambda)$$

a tudíž

$$\sum_{j=0}^{\infty} a_j \left(B - \alpha I\right)^j = \sum_{i=0}^{m-1} \frac{E^i}{i!} f^{(i)}(\lambda) = f(B)$$

dle definice 1 maticové funkce. □

Za předpokladů předchozí věty lze aproximovat

$$f(A) \approx f_N(A),$$

N buď známe nebo iteračně zvětšujeme.

Příklad.

$$\begin{aligned}
\exp A) &= \sum_{i=0}^{\infty} \frac{A^i}{i!}, \quad \alpha = 0, \quad r = \infty, \\
\cos(A) &= \sum_{i=0}^{\infty} (-1)^i \frac{A^{2i}}{(2i)!}, \quad \alpha = 0, \quad r = \infty, \\
\log(I - A) &= \sum_{i=1}^{\infty} \frac{A^i}{i}, \quad \rho(A) < 1.
\end{aligned}$$

Využití interpolantů

- Obecné interpolační polynomy $\to f$ nahradíme interpolantem p a spočteme $p(A)$. Ano, ale p může oscilovat, problémy se stabilitou.
- Čebyševovy interpolanty konvergují stejnoměrně k f na reálné ose, je-li f Lipschitzovsky spojitá. Vhodné, má-li A reálné spektrum.

Racionální aproximace

- Racionální aproximace f

$$f(z) \approx r(z) = \frac{p(z)}{q(z)},$$

pak

$$f(A) \approx r(A) = p(A)\,(q(A))^{-1}.$$

- Často p a q stejného stupně, $r(z)$ volíme pomocí interpolace či Padé aproximace.

8.4 Maticové iterace

- X_0 počáteční aproximace, stacionární iterace
$$X_{k+1} = g(X_k),$$
g je vhodná (iterační) funkce, nezávisí na k.
- g je voleno tak, aby $f(A)$ byla pevným bodem g,
$$X_* = f(A), \quad g(X_*) = X_*,$$
tj. volba g závisí f.
- Postupy vycházející z Newtonovy metody.
- Konverguje proces? Jak rychle?

Příklad 8.4. Newton pro $\sqrt{a}$, $a > 0$, řešíme
$$f(x) = x^2 - a = 0$$
pomocí Newtona
$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)} = \frac{1}{2}\left(x_k + \frac{a}{x_k}\right).$$
Pro $A \in \mathbb{C}^{n\times n}$ s $\mathrm{Re}(\lambda_i) > 0$ lze použít Newtonovy iterace
$$X_{k+1} = \frac{1}{2}\left(X_k + X_k^{-1}A\right).$$
Lokálně kvadratická konvergence, zastavujeme pokud
$$\frac{\|X_{k+1} - X_k\|}{\|X_k\|} < \mathrm{tol}.$$
Stagnační kritérium, vhodné pro stacionární metody.

8.5 Odhad normy $f(A)$

- V některých aplikacích stačí odhad $\|f(A)\|$.
- A normální, $A = U\Lambda U^*$

$$\|f(A)\| = \|f(\Lambda)\| = \max_{i=1,\dots,n} |f(\lambda_i)|.$$

- A obecná diagonalizovatelná, $A = X\Lambda X^{-1}$,

$$\|f(A)\| = \|Xf(\Lambda)X^{-1}\| \leq \kappa(X) \max_{i=1,\dots,n} |f(\lambda_i)|,$$

  problém je-li $\kappa(X) \gg 1$.

- Další přístupy, kde v $\mathbb{C}$ matice A žije?

$$\|f(A)\| \leq C \max_{z\in\Omega\subset\mathbb{C}} |f(z)|,$$

  C je konstanta a Ω =? Otevřený problém.

Příklad 8.5. Pole hodnot matice A,

$$\Omega = W(A) = \{v^*Av : \ \|v\| = 1, \ v \in \mathbb{C}^n\}$$

je konvexní uzavřená množina v $\mathbb{C}$ definovaná pomocí A. Crouzeix conjecture (2006)

$$\|f(A)\| \leq 2 \max_{z\in W(A)} |f(z)|,$$

dokázáno s konstantou $1 + \sqrt{2}$, viz [1].

8.6 Aproximace $f(A)v$

- K aproximaci $f(A)v$ lze použít $\mathcal{K}_k(A, v)$.
- Arnoldi (Lanczos) konstruuje ortogonální bázi

$$\mathcal{K}_k(A, v) = \text{span}\{v_1, \ldots, v_k\},$$

  kde $k \leq d(A, v) \equiv d$.
- Bez újmy na obecnosti $\|v\| = 1$, tj. $v_1 = v$.
- Platí

$$AV_k = V_k H_k + h_{k+1,k} v_{k+1} e_k^T,$$

  H_k horní Hessenbergova či tridiagonální ($A = A^*$).
- Jak hledat aproximaci $f(A)v$ v $\mathcal{K}_k(A, v)$?

Následující lemma ukazuje, že kdyby f byl polynom, $\deg(f) < k$, pak bychom byli schopni spočíst $f(A)v$ po k iteracích Arnoldiho algoritmu pomocí $f(H_k)$.

Lemma 8.6. *Nechť $k \leq d$. Potom $\forall p \in \mathcal{P}_{k-1}$ platí*

$$p(A)v = V_k p(H_k) e_1.$$

Je-li $k = d$, pak $p(A)v = V_k p(H_k) e_1$ pro každý polynom p.

Důkaz. Cvičení či [8, Lemma 13.4]: Stačí ukázat, že vztah platí pro mocniny z^j, tj. $p(z) = z^j$, $j = 0, \ldots, k-1$.

Pro $j = 0$ vztah platí neboť $v = v_1 = V_k e_1$. Nechť vztah platí pro $j \leq k-2$. Potom

$$\begin{aligned} A^{j+1}v &= A\left(A^j v\right) \\ &= A\left(V_k H_k^j e_1\right) = \left(AV_k\right) H_k^j e_1 \\ &= (V_k H_k + h_{k+1,k} v_{k+1} e_k^T) H_k^j e_1 \\ &= V_k H_k^{j+1} e_1 + h_{k+1,k} v_{k+1} e_k^T H_k^j e_1 \\ &= V_k H_k^{j+1} e_1 \end{aligned}$$

neboť $e_k^T H_k^j e_1 = 0$ pro $j < k-1$.

Pro $k = d$ má $\mathcal{K}_k(A, v)$ maximální dimenzi a platí $AV_d = V_d H_d$. Odsud $A^i V_d = V_d H_d^i$ $\forall i \geq 0$ a proto

$$p(A)v = V_d p(H_d) e_1$$

$\forall$ polynom p. □

Dále ukazujeme, že pokud zvolíme p v předchozím lemmatu vhodným způsobem (pomocí Hermiteovy interpolace f na spektru H_k), pak $p(A)v = V_k f(H_k)e_1$.

Lemma 8.7. *Nechť $k \leq d$. Potom platí*

$$V_k f(H_k)e_1 = p_{k-1}(A)v, \tag{8.1}$$

kde p_{k-1} je jednoznačný Hermiteův interpolační polynom interpolující f na spektru H_k.

Důkaz. Nechť p_{k-1} je jednoznačný Hermiteův interpolační polynom interpolující f na spektru H_k. Potom dle Definice 7.6 maticové funkce platí

$$f(H_k) = p_{k-1}(H_k).$$

Jelikož H_k je řádu k, je $\deg(p_{k-1}) \leq k-1$ a dle Lemma 8.6

$$p_{k-1}(A)v = V_k p_{k-1}(H_k)e_1 = V_k f(H_k)e_1.$$

□

Jelikož lze na $f(A)$ nahlížet jako na $h(A)$, kde h je Hermiteův interpolační polynom interpolující f na spektru A, nabízí se aproximovat

$$f(A)v = h(A)v \approx p_{k-1}(A)v = V_k f(H_k)e_1,$$

tedy

$$f(A)v \approx V_k f(H_k)e_1.$$

Další podpůrný argument pro tento způsob aproximace poskytuje následující věta.

Věta 8.8. *Dána $A \in \mathbb{C}^{n\times n}$ a $v \in \mathbb{C}^n$, $\|v\| = 1$. Označme $d = d(A, v)$. Dále nechť je dána f splňující předpoklady definic 7.3 a 7.6, tj. $f(A)$ je dobře definována. Potom*

$$f(A)v = V_d f(H_d)e_1.$$

Důkaz. Pro $k = d$ je $p(A)v = V_d p(H_d)e_1$ pro každý polynom p (Lemma 8.6), tedy i Hermiteův interpolační polynom h interpolující f na spektru A, a platí

$$f(A)v = h(A)v = V_d h(H_d)e_1. \tag{8.2}$$

Jordanův tvar matice H_d obsahuje stejné nebo menší Jordanovy bloky jako Jordanův tvar A. Tudíž h splňuje na spektru H_d stejné interpolační podmínky jako p_{d-1} a dle Věty 7.2

$$h(H_d) = p_{d-1}(H_d). \tag{8.3}$$

Celkem tedy dostáváme

$$f(A)v \overset{(8.2)}{=} V_d h(H_d)e_1 \overset{(8.3)}{=} V_d p_{d-1}(H_d)e_1 \overset{(8.1)}{=} V_d f(H_d)e_1.$$

□

Dává tedy dobrý smysl aproximovat

$$f(A)v \approx V_k f(H_k)e_1$$

s tím, že pro $k = d$ dostáváme rovnost.

Část V

FFT

9

Rychlá Fourierova transformace

9.1 Spojitá transformace

- Spojitá Fourierova transformace převádí funkce do frekveční báze (znají fyzikové).
- Dána $f : \mathbb{R} \to \mathbb{C}$, spojitá integrovatelná.
- Fourierova transformace funkce f:

$$\hat{f}(\xi) = \int_{-\infty}^{\infty} f(x) e^{-(2\boldsymbol{i}\pi x)\xi} \mathrm{d}x.$$

- Inverzní transformace

$$f(x) = \int_{-\infty}^{\infty} \hat{f}(\xi) e^{(2\boldsymbol{i}\pi x)\xi} \mathrm{d}\xi,$$

  předpoklad $\hat{f}(\xi)$ integrovatelná, $x \in \mathbb{R}$.

9.2 Diskrétní transformace

- Diskrétní je analogie spojité. Dáno $x \in \mathbb{C}^N$,

$$x = \begin{bmatrix} x_0 \\ \vdots \\ x_{N-1} \end{bmatrix} \quad \xrightarrow{\mathcal{F}} \quad \begin{bmatrix} y_0 \\ \vdots \\ y_{N-1} \end{bmatrix} = y.$$

  Vektor y s prvky

$$y_k = \sum_{j=0}^{N-1} e^{-\frac{2\pi \boldsymbol{i}}{N}(kj)} x_j, \qquad k = 0, \ldots, N-1$$

  nazveme diskrétní Fourierovou transformací (DFT) vektoru x. Přitom

$$e^{-\frac{2\pi \boldsymbol{i}}{N}(kj)}$$

  reprezentuje diskrétní hodnoty funkce $e^{-\frac{2\pi \boldsymbol{i} k}{N} z}$ na N bodech, $z = 0, \ldots, N-1$, kde $\frac{k}{N}$ určuje frekvenci.

- Inverzní transformace $\mathcal{F}^{-1}(y) = x$, kde

$$x_j = \frac{1}{N} \sum_{k=0}^{N-1} e^{\frac{2\pi \boldsymbol{i}}{N}(kj)} y_k, \qquad j = 0, \ldots, N-1$$

- Definujme

$$\omega_N = e^{-\frac{2\pi \boldsymbol{i}}{N}}, \qquad e^{-\boldsymbol{i}\theta} = \cos\theta - \boldsymbol{i} \sin\theta.$$

- DFT matice F_N s prvky

$$f_{k,j} = \omega_N^{(k-1)(j-1)}$$

$$F_N = \begin{bmatrix} \omega_N^0 & \omega_N^0 & \dots & \dots & \omega_N^0 \\ \omega_N^{1\cdot 0} & \omega_N^1 & \omega_N^2 & \dots & \omega_N^{N-1} \\ \omega_N^{2\cdot 0} & \omega_N^2 & \omega_N^4 & \dots & \omega_N^{2(N-1)} \\ \vdots & \vdots & & & \vdots \\ \omega_N^0 & \omega_N^{N-1} & \dots & \dots & \omega_N^{(N-1)(N-1)} \end{bmatrix}.$$

  Komplexní symetrická (ne Hermitovská), hodnoty se opakují.

- Diskrétní Fourierova transformace

$$\mathcal{F}(x) = F_N x = y$$

  Inverzní DFT

$$\mathcal{F}^{-1}(y) = F_N^{-1} y = x$$

- Platí [4, Section 4.8.1]

$$F_N^{-1} = \frac{1}{N} F_N^* = \frac{1}{N} \overline{F}_N,$$

  je unitární až na škálování ($\frac{1}{\sqrt{N}} F_N$ je unitární).

9.3 Aproximace periodické f

$f \in C_{2\pi}$ lze aproximovat (nejlepší L_2 aproximace) pomocí částečného součtu Fourierovy řady

$$f \approx S_n f = \frac{1}{2}a_0 + \sum_{j=1}^{n}(a_j \cos(jx) + b_j \sin(jx)),$$

$$a_j = \frac{1}{\pi}\int_{-\pi}^{\pi} f(\theta)\cos(j\theta)\mathrm{d}\theta, \quad b_j = \frac{1}{\pi}\int_{-\pi}^{\pi} f(\theta)\sin(j\theta)\mathrm{d}\theta.$$

Za určitých předpokladů (omezená variace či absolutní spojitost) $S_n f \rightrightarrows f$.

- Jak realizovat výpočet $S_n f$ numericky?
- Zvol N a aproximuj integrály Riemannovými součty,

$$\frac{1}{\pi}\int_0^{2\pi} g(\theta)\mathrm{d}\theta \approx \frac{1}{\pi}\sum_{k=0}^{N-1} g\left(\frac{2k\pi}{N}\right)\frac{2\pi}{N},$$

tedy

$$\begin{aligned} a_j &\approx \frac{2}{N}\sum_{k=0}^{N-1} f\left(\frac{2k\pi}{N}\right)\cos\left(\frac{2jk\pi}{N}\right) \equiv \alpha_j, \\ b_j &\approx \frac{2}{N}\sum_{k=0}^{N-1} f\left(\frac{2k\pi}{N}\right)\sin\left(\frac{2jk\pi}{N}\right) \equiv \beta_j. \end{aligned}$$

- Vektorově, definuj $\vec{f}$, $\vec{c}_j$, $\vec{s}_j$,

$$\vec{f} \equiv \begin{bmatrix} f\left(\frac{2\cdot 0\cdot\pi}{N}\right) \\ f\left(\frac{2\cdot 1\cdot\pi}{N}\right) \\ \vdots \\ f\left(\frac{2\cdot(N-1)\cdot\pi}{N}\right) \end{bmatrix}, \quad \vec{c}_j \equiv \begin{bmatrix} \cos\left(\frac{2\cdot 0\cdot j\cdot\pi}{N}\right) \\ \cos\left(\frac{2\cdot 1\cdot j\cdot\pi}{N}\right) \\ \vdots \\ \cos\left(\frac{2\cdot(N-1)\cdot j\cdot\pi}{N}\right) \end{bmatrix},$$

pak $\alpha_j = \frac{2}{N}\langle\vec{c}_j, \vec{f}\rangle$, $\beta_j = \frac{2}{N}\langle\vec{s}_j, \vec{f}\rangle$, tj.

$$\begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_n \end{bmatrix} = \frac{2}{N}\begin{bmatrix} \langle\vec{c}_0, \vec{f}\rangle \\ \langle\vec{c}_1, \vec{f}\rangle \\ \vdots \\ \langle\vec{c}_n, \vec{f}\rangle \end{bmatrix} \equiv \frac{2}{N} C_{n,N}\vec{f},$$

$$\begin{bmatrix} 0 \\ \beta_1 \\ \vdots \\ \beta_n \end{bmatrix} = \frac{2}{N}\begin{bmatrix} \langle\vec{s}_0, \vec{f}\rangle \\ \langle\vec{s}_1, \vec{f}\rangle \\ \vdots \\ \langle\vec{s}_n, \vec{f}\rangle \end{bmatrix} \equiv \frac{2}{N} S_{n,N}\vec{f},$$

kde $C_{n,N}$ resp. $S_{n,N} \in \mathbb{R}^{(n+1)\times N}$ má prvky

$$\cos\left(\frac{2\cdot j\cdot k\cdot\pi}{N}\right), \text{ resp. } \sin\left(\frac{2\cdot j\cdot k\cdot\pi}{N}\right),$$

$j = 0, \ldots, n$, $k = 0, \ldots, N-1$.

- Je-li $N > 2n$, vektory $\vec{c}_0, \vec{c}_1, \ldots, \vec{c}_n$, $\vec{s}_1, \ldots, \vec{s}_n$ jsou lineárně nezávislé a vzájemně ortogonální.

- Celkem

$$\begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_n \end{bmatrix} - \mathbf{i} \begin{bmatrix} 0 \\ \beta_1 \\ \vdots \\ \beta_n \end{bmatrix} = \frac{2}{N} E_{n,N} \vec{f},$$

  kde $E_{n,N} \in \mathbb{R}^{(n+1)\times N}$ má prvky

$$\exp\left(-\frac{2 \cdot \mathbf{i} \cdot j \cdot k \cdot \pi}{N}\right) = \omega_N^{jk},$$

  tj. $E_{n,N}$ obsahuje $n+1$ řádků matice F_N!

- Stačí spočíst

$$\frac{2}{N} F_N \vec{f}$$

  a z reálné a imaginární části prvních $n+1$ složek výsledného vektoru pak zjistit α_j a β_j.

- Závěr: Z $\vec{f}$ lze spočíst aproximace Fourierových koeficientů pomocí DFT. Pomocí inverzní transformace, z aproximací Fourierových koeficientů umíme spočíst hodnoty f.

9.4 Algoritmus FFT

- Pro jednoduchost, nechť N je mocnina dvojky. Jak efektivně realizovat

$$y = F_N x?$$

- Idea: Využití periodicity sinu a cosinu. Vyjádři například α_j a $\alpha_{\frac{N}{2}-j}$ z předchozího

$$\begin{aligned}\alpha_j &= \frac{2}{N}\sum_{k=0}^{N-1} f\left(\frac{2k\pi}{N}\right)\cos\left(\frac{2jk\pi}{N}\right)\\ &= \frac{2}{N}\sum_{k\ \text{sudé}}^{N-1} + \frac{2}{N}\sum_{k\ \text{liché}}^{N-1},\end{aligned}$$

$$\begin{aligned}\alpha_{\frac{N}{2}-j} &= \frac{2}{N}\sum_{k=0}^{N-1} f\left(\frac{2k\pi}{N}\right)\cos\left(\frac{2\left(\frac{N}{2}-j\right)k\pi}{N}\right)\\ &= \frac{2}{N}\sum_{k=0}^{N-1} f\left(\frac{2k\pi}{N}\right)(-1)^k\cos\left(\frac{2jk\pi}{N}\right)\\ &= \frac{2}{N}\sum_{k\ \text{sudé}}^{N-1} - \frac{2}{N}\sum_{k\ \text{liché}}^{N-1}.\end{aligned}$$

- Stačí přibližně polovina operací na spočtení téhož.

- Obecně, uvažujme F_N, N sudé a označme $m = N/2$. Definujme permutační vektor cols délky N, s lichými a sudými indexy, v matlabovské notaci

$$\text{cols} = [1:2:N, 2:2:N]^T.$$

- Použijeme-li $\omega_N^2 = \omega_m$, lze ukázat

$$F_N(:, \text{cols}) = \begin{bmatrix} F_m & \Omega_m F_m \\ F_m & -\Omega_m F_m \end{bmatrix},$$

kde

$$\Omega_m \quad = \quad \text{diag}(1, \omega_N, \omega_N^2, \ldots, \omega_N^{m-1}).$$

- Jelikož

$$y = F_N x = F_N(:, \text{cols}) x(\text{cols}),$$

platí

$$\begin{aligned} y &= \begin{bmatrix} F_m & \Omega_m F_m \\ F_m & -\Omega_m F_m \end{bmatrix} \begin{bmatrix} x_{1:2:N} \\ x_{2:2:N} \end{bmatrix} \\ &= \begin{bmatrix} I_m & \Omega_m \\ I_m & -\Omega_m \end{bmatrix} \begin{bmatrix} F_m x_{1:2:N} \\ F_m x_{2:2:N} \end{bmatrix}. \\ &= \begin{bmatrix} y_T + \Omega_m y_B \\ y_T - \Omega_m y_B \end{bmatrix}, \end{aligned}$$

kde jsme označili $y_T = F_m x_{1:2:N}$, $y_B = F_m x_{2:2:N}$.

- Aplikuj rekursivně, konec rekurse $F_1 x = x$.

Nechť $N = 2^t$ a $x \in \mathbb{C}^N$, volej

$$y = \texttt{fft}(x, N);$$

Algorithm 9.1 Rychlá Fourierova transformace (FFT)

function $y = \texttt{fft}(x,n)$

1: **if** $n = 1$ **then**
2: $y = x$
3: **else**
4: $m = n/2$
5: $y_T = \texttt{fft}(x_{1:2:n}, m)$
6: $y_B = \texttt{fft}(x_{2:2:n}, m)$
7: $\omega = \exp(-\frac{2\pi \mathbf{i}}{n})$
8: $d = [1, \omega, \ldots, \omega^{m-1}]$
9: $z = d. * y_B$
10: $y = \begin{bmatrix} y_T + z \\ y_T - z \end{bmatrix}$
11: **end if**

- FFT $\mathcal{O}(N \log N)$ operací, přesněji $8N \log_2(N)$
- Software FFTW (Fastest Fourier Transform in the West), napsáno v C, „Wilkinson Prize for Numerical Software“, Matteo Frigo a Steven G. Johnson, MIT.

10

Maticové výpočty a FFT

10.1 Cirkulantní matice

- Matice se strukturou

$$C = \begin{bmatrix} c_0 & c_{n-1} & \dots & c_1 \\ c_1 & c_0 & \dots & \vdots \\ \vdots & \vdots & \dots & c_{n-1} \\ c_{n-1} & c_{n-2} & \dots & c_0 \end{bmatrix}.$$

- Definujme „downshift“ permutační matici

$$D = \begin{bmatrix} 0 & \dots & 0 & 1 \\ 1 & \ddots & & 0 \\ & \ddots & 0 & \vdots \\ & & 1 & 0 \end{bmatrix}.$$

- Potom

$$C = c_0 D^0 + c_1 D^1 + \cdots + c_{n-1} D^{n-1} = p_c(D),$$

kde $c = Ce_1$ je vektor koeficientů.

- Lze ukázat [4, Lemma 4.8.1 a Theorem 4.8.2], že

$$D = F_n^{-1} \Lambda_n F_n, \quad \Lambda_n = \mathrm{diag}(1, \omega_n, \omega_n^2, \dots, \omega_n^{n-1})$$

a

$$C = F_n^{-1} p_c(\Lambda_n) F_n = F_n^{-1} \mathrm{diag}(F_n c) F_n.$$

- Důsledek: umíme rychle násobit C i C^{-1}.

10.2 Toeplitz a Hankel matice

- $T \in \mathbb{R}^{n\times n}$ je toeplitzovská

$$T = \begin{bmatrix} a & c & \ddots & \\ b & a & \ddots & \ddots \\ \ddots & \ddots & \ddots & c \\ & \ddots & b & a \end{bmatrix}$$

- H je hankelovská

$$H = \begin{bmatrix} & \iddots & b & a \\ \iddots & \iddots & a & c \\ b & \iddots & \iddots & \iddots \\ a & c & \iddots & \end{bmatrix}.$$

- Umíme rozšířit na cirkulatní, například Tv realizuji

$$C = \begin{bmatrix} T & * \\ * & * \end{bmatrix}, \quad C \begin{bmatrix} v \\ 0 \end{bmatrix} = \begin{bmatrix} Tv \\ * \end{bmatrix}.$$

- Circulat, Toeplitz, Hankel - efektivní výpočty
  - Av ... $\mathcal{O}(n \log n)$ operací ($\mathcal{O}(n^2)$ obecná)
  - $A + B$ ... $\mathcal{O}(n)$ operací ($\mathcal{O}(n^2)$ obecná)
  - A^{-1}, $Ax = b$ ... $\mathcal{O}(n^2)$ operací ($\mathcal{O}(n^3)$ obecná)

10.3 Struktura DFT matice

- Označme S a $C \in \mathbb{R}^{(m-1)\times(m-1)}$ matice s prvky

$$\sin\left(\frac{jk\pi}{m}\right) \text{ a } \cos\left(\frac{jk\pi}{m}\right),\ j,k = 1,\ldots,m-1,$$

  Nechť $e = \texttt{ones}(m-1,1)$,

$$e_\pm = [(-1)^1,(-1)^2,\ldots,(-1)^{m-1}]^T,$$

$$E = \begin{bmatrix} & & 1 \\ & \iddots & \\ 1 & & \end{bmatrix}.$$

- Potom [4, p. 37]

$$F_{2m} = \begin{bmatrix} 1 & e^T & 1 & e^T \\ e & C - \mathbf{i}S & e_\pm & (C+\mathbf{i}S)E \\ 1 & e_\pm^T & (-1)^m & e_\pm^T E \\ e & E(C+\mathbf{i}S) & Ee_\pm & E(C-\mathbf{i}S)E \end{bmatrix}.$$

- DST: $y = Sx$, $x \in \mathbb{R}^{m-1}$ lze zrealizovat takto

$$x_{\sin} = \begin{bmatrix} 0 \\ x \\ 0 \\ -Ex \end{bmatrix} \Rightarrow \frac{\mathbf{i}}{2} F_{2m} x_{\sin} = \begin{bmatrix} 0 \\ Sx \\ 0 \\ -ESx \end{bmatrix}.$$

- Podobně DCT.

10.4 Fast Poisson solver

- Viz [4, p. 226], 2D Poisson, obdélníková Ω, Dirichlet, konečné diference, rovnoměrné dělení ve směru x a y, n vnitřních bodů ve směru x a m ve směru y.
- Vede na $Ax = b$, bloková matice $A \in \mathbb{R}^{nm \times nm}$,

$$A = \begin{bmatrix} T_n & -I_n & & \\ -I_n & \ddots & \ddots & \\ & \ddots & \ddots & -I_n \\ & & -I_n & T_n \end{bmatrix},$$

  $I_n \in \mathbb{R}^{n \times n}$, $T_n \in \mathbb{R}^{n \times n}$, $T_n = \text{tridiag}(-1, 4, -1)$.
- Připomeňme Kroneckerův součin matic

$$C \otimes D = \begin{bmatrix} c_{1,1}D & \dots & c_{1,n}D \\ \vdots & & \vdots \\ c_{n,1}D & \dots & c_{n,n}D \end{bmatrix}.$$

- Potom

$$A = (I_m \otimes P_n) + (P_m \otimes I_n),$$

  kde $P_n = \text{tridiag}(-1, 2, -1) \in \mathbb{R}^{n \times n}$.
- Je známo, že

$$P_n = S_n^{-1} \Lambda_n S_n,$$

  $[S_n]_{j,k} = \sin\left(\frac{jk\pi}{n+1}\right)$, $j, k = 1, \dots, n$ a Λ_n diagonální,

$$\lambda_j^{(n)} = 2 - 2\cos\left(\frac{j\pi}{n+1}\right).$$

- Blok $X = \texttt{reshape}(x, n, m)$, vektor $x = \texttt{vec}(X)$.
- Pro F a G regulární použij kalkulus [4, p. 28]

$$FXG = H \;\Leftrightarrow\; (G^T \otimes F)\texttt{vec}(X) = \texttt{vec}(H).$$

- Definuj $B = \texttt{reshape}(b, n, m)$. Pak $Ax = b \Leftrightarrow$

$$\left[(I_m \otimes P_n) + (P_m \otimes I_n)\right] \texttt{vec}(X) = \texttt{vec}(B)$$

  a užitím pravidel kalkulu

$$P_n X + X P_m = B.$$

- Použití spektrálního rozkladu P_n a P_m

$$\left(S_n^{-1}\Lambda_n S_n\right) X + X \left(S_m^{-1}\Lambda_m S_m\right) = B$$

  dává

$$\Lambda_n \underbrace{S_n X S_m^{-1}}_{\widetilde{X}} + \underbrace{S_n X S_m^{-1}}_{\widetilde{X}} \Lambda_m = \underbrace{S_n B S_m^{-1}}_{\widetilde{B}}.$$

- Jelikož jsou Λ_n i Λ_m známé a diagonální, dostáváme srovnáním prvků na pravé a levé straně

$$\widetilde{x}_{i,j} = \frac{\widetilde{b}_{i,j}}{\lambda_i^{(n)} + \lambda_j^{(m)}}.$$

- Známe-li $\widetilde{X}$, pak $X = S_n^{-1}\widetilde{X}S_m$.

- Algoritmus řešení $Ax = b$ pomocí FFT:
  - $B = \texttt{reshape}(b, n, m)$
  - spočti $\widetilde{B} = S_n B S_m^{-1}$ pomocí FFT
  - spočti $\widetilde{X}$
  - spočti $X = S_n^{-1} \widetilde{X} S_m$ pomocí FFT
  - $x = \texttt{vec}(X)$

Část VI

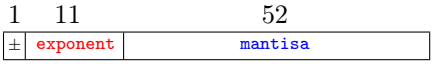
Analýza výpočtů

11

Zpětná stabilita

11.1 Reprezentace čísel v počítači

`double precision (64 bitů)`



- `znaménko` (sign) - 1 bit
- `exponent` 11 bitů - zakóduji čísla

$$-1023, \underbrace{-1022}_{e_{\min}}, \ldots, 0, \ldots, \underbrace{1023}_{e_{\max}}, 1024$$

- `mantisa` - binární číslo například 01110011011
- reprezentuje číslo v normalizovaném tvaru

$$\pm 1, \texttt{mantisa} \cdot 2^e$$

je-li $e_{\min} \leq e \leq e_{\max}$.

- reprezentace speciálních hodnot

exponent	mantisa	hodnota
$e_{\min} - 1$	0	0
$e_{\min} - 1$	$\neq 0$	$\pm 0, \texttt{mantisa} \cdot 2^e$
$e_{\max} + 1$	0	$\pm\infty$
$e_{\max} + 1$	$\neq 0$	`NaN` (not-a-number)

- Nejmenší a největší vyjádřitelné kladné číslo

$$x_{\min} \approx 2,2 \cdot 10^{-308}, \quad x_{\max} \approx 1,8 \cdot 10^{308}$$

`realmin` a `realmax` v Matlabu.

- V $[2^e, 2^{e+1}]$ jsou čísla

$$\pm 1, \texttt{mantisa} \cdot 2^e$$

rovnoměrně rozložena, vzdálenost $2^{e-52} \Rightarrow$ vzdálenost mezi 1 a nejbližším větším číslem je $2^{-52} \approx 2,2 \cdot 10^{-16} \rightarrow$ strojové epsilon $\varepsilon_{\text{mach}}$. Obecně

$$\varepsilon_{\text{mach}} = 2^{-t},$$

kde t je počet bitů mantisy.

- Označení $\mathbb{F}$ (floating point numbers).

Double precision (`binary64`) nejpoužívanější. Standardní procesory podporují ještě single precision (`binary32`) ($e = 8, t = 23$). Dříve různé aritmetiky, dnes zafixováno standardem IEEE 754 (Institute of Electrical and Electronic Engineers) dvojková (binární) aritmetika s pohyblivou řádovou čárkou[1]. V souvislosti s AI a výpočty na grafických kartách další (nestandardní) formáty[2].

[1] https://www.siam.org/publications/siam-news/articles/a-multiprecision-world/

[2] https://www.siam.org/publications/siam-news/articles/mixed-precision-computing-high-accuracy-with-low-precision/

11.2 Počítačová aritmetika

Standard IEEE 754 garantuje

- **round-to-nearest** strategy, označení

  $$\text{round}: \ x \in \mathbb{R} \to \bar{x} \in \mathbb{F},$$

  $x_{\min} \leq |x| \leq x_{\max}$, kde $\bar{x}$ je nejbližší číslo k x z $\mathbb{F}$,

  $$\text{round}(x) = x(1+\delta), \qquad |\delta| \leq \mathbf{u},$$

  $\mathbf{u}$ je zaokrouhlovací jednotka (strojová přesnost),

  $$\mathbf{u} = \frac{1}{2}2^{-t} = \frac{1}{2}\varepsilon_{\text{mach}},$$

  kde t je počet bitů mantisy, double $\mathbf{u} \approx 1,1 \cdot 10^{-16}$.

- **standardní model** aritmetiky s konečnou přesností (FPA - floating point arithmetic) předpokládá

  $$\text{fl}(x \ \circ \ y) = (x \ \circ \ y)(1+\delta), \qquad |\delta| \leq \mathbf{u},$$

  pro $x \in \mathbb{F}$ a $y \in \mathbb{F}$, kde $\circ \to$ operace $+, -, *, /, \sqrt{}$
  označení $\text{fl}(\cdot) \to$ výpočet proveden v FPA

Danou operaci je možné provést s relativní přesností $\mathbf{u}$. Obecně tedy není možné očekávat, že algoritmus spočte řešení s lepší relativní přesností než je $\mathbf{u}$.

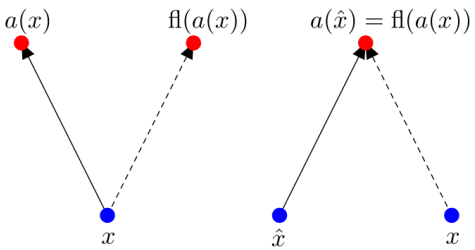
11.3 Zpětná stabilita algoritmu

- Algoritmus $a(x) \rightarrow$ výpočet $\rightarrow$ nepřesné výsledky.

$$\text{algoritmus:} \quad x \mapsto a(x)$$

  Označme $a(x)$ přesný a $\mathrm{fl}(a(x))$ spočtený výsledek.

- Pohled na chyby výpočtu



- **Přímá chyba** (forward error)

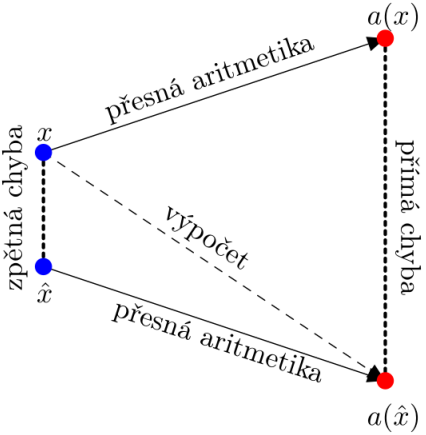
$$\|a(x) - \mathrm{fl}(a(x))\| \,.$$

  Efektivní odhad možný zřídka.

- **Zpětná chyba** (backward error)

$$\inf \{\|x - \hat{x}\| : \; a(\hat{x}) = \mathrm{fl}(a(x))\} \,.$$

  Hledáme modifikovaná vstupní data úlohy tak, aby spočtené řešení bylo přesným řešením té samé úlohy s modifikovanými vstupními daty. Chyby při výpočtu interpretujeme pomocí změn vstupních dat.



Označení $\mathcal{O}(\mathbf{u}) \to$ neurčité číslo,

$$|\mathcal{O}(\mathbf{u})| \leq K\mathbf{u},$$

kde K nezávisí na datech, může záviset na dimenzi úlohy (lineárně, kvadraticky, kubicky), viz skalární součin.

- Algoritmus $a(x)$ je **zpětně stabilní**, pokud $\forall x$ (přípustné) $\exists\, \hat{x}$:

$$\mathrm{fl}(a(x)) = a(\widehat{x}) \quad \text{a} \quad \frac{\|x - \widehat{x}\|}{\|x\|} = \mathcal{O}(\mathbf{u}).$$

- Algoritmus $a(x)$ je zpětně stabilní, jestliže se chyby výpočtu způsobené zaokrouhlováním promítnou do malých změn vstupních dat.

11.4 Výpočet skalárního součinu

Viz též [7, Lemma 3.1] a [7, s. 535].

Lemma 11.1. *Je-li* $|\delta_i| \leq \boldsymbol{u}$ *a* $\rho_i = \pm 1$ *pro* $i = 1, \ldots, n$, *a je-li* $n\boldsymbol{u} < 1$, *potom*

$$\prod_{i=1}^{n} (1 + \delta_i)^{\rho_i} = 1 + \theta_n,$$

kde

$$|\theta_n| \leq \gamma_n \equiv \frac{n\boldsymbol{u}}{1 - n\boldsymbol{u}}.$$

Poznámka 11.2. Je-li $n\boldsymbol{u} < 0.5$, pak je

$$\gamma_n \leq 2n\boldsymbol{u} = \mathcal{O}(\mathbf{u}).$$

Důkaz. Pro $\rho_n = 1$ dostáváme

$$\prod_{i=1}^{n} (1+\delta_i)^{\rho_i} = (1+\theta_{n-1})\,(1+\delta_n) = 1+\theta_n$$

a platí

$$\theta_n = \delta_n + (1+\delta_n)\theta_{n-1}.$$

Uvažujeme-li absolutní hodnotu θ_n, pak

$$\begin{aligned} |\theta_n| &\leq \boldsymbol{u} + (1+\boldsymbol{u})\frac{(n-1)\boldsymbol{u}}{1-(n-1)\boldsymbol{u}} \\ &= \frac{n\boldsymbol{u}}{1-(n-1)\boldsymbol{u}} \leq \gamma_n. \end{aligned}$$

Pro $\rho_n = -1$ analogicky (dobrovolné)

$$\prod_{i=1}^{n} (1+\delta_i)^{\rho_i} = (1+\theta_{n-1})\,(1+\delta_n)^{-1} = 1+\theta_n$$

a platí

$$\begin{aligned} |\theta_n| &= \left|\frac{\theta_{n-1}-\delta_n}{1+\delta_n}\right| \\ &\leq \frac{\frac{(n-1)u}{1-(n-1)u}+u}{1-u} \\ &= \frac{nu-(n-1)u^2}{1-nu+(n-1)u^2} \\ &\leq \frac{nu}{1-nu+2(n-1)u^2} \leq \gamma_n. \end{aligned}$$

□

Výpočet skalárního součinu

- Rozepišme algoritmus výpočtu detailně
  $s = 0$
  for $j = 1 : n$ **do**
  $s = s + x_j y_j$
  end
  a použijme model aritmetiky

$$\mathrm{fl}(x \circ y) = (x \circ y)(1+\delta), \qquad |\delta| \le \mathbf{u}.$$

- Je-li s_j mezivýsledek na konci j-tého cyklu, pak

$$\begin{aligned} s_j &= \mathrm{fl}(s_{j-1} + \mathrm{fl}(x_j y_j)), \quad s_0 = 0, \\ s_j &= [s_{j-1} + x_j y_j (1+\varepsilon_j)](1+\delta_j), \end{aligned}$$

  pro nějaké $|\varepsilon_j| \le \mathbf{u}$, $|\delta_j| \le \mathbf{u}$, $\delta_1 = 0$.

- Indukcí

$$\begin{aligned} s_n &= \sum_{j=1}^{n} x_j y_j \underbrace{(1+\varepsilon_j) \prod_{i=j}^{n} (1+\delta_i)}_{1+\mu_j} \\ &= \sum_{j=1}^{n} x_j y_j (1+\mu_j), \end{aligned}$$

  kde

$$|\mu_j| \le \gamma_{n-j+2} \le \gamma_n = \mathcal{O}(\mathbf{u}).$$

Zpětná stabilita

Celkem pro $n\boldsymbol{u} < 0.5$

$$s_n \quad = \quad \sum_{j=1}^{n} \underbrace{x_j}_{\widehat{x}_j} \underbrace{y_j(1+\mu_j)}_{\hat{y}_j} = \widehat{x}^T \widehat{y}$$

kde tj. s_n je přesným skalárním součinem perturbovaných vektorů, relativní chyby perturbovaných dat omezeny

$$\frac{\|x - \widehat{x}\|}{\|x\|} = 0 \quad \text{a} \quad \frac{\|y - \widehat{y}\|}{\|y\|} \; \leq \; \underbrace{2\,n\,\mathbf{u}}_{\mathcal{O}(\mathbf{u})}.$$

Výpočet skalárního součinu pomocí uvažovaného algoritmu je zpětně stabilní.

Poznámka o přímé chybě

- Dle předchozího platí

$$s_n \quad = \quad x^T y + \underbrace{\sum_{j=1}^{n} x_j y_j \mu_j}_{\hat{\delta}_n},$$

kde

$$|\hat{\delta}_n| \quad \leq \quad \underbrace{2\,n\,\mathbf{u}}_{\mathcal{O}(\mathbf{u})}\, |x|^T |y|\,.$$

Je-li

$$x^T y \ll |x|^T |y|,$$

relativní přesnost na úrovni $\mathcal{O}(\mathbf{u})$ není garantována.

- Pro výpočet $x^T x$ však platí

$$s_n = \|x\|^2 + \hat{\delta}_n, \quad |\hat{\delta}_n| \leq 2\,n\,\mathbf{u}\,\|x\|^2,$$

tj. výpočet $\|x\|$ je stabilní (forward stable), jelikož

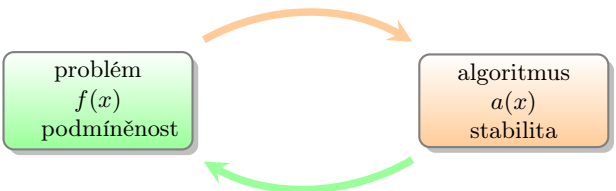
$$\frac{s_n - \|x\|^2}{\|x\|^2} = \mathcal{O}(\mathbf{u}).$$

11.5 Přesnost výpočtu

Algoritmus $a(x)$, (relativní) **přesnost výpočtu**

$$\frac{\|a(x) - \mathrm{fl}(a(x))\|}{\|a(x)\|}.$$

Lze o ní něco říci?



Tvrzení 11.3. Pro **zpětně stabilní** algoritmu $a(x)$ řešící problém $f(x)$ jež je „rozumně“ **spojitý** v x, platí:

$$\frac{\|a(x) - \mathrm{fl}(a(x))\|}{\|a(x)\|} = \mathcal{O}(\kappa_f(x)\mathbf{u}),$$

kde $\kappa_f(x)$ je podmíněnost problému f v x,

$$\kappa_f(x) \equiv \lim_{\delta \to 0} \sup_{\frac{\|\Delta x\|}{\|x\|} \leq \delta} \left(\frac{\|\Delta f(x)\|}{\|f(x)\|} \Big/ \frac{\|\Delta x\|}{\|x\|} \right).$$

Důkaz: Definujme

$$\kappa_{f,\delta}(x) \equiv \sup_{\frac{\|\Delta x\|}{\|x\|} \leq \delta} \left(\frac{\|\Delta f(x)\|}{\|f(x)\|} / \frac{\|\Delta x\|}{\|x\|} \right).$$

- „a řeší f“ znamená $f(x) = a(x)$ (matematicky)
- Je-li a zpětně stabilní, pak $\mathrm{fl}(a(x)) = a(\hat{x})$ pro

$$\frac{\|x - \hat{x}\|}{\|x\|} \leq K\mathbf{u}$$

  a platí

$$\frac{\|a(x) - \overbrace{\mathrm{fl}(a(x))}^{a(\hat{x})}\|}{\|a(x)\|} = \frac{\|f(x) - f(\hat{x})\|}{\|f(x)\|}.$$

- Zároveň

$$\frac{\frac{\|f(x)-f\hat{x})\|}{\|f(x)\|}}{\frac{\|x-\hat{x}\|}{\|x\|}} \leq \kappa_{f,K\mathbf{u}}(x).$$

- Celkem

$$\frac{\|a(x) - \mathrm{fl}(a(x))\|}{\|a(x)\|} \leq K\ \kappa_{f,K\mathbf{u}}(x)\,\mathbf{u}.$$

- Je-li problém „rozumně“ spojitý, lze očekávat

$$\kappa_{f,K\mathbf{u}}(x) \approx \kappa_f(x).$$

12

Analýza výpočtů

- přímé algoritmy (například Householder QR)
  - analýza zpětné chyby
- iterační (například sdružené gradienty)
  - jiné přístupy analýzy

12.1 Householderovy reflexe

- Refllexe a rotace → velmi užitečné, QR rozklad, QR algoritmus, transformace na požadovaný tvar.
- Detailní analýza v knize Wilkinsona [103, pp. 126-162] a Highama [7, 357-360].

Nulování složek $x \in \mathbb{R}^n$ pomocí Householderovy reflexe

$$H = I - 2qq^T.$$

Použijeme

$$q = \frac{x + \text{sign}(x_1)\|x\|e_1}{\|x + \text{sign}(x_1)\|x\|e_1\|},$$

potom

$$Hx = -\text{sign}(x_1)\|x\|e_1.$$

Pro zjednodušení uvažujme H ve tvaru

$$H = I - vv^T, \quad \|v\|_2 = \sqrt{2},$$

tj. $v = \sqrt{2}q$ v konstrukci výše.

Pro spočtené $\widehat{v}$ platí

$$\widehat{v} = v + \Delta v, \quad |\Delta v| \leq 5\,\gamma_n |v|, \tag{12.1}$$

viz [7, Lemma 19.1 a (19.5)].

Aplikace Householderových matic

Lemma 12.1 ([7, Lemma 19.3, s. 359]). *Uvažujme posloupnost transformací*

$$A_{j+1} = H_j A_j, \quad j = 1, 2, \ldots . k,$$

kde $A_1 = A \in \mathbb{R}^{n\times m}$ *a* $H_j = I - v_j v_j^T$ *je Householderova matice. Předpokládejme, že transformace jsou provedeny pomocí spočtených Householderových vektorů* $\widehat{v}_j \approx v_j$ *splňujících* (12.1).

Pak spočtená matice $\widehat{A}_{k+1}$ *splňuje*

$$\widehat{A}_{k+1} = Q^T(A + \Delta A), \qquad Q \equiv H_k H_{k-1} \ldots H_1$$

a

$$\frac{\|\Delta A\|_F}{\|A\|_F} \leq ck\gamma_n.$$

Householder QR

Věta 12.2 ([7, Theorem 19.4 a (19.3)]). *Nechť $\widehat{Q} \in \mathbb{R}^{n\times n}$ a $\widehat{R} \in \mathbb{R}^{n\times m}$ jsou spočtené faktory QR rozkladu matice $A \in \mathbb{R}^{n\times m}$, $n \geq m$, počítaného pomocí Householder QR.*

Potom existuje ortonormální $Q \in \mathbb{R}^{n\times n}$ taková, že

$$A + \Delta A = Q\widehat{R},$$

kde

$$\|\Delta A\|_F \leq c\gamma_{nm}\|A\|_F \quad a \quad \|\widehat{Q} - Q\|_F \leq c\sqrt{m}\gamma_{nm},$$

kde $c > 0$ je malá konstanta nezávislá na n a m.

Celkem tedy platí, že

$$A + \Delta A = \widehat{Q}\widehat{R},$$

kde $\widehat{Q}$ je téměř ortonormální a

$$\frac{\|\Delta A\|_F}{\|A\|_F} = \mathcal{O}(\boldsymbol{u})\,.$$

12.2 Iterační metody

Základní operace

Pomocí standardního modelu [4]:

$$\begin{aligned} \|\alpha\, v - \mathrm{fl}\,(\alpha\, v)\| &\leq \boldsymbol{u}\ \|\alpha\, v\|\,, \\ \|v + w - \mathrm{fl}\,(v + w)\| &\leq \boldsymbol{u}\,(\|v\| + \|w\|), \\ |v^T w - \mathrm{fl}\left(v^T w\right)| &\leq \boldsymbol{u}\, n\,(1 + O(\boldsymbol{u}))\ \|v\|\,\|w\|\,, \\ \|Av - \mathrm{fl}\,(Av)\| &\leq \boldsymbol{u}\, m\, n^{1/2}\ \|A\|\,\|v\|\,, \end{aligned}$$

m je maximální počet nenulových prvků na řádku.

Lokální chyby

- chyby v jednotlivých iteracích
- co se s nimi děje během iterací?
  - akumulace
  - amplifikace

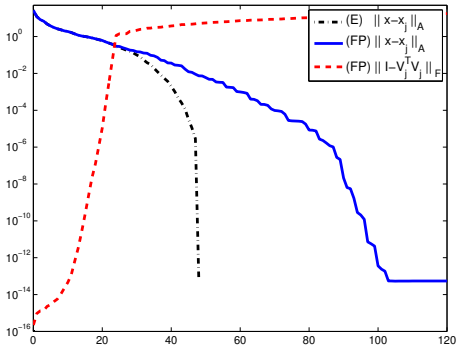
Globální pohled

- CG a Lanczos → ztráta ortogonality
- Chování velmi odlišné od přesného algoritmu!
- Aproximují co mají?

12.3 Finite precision CG

Ztráta ortogonality mezi $r_0, \ldots, r_{k-1}$, měřena pomocí

$$\|I - V_k^T V_k\|_F\,, \qquad v_j = \frac{r_j}{\|r_j\|}\,.$$



Od určité iterace se již nepřibližujeme k řešení → **limitní přesnost**.

Aproximuje CG to, co má, tj. řešení $Ax = b$?

Ano, ale … vlivem FP dochází ke **zpoždění** konvergence a **stagnaci** normy chyby na hladině limitní přesnosti, netriviální analýza [Paige, Greenbaum, Strakoš].

Příklad akumulace chyb

Nechť x_k a r_k jsou počítány pomocí FP CG

$$\begin{aligned} x_k &= x_{k-1} + \alpha_{k-1} p_{k-1} + \delta x_k, \\ r_k &= r_{k-1} - \alpha_{k-1} A p_{k-1} + \delta r_k, \end{aligned}$$

$$\begin{aligned} \|\delta x_k\| &\leq \mathcal{O}(\boldsymbol{u}) \max\{\|x_{k-1}\|, \|x_k\|\}, \\ \|\delta r_k\| &\leq \mathcal{O}(\boldsymbol{u}) \max\{\|r_{k-1}\|, \|\alpha_{k-1} A p_{k-1}\|\}. \end{aligned}$$

Analýza rozdílu r_k a $b - Ax_k$

$$f_k \equiv r_k - (b - Ax_k)\,.$$

Dosazením za r_k a x_k dostaneme

$$f_k = f_{k-1} + \delta r_k + A\delta x_k = f_0 + \sum_{i=1}^{n} \delta r_i + A \sum_{i=1}^{n} \delta x_i.$$

Lze dokázat, viz [5],

$$\|f_k\| \leq \mathcal{O}(\boldsymbol{u}) \|A\| \left(\|x\| + \max_{i=0,\ldots,k} \|x_i\| \right).$$

Důsledek: hladina limitní přesnosti

Jelikož $\|r_k\| \to 0$, lze očekávat stagnaci na hladině

$$\|b - Ax_k\| \approx \boldsymbol{u} \|A\| \|x\|.$$

12.4 Lanczosova metoda

- Lanczosův algoritmus:

$$AV_k = V_k \underbrace{T_k}_{S_k \Theta_k S_k^T} + \beta_{k+1} v_{k+1} e_k^T.$$

Použij spektrální rozklad T_k,

$$A \underbrace{V_k S_k}_{Z_k} = \underbrace{V_k S_k}_{Z_k} \Theta_k + \beta_{k+1}\, v_{k+1} e_k^T S_k,$$

Z_k je matice **Ritzových** vektorů.

- Kvalitu $\theta_i^{(k)}$ a $z_i^{(k)}$ měříme pomocí

$$\delta_{k,i} \equiv \|Az_i^{(k)} - \theta_i^{(k)} z_i^{(k)}\|.$$

Nerovnost

$$\min_{j=1,\ldots,n} |\theta_i^{(k)} - \lambda_j| \leq \delta_{k,i}$$

platí **díky ortogonalitě** V_k.

- V FP je však ortogonalita **ztracena**!

Finite precision Lanczos

- ztráta ortogonality $I - V_k^T V_k$ ... chaos
- prvky $I - Z_k^T Z_k$ ... struktura
- obtížná analýza → Paige
- při výpočtech v FP platí

$$\min_{j=1,\ldots,n} |\theta_i^{(k)} - \lambda_j| \leq 3\,\delta_{k,i} + \mathcal{O}(\boldsymbol{u})\|A\|\,.$$

- příklad výsledku o ztrátě ortogonality

$$\left| v_{k+1}^T z_i^{(k)} \right| \leq \frac{\mathcal{O}(\boldsymbol{u})\|A\|}{\delta_{k,i}}.$$

Aproximuje Lanczos to, co má, tj. vlastní páry?

Ano, ale ... vlivem FP může být jeden vlastní pár aproximován několika Ritzovými páry, dochází k vytváření tzv. **kopií** aproximujících původní vlastní pár, netriviální analýza [Paige, Greenbaum, Strakoš, Wuelling].

Literatura

[1] M. Crouzeix and C. Palencia, *The numerical range is a* $(1+\sqrt{2})$*-spectral set*, SIAM J. Matrix Anal. Appl., 38 (2017), pp. 649–655.

[2] V. Faber, J. Liesen, and P. Tichý, *The Faber-Manteuffel theorem for linear operators*, SIAM J. Numer. Anal., 46 (2008), pp. 1323–1337.

[3] V. Faber and T. Manteuffel, *Necessary and sufficient conditions for the existence of a conjugate gradient method*, SIAM J. Numer. Anal., 21 (1984), pp. 352–362.

[4] G. H. Golub and C. F. Van Loan, *Matrix computations*, Johns Hopkins Studies in the Mathematical Sciences, Johns Hopkins University Press, Baltimore, MD, fourth ed., 2013.

[5] A. Greenbaum, *Estimating the attainable accuracy of recursively computed residual methods*, SIAM J. Matrix Anal. Appl., 18 (1997), pp. 535–551.

[6] A. Greenbaum, V. Pták, and Z. Strakoš, *Any nonincreasing convergence curve is possible for GMRES*, SIAM J. Matrix Anal. Appl., 17 (1996), pp. 465–469.

[7] N. J. Higham, *Accuracy and stability of numerical algorithms*, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, second ed., 2002.

[8] ——, *Functions of matrices*, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2008. Theory and computation.

[9] K. Hoffman and R. Kunze, *Linear algebra*, Second edition, Prentice-Hall Inc., Englewood Cliffs, N.J., 1971.

[10] R. A. Horn and C. R. Johnson, *Topics in matrix analysis*, Cambridge University Press, Cambridge, 1994. Corrected reprint of the 1991 original.

[11] J. Liesen, *When is the adjoint of a matrix a low degree rational function in, the matrix?*, SIAM J. Matrix Anal. Appl., 29 (2007), pp. 1171–1180.

[12] J. Liesen and Z. Strakoš, *On optimal short recurrences for generating orthogonal Krylov subspace bases*, SIAM Rev., 50 (2008), pp. 485–503.

[13] J. Liesen and Z. Strakoš, *Krylov subspace methods*, Numerical Mathematics and Scientific Computation, Oxford University Press, Oxford, 2013. Principles and analysis.

[14] S. Van Huffel and J. Vandewalle, *The total least squares problem*, vol. 9 of Frontiers in Applied Mathematics, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1991. Computational aspects and analysis, With a foreword by Gene H. Golub.