Numerická matematika 1

(poznámky z přednášek doc. Tomáše Oberhubera 2024/2025)

Daniel Janda, FJFI ČVUT

verze 27. června 2025

Obsah

Úvod **7**

1 Lineární algebra pro numeriku **11**
1.1 Základní pojmy . . . 11
1.1.1 Vektory . . . 11
1.1.2 Matice . . . 11
1.1.3 Skalární součin . . . 13
1.1.4 Normální matice . . . 14
1.1.5 Trojúhelníkové matice . . . 15
1.1.6 Vlastní čísla a podobnost . . . 16
1.2 Rozklady matic . . . 18
1.3 Posloupnosti vektorů a matic a normy . . . 24
1.3.1 Posloupnosti . . . 24
1.3.2 Normy . . . 25
1.3.3 Geometrické posloupnosti . . . 29
1.4 Otázky . . . 32

2 Úvod do numerických výpočtů **33**
2.1 Reprezentace čísel v počítači . . . 33
2.1.1 Čísla s pohyblivou desetinnou čárkou . . . 33
2.1.2 Standard IEEE 754 . . . 34
2.1.3 Zaokrouhlovací chyby . . . 35
2.2 Stabilita úlohy a podmíněnost matic . . . 36
2.3 Klasifikace numerických metod . . . 38
2.3.1 Přímé metody . . . 38
2.3.2 Iterační metody . . . 38
2.4 Základní pojmy analýzy iteračních metod . . . 38
2.4.1 Apriorní a aposteriorní odhady . . . 39
2.5 Zdroje chyb v numerických simulacích . . . 39
2.6 Otázky . . . 39

3 Přímé metody řešení soustav lineárních rovnic **41**
3.1 Gaussova eliminační metoda . . . 41
3.1.1 Implementace . . . 42
3.1.2 Analýza složitosti . . . 42
3.1.3 Numerická analýza . . . 42
3.1.4 Modifikovaná Gaussova eliminační metoda . . . 47
3.1.5 Otázky . . . 48
3.2 LU rozklad . . . 48
3.2.1 LU rozklad pomocí Gaussovy eliminační metody . . . 49

3.2.2 Kompaktní schéma pro LU faktorizaci . . . 51
3.2.2.1 Implementace . . . 52
3.2.2.2 Analýza složitosti . . . 52
3.2.3 Choleského rozklad . . . 53
3.2.4 Otázky . . . 53
3.3 Modifikace Gaussovy eliminační metody . . . 53
3.3.1 Thomasův algoritmus . . . 53
3.3.1.1 Význam . . . 55
3.3.2 Schurův doplněk . . . 55
3.3.2.1 Význam . . . 56
3.3.2.2 Speciální případ pro $r = 2$ . . . 56
3.3.2.3 GEM a blokový tvar . . . 56
3.3.3 Otázky . . . 57

4 Iterativní metody řešení soustav lineárních rovnic 59
4.1 Základní pojmy . . . 59
4.1.1 Složitost iteračních metod . . . 60
4.1.2 Odhady chyb a ukončovací kritéria . . . 61
4.1.3 Metoda postupných aproximací . . . 63
4.1.4 Předpodmínění . . . 64
4.1.5 Otázky . . . 66
4.2 Richardsonovy iterace . . . 66
4.2.1 Implementace . . . 67
4.3 Jacobiho metoda . . . 67
4.3.1 Implementace . . . 69
4.4 Gaussova-Seidelova metoda . . . 69
4.5 Superrelaxační metoda (SOR) . . . 71
4.5.1 Optimální volba relaxačního parametru . . . 74
4.6 Korektnost a porovnání konvergence probraných iterativních metod . . . 74
4.7 Porovnání přímých a iteračních metod . . . 74
4.7.1 Výhody iteračních metod . . . 74
4.7.2 Nevýhody iteračních metod . . . 75
4.8 Otázky . . . 75

5 Metody výpočtu vlastních čísel matic 77
5.1 Aplikace a motivace . . . 77
5.2 Základní pojmy . . . 77
5.2.1 Lokalizace a odhady chyb vlastních čísel . . . 78
5.2.2 Otázky . . . 79
5.3 Částečný problém vlastních čísel . . . 79
5.3.1 Mocninná metoda . . . 79
5.3.1.1 Odvození mocninné metody . . . 79
5.3.1.2 Ukázkové příklady průběhu mocninné metody . . . 80
5.3.1.3 Konvergence mocninné metody pro hermitovské matice . . . 81
5.3.1.4 Obecné konvergenční věty . . . 82
5.3.1.5 Praktické aspekty a varianty . . . 85
5.3.2 Redukční metoda (Deflace) . . . 86
5.3.3 Otázky . . . 87
5.4 Analýza kompletního spektra matice . . . 87
5.4.1 Trojúhelníková metoda . . . 87
5.4.1.1 Konvergence a existence rozkladu . . . 88

5.4.2 LR algoritmus . . . 90
5.4.2.1 Konvergence LR algoritmu . . . 91
5.5 QR algoritmus . . . 92
5.5.1 QR rozklad . . . 92
5.5.1.1 Gramův-Schmidtův ortonormalizační proces . . . 93
5.5.1.2 Householderovy transformace . . . 94
5.5.1.3 Givensovy rotace . . . 96
5.5.1.4 Shrnutí a srovnání metod QR rozkladu . . . 97
5.5.2 QR algoritmus . . . 98
5.5.2.1 Konvergence QR algoritmu . . . 98
5.5.3 QR algoritmus s Hessenbergovými maticemi . . . 99
5.5.3.1 Převod na Hessenbergův tvar . . . 100
5.5.3.2 Vlastní QR iterace s Hessenbergovou maticí . . . 100
5.6 Otázky . . . 101

6 Metody řešení nelineárních rovnic 103
6.1 Separace kořenů . . . 103
6.1.1 Metoda půlení intervalu (bisekce) . . . 104
6.1.2 Obecná iterační metoda a podmínky konvergence . . . 105
6.1.2.1 Podmínky konvergence . . . 105
6.1.3 Metoda regula falsi (metoda sečen) . . . 106
6.1.4 Newtonova metoda (metoda tečen) . . . 107
6.1.5 Globálně konvergující metody . . . 108
6.2 Řešení soustav nelineárních rovnic . . . 109
6.2.1 Newtonova metoda pro soustavy . . . 109
6.3 Otázky . . . 110

7 Numerická interpolace funkcí 111
7.1 Interpolační polynom . . . 111
7.1.1 Obecná konstrukce a existence . . . 111
7.1.2 Lagrangeův tvar interpolačního polynomu . . . 112
7.1.3 Newtonova formule . . . 112
7.1.3.1 Poměrné diference . . . 113
7.2 Analýza interpolačního polynomu . . . 116
7.2.1 Chyba aproximace . . . 116
7.2.2 Řád aproximace . . . 117
7.2.3 Rungův jev . . . 117
7.3 Interpolace po částech . . . 118
7.3.1 Hermitova-Birkhoffova interpolace* . . . 118
7.4 Interpolace ve vyšších dimenzích . . . 119
7.4.1 Otázky . . . 119

8 Numerický výpočet derivace 121
8.1 Výpočet derivace pomocí interpolačního polynomu . . . 121
8.1.1 Derivace interpolačního polynomu . . . 121
8.1.2 Chyba numerické derivace . . . 121
8.1.3 Konečné diference . . . 123
8.1.3.1 Dopředná diference 1. řádu přesnosti . . . 124
8.1.3.2 Zpětná diference 1. řádu přesnosti . . . 124
8.1.3.3 Centrální diference 2. řádu přesnosti . . . 124
8.2 Výpočet derivace pomocí Taylorova polynomu . . . 125
8.2.1 Konečné diference . . . 125

8.2.1.1 Dopředná diference . . . 125
8.2.1.2 Zpětná diference . . . 125
8.2.1.3 Centrální diference pro první derivaci . . . 126
8.2.1.4 Aproximace druhé derivace . . . 126
8.2.1.5 Obecná konstrukce konečných diferencí . . . 127
8.3 Shrnutí . . . 128
8.4 Otázky . . . 128

9 Numerická integrace **129**

Upozornění

Tento text vznikl jako pomocný studijní materiál na základě zápisků z přednášek. Nejedná se o oficiální učebnici ani garantovaný zdroj pravdivých informací. Přestože se autor snažili o co největší přesnost, dokument může obsahovat chyby věcné, jazykové i formální. Doporučujeme konzultovat též doporučenou literaturu a originální studijní materiály.

V tuto chvíli se také jedná o pracovní verzi textu. Obsahuje poznámky k plánovaným úpravám, nebyla provedena jazyková korektura a rozložení obsahu na stránkách zatím není optimalizováno. Dokument se bude v čase dále vyvíjet a upravovat.

Máte-li podněty, zpětnou vazbu nebo jste objevili chybu, napište mi prosím na: jandada4@fjfi.cvut.cz.

Historický kontext a úvod

Numerická matematika je samostatnou disciplínou aplikované matematiky, která se zabývá návrhem, analýzou a praktickou realizací algoritmů pro řešení matematických úloh pomocí počítače. Typickým rysem numerických metod je, že poskytují pouze *přibližná řešení*, která se snaží co nejlépe přiblížit teoretickému (často neznámému) výsledku s garantovanou nebo odhadnutelnou chybou.

Historie numeriky

Počátky numerické matematiky sahají hluboko do historie – již staří Babylóňané a Egypťané používali praktické algoritmy k výpočtům odmocnin nebo řešení rovnic. Za první „numerické metody“ lze považovat například Eukleidův algoritmus pro výpočet největšího společného dělitele nebo metody používané Číňany ve 3. století pro řešení soustav lineárních rovnic.

Skutečný rozvoj numerické matematiky však nastal až s příchodem elektronických počítačů ve 20. století. Potřeba řešit složité inženýrské, fyzikální či ekonomické problémy vedla ke vzniku nových algoritmů, teorií stability a aproximace a celé řady specializovaných metod (např. metody konečných prvků, rychlé transformace, adaptivní integrace apod.). Numerika se tak stala nepostradatelným nástrojem moderní vědy a techniky.

Předmět a cíl kurzu

V tomto kurzu se zaměříme na klíčové numerické metody, které se používají při řešení běžných matematických problémů vznikajících v aplikacích. Konkrétně se budeme věnovat:

- řešení soustav lineárních rovnic,
- výpočtu (části) spektra matic,
- řešení nelineárních rovnic a soustav,
- interpolaci funkcí a konstrukci aproximačních metod,
- numerickému výpočtu derivací a integrálů.

Ve všech těchto oblastech nás bude zajímat:

- jak metoda funguje (algoritmický princip),
- jak je přesná (odhad chyby),
- jak je stabilní (citlivost na chyby v datech nebo aritmetice),
- jak je efektivní (počet operací, paměťová náročnost),
- a zda je její použití v dané situaci vhodné.

Numerické metody jsou navrženy tak, aby byly co nejodolnější vůči chybám způsobeným omezenou přesností počítačové aritmetiky, a zároveň aby bylo možné výsledek spočítat v rozumném čase. Zásadním aspektem je proto i pochopení základních principů *numerické stability* a *podmíněnosti úloh*.

Organizační poznámky

Tento studijní materiál vznikl na základě poznámek docenta Tomáše Oberhubera na FJFI ČVUT v roce 2024/2025. Doplněny byly o poznatky z doprovodných videí k přednášce dostupných na YouTube, žádné další zdroje nebyly využity.

Zdrojové kódy k vybraným metodám poskytuje docent Oberhuber na adrese:

`https://gitlab.com/oberhuber.tomas/fjfi-num-src`

Kód lze stáhnout pomocí příkazu: `git clone git@gitlab.com:oberhuber.tomas/fjfi-num-src.git` nebo prostřednictvím tlačítka *Download* na dané stránce.

Kapitola 1

Lineární algebra pro numeriku

1.1 Základní pojmy

1.1.1 Vektory

Definice 1.1. Uspořádanou n-tici čísel $x_i \in \mathbb{C}\ \forall i \in \hat{n}$ nazeme **vektorem** a píšeme

$$\vec{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \in \mathbb{C}^n.$$

Definice 1.2. Sčítání vektorů a násobení vektoru číslem definujeme po složkách, tj. $\forall\, \vec{x}, \vec{y} \in \mathbb{C}^n$ a $\forall\, \alpha \in \mathbb{C}$ platí

$$\vec{x} + \vec{y} = \begin{pmatrix} x_1 + y_1 \\ \vdots \\ x_n + y_n \end{pmatrix}, \qquad \alpha \cdot \vec{x} = \begin{pmatrix} \alpha x_1 \\ \vdots \\ \alpha x_n \end{pmatrix}.$$

1.1.2 Matice

Definice 1.3. Maticí s rozměry $m \times n$ rozumíme tabulku s m řádky a n sloupci hodnot $a_{ij} \in \mathbb{C}$ kde $i \in \hat{m}, j \in \hat{n}$. Píšeme

$$\mathbb{A} = \begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \dots & a_{mn} \end{pmatrix} \in \mathbb{C}^{m,n}.$$

Definice 1.4. Sčítání matic a násobení matic číslem definujeme po složkách, tj. $\forall\, \mathbb{A}, \mathbb{B} \in \mathbb{C}^{m,n}$ a $\forall\, \alpha \in \mathbb{C}$ platí

$$\mathbb{A} + \mathbb{B} = \begin{pmatrix} a_{11} + b_{11} & \dots & a_{1n} + b_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} + b_{m1} & \dots & a_{mn} + b_{mn} \end{pmatrix}, \qquad \alpha \cdot \mathbb{A} = \begin{pmatrix} \alpha a_{11} & \dots & \alpha a_{1n} \\ \vdots & \ddots & \vdots \\ \alpha a_{m1} & \dots & \alpha a_{mn} \end{pmatrix}.$$

Definice 1.5. Nechť $\mathbb{A} \in \mathbb{C}^{m,n}, \mathbb{B} \in \mathbb{C}^{n,p}$. **Součinem** $\mathbb{A}$ a $\mathbb{B}$ nazveme matici typu $m \times p$, značíme ji $\mathbb{AB}$ a definujeme

$$[\mathbb{AB}]_{ij} = \sum_{k=1}^{n} \mathbb{A}_{ik}\mathbb{B}_{kj} \quad \forall\, i \in \hat{m}, j \in \hat{n}.$$

Věta 1.6 (vlastnosti součinu matic)**.** Nechť $\mathbb{A} \in \mathbb{C}^{m,n}, \mathbb{B} \in \mathbb{C}^{n,p}, \mathbb{M} \in \mathbb{C}^{p,s}$. Násobení matic má následující vlastnosti:

1. Je asociativní, tj. platí, že $(\mathbb{AB})\mathbb{M} = \mathbb{A}(\mathbb{BM})$,
2. Je distributivní vůči sčítání matic, tj. platí, že $\mathbb{A}(\mathbb{B} + \mathbb{M}) = \mathbb{AB} + \mathbb{AM}$,
3. Není obecně komunitativní, tj. platí, že $\mathbb{AB} \neq \mathbb{BA}$.

Důkaz. Je třeba zkontrolovat rozměry, následně plyne triviálně z definice, viz str. 81 [1]. Nezkouší se. □

Definice 1.7. Nechť $\mathbb{A} \in \mathbb{C}^{n,n}$. Pak **determinantem** matice A nazveme číslo

$$\det \mathbb{A} = \sum_{\pi \in S_n} \operatorname{sgn} \pi \cdot \mathbb{A}_{1\pi(1)} \cdot \mathbb{A}_{2\pi(2)} \cdot \ldots \cdot \mathbb{A}_{n\pi(n)}$$

kde S_n je množina všech permutací na $\hat{n}$ a $\operatorname{sgn} \pi = (-1)^{I_\pi}$, kde I_π je počet inverzí v π.

Věta 1.8 (determinant trojíhelníkové matice)**.** Nechť $\mathbb{A} \in \mathbb{C}^{n,n}$ je dolní, resp. horní trojúhelníková matice. Pak $\det \mathbb{A} = \mathbb{A}_{11} \cdot \mathbb{A}_{22} \cdot \ldots \cdot \mathbb{A}_{nn}$.

Důkaz. Zřejmě lze vybírat jen diagonální prvky, jinak bude člen sumy v definici determinantu nulový. Podrobněji viz strana 27 v [1]. Nezkouší se. □

Definice 1.9. Nechť $\mathbb{A} \in \mathbb{C}^{m,n}$. Pak

- matice **transponovaná** k matici $\mathbb{A}$ je typu $n \times m$, značí se $\mathbb{A}^T$ a $\forall\, i \in \hat{n}, j \in \hat{m}$ splňuje

$$\left[\mathbb{A}^T\right]_{ij} = A_{ji}$$

- matice **komplexně sdružená** k matici $\mathbb{A}$ je typu $m \times n$, značí se $\overline{\mathbb{A}}$ a $\forall\, i \in \hat{m}, j \in \hat{n}$ splňuje

$$\left[\overline{\mathbb{A}}\right]_{ij} = \overline{\mathbb{A}_{ij}}$$

- matice **hermitovsky sdružená** k matici $\mathbb{A}$ je typu $n \times m$, značí se $\mathbb{A}^H$ nebo $\mathbb{A}^*$ a $\forall\, i \in \hat{n}, j \in \hat{m}$ splňuje

$$\left[\mathbb{A}^*\right]_{ij} = \overline{A_{ji}}$$

Věta 1.10. Nechť $\mathbb{A} \in \mathbb{C}^{m,n}, \mathbb{B} \in \mathbb{C}^{n,p}$. Potom platí

1. $(\mathbb{AB})^T = \mathbb{B}^T\mathbb{A}^T$
2. $\overline{\mathbb{AB}} = \overline{\mathbb{A}}\,\overline{\mathbb{B}}$
3. $(\mathbb{AB})^* = \mathbb{B}^*\mathbb{A}^*$

Důkaz. Lze triviálně ověřit z definice maticového násobení pro ij-tý prvek. Nezkouší se. □

Definice 1.11. Nechť $\mathbb{A} \in \mathbb{C}^{m,n}$. **Hodnost** *(rank)* matice $\mathbb{A}$ značíme $h(\mathbb{A})$ nebo $\operatorname{rank}(\mathbb{A})$ a definujeme jako maximální počet nenulových subdeterminantnů různého řádu vybraných z matice $\mathbb{A}$.

Poznámka 1.12. Tato definice se liší od definice z Lineární algebry 1, viz [1].

Definice 1.13. Nechť $\mathbb{A} \in \mathbb{C}^{m,n}$. **Obraz** *(range)* matice $\mathbb{A}$ je vektorový prostor, definovaný jako

$$\operatorname{range}(\mathbb{A}) = \{\mathbb{A}\vec{x} \in \mathbb{C}^m \mid \vec{x} \in \mathbb{C}^m\}$$

Definice 1.14. Nechť $\mathbb{A} \in \mathbb{C}^{m,n}$. **Jádro** *(kernel)* matice $\mathbb{A}$ je vektorový prostor, definovaný jako

$$\ker(\mathbb{A}) = \{\vec{x} \in \mathbb{C}^m \mid \mathbb{A}\vec{x} = \mathbb{O}\}$$

Věta 1.15. Nechť $\mathbb{A} \in \mathbb{C}^{m,n}$. Potom platí

1. $\operatorname{rank}(\mathbb{A}) = \operatorname{rank}\left(\mathbb{A}^T\right)$
2. $\operatorname{rank}(\mathbb{A}) = \operatorname{rank}\left(\mathbb{A}^*\right)$
3. $\operatorname{rank}(\mathbb{A}) + \dim(\ker(\mathbb{A})) = n$

Důkaz. Důkazy těchto tvrzení jsou poměrně složité, viz strana 74 a 93-95 [1]. Nezkouší se. □

Definice 1.16. Čtvercovou matici $\mathbb{A} \in \mathbb{C}^{n,n}$ nazveme **regulární** právě tehdy když $h(\mathbb{A}) = n$. V opačném případě se $\mathbb{A}$ nazývá **singulární**.

Definice 1.17. Čtvercovou matici $\mathbb{A} \in \mathbb{C}^{n,n}$ nazveme **silně regulární** právě tehdy když platí

$$\det\begin{pmatrix} a_{11} \end{pmatrix} \neq 0,$$

$$\det\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \neq 0,$$

$$\det\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \neq 0,$$

$$\vdots$$

$$\det \mathbb{A} \neq 0.$$

Definice 1.18. Čtvercovou matici $\mathbb{A} \in \mathbb{C}^{n,n}$ nazveme **diagonální**, právě tehdy když $\forall i, j \in \hat{n}$, $i \neq j$ platí $a_{ij} = 0$.

Definice 1.19. Jednotkovou maticí myslíme diagonální matici takovou, že $\forall i \in \hat{n}$ platí $a_{ii} = 1$. Značíme $\mathbb{I}$.

Věta 1.20. Nechť $\mathbb{A} \in \mathbb{C}^{n,n}$ je regulární matice. Potom k ní existuje právě jedna taková matice $B \in \mathbb{C}^{n,n}$, že platí $\mathbb{A}\mathbb{B} = \mathbb{B}\mathbb{A} = \mathbb{I}$. Matici B nazveme inverzní k $\mathbb{A}$ a značíme $\mathbb{A}^{-1}$.

Důkaz. Důkaz viz strana 14 [2]. Nezkouší se. □

Definice 1.21. Bloková matice je taková matice, jejíž prvky jsou matice. Přitom prvky blokové matice ve stejném sloupci musí mít stejný počet sloupců a ve stejných řádcích musí mít stejný počet řádku. Sčítání matic, násobení matice číslem i součin matic je definováno standardně.

Definice 1.22. Řídká matice je taková matice, jejíž většina prvků je nulová.

To umožňuje šetřit paměť, neb můžeme ukládat jen nenulové hodnoty.

1.1.3 Skalární součin

Definice 1.23. V numerické matematice budeme uvažovat **standardní skalární součin** definovaný $\forall \vec{x}, \vec{y} \in \mathbb{C}^n$

$$(\vec{x}, \vec{y}) = \vec{x}^T \overline{\vec{y}} = \sum_{i=1}^{n} x_i \overline{y_i}.$$

Věta 1.24 (vlastnosti skalárního součinu)**.** Pro takto definovaný skalární součin a $\forall\, \vec{x}, \vec{y}, \vec{z} \in \mathbb{C}^n$, $\forall\, \alpha \in \mathbb{C}$ zřejmě platí následující vlastnosti:

1. $(\vec{x}, \vec{x}) \geq 0$ a $(\vec{x}, \vec{x}) = 0 \iff \vec{x} = \vec{0}$ *(pozitivní definitnost)*
2. $(\vec{x}, \vec{y}) = \overline{(\vec{y}, \vec{x})}$ *(hermitovskost)*
3. $(\alpha\vec{x} + \vec{y}, \vec{z}) = \alpha(\vec{x}, \vec{z}) + (\vec{y}, \vec{z})$ *(linearita v prvním argumentu)*

Důkaz. Plyne přímo z definice. Nezkouší se. □

Věta 1.25 (Cauchyho-Schwartzova nerovnost)**.** Nechť $\vec{x}, \vec{y} \in \mathbb{C}^n$. Potom platí

$$|(\vec{x}, \vec{y})|^2 \leq (\vec{x}, \vec{x}) \cdot (\vec{y}, \vec{y})$$

Důkaz. Viz str. 79 [2]. Nezkouší se. □

Věta 1.26. Nechť $\vec{x}, \vec{y} \in \mathbb{C}^n$. Potom platí

$$(\mathbb{A}\vec{x}, \vec{y}) = (\vec{x}, \mathbb{A}^*\vec{y}).$$

Důkaz. $(\mathbb{A}\vec{x}, \vec{y}) = (\mathbb{A}\vec{x})^T\overline{\vec{y}} = \vec{x}^T\mathbb{A}^T\overline{\vec{y}} = \vec{x}^T\overline{\overline{\mathbb{A}^T}\vec{y}} = \vec{x}^T\overline{\mathbb{A}^*\vec{y}} = (\vec{x}, \mathbb{A}^*\vec{y})$ □

1.1.4 Normální matice

Definice 1.27. Nechť $\mathbb{A} \in \mathbb{C}^{n,n}$. Potom $\mathbb{A}$ nazvu

- **normální** $\iff \mathbb{A}\mathbb{A}^* = \mathbb{A}^*\mathbb{A}$,
- **samosdruženou**/hermitovskou $\iff \mathbb{A}^* = \mathbb{A}$,
- **izometrickou**/unitární $\iff \mathbb{A}^* = \mathbb{A}^{-1}$.

Poznámka 1.28. Je-li matice $\mathbb{A} \in \mathbb{C}^{n,n}$ samosdružená/hermitovská, je také normální.

Důkaz. $\mathbb{A}\mathbb{A}^* = \mathbb{A}\mathbb{A} = \mathbb{A}^*\mathbb{A}$ □

Poznámka 1.29. Je-li matice $\mathbb{A} \in \mathbb{C}^{n,n}$ izometrická/unitární, je také normální.

Důkaz. $\mathbb{A}\mathbb{A}^* = \mathbb{A}\mathbb{A}^{-1} = \mathbb{I} = \mathbb{A}^{-1}\mathbb{A} = \mathbb{A}^*\mathbb{A}$ □

Poznámka 1.30. Jednotková matice $\mathbb{I}$ je hermitovská i unitární.

Věta 1.31. Nechť $\mathbb{A}, \mathbb{B} \in \mathbb{C}^{n,n}$ jsou unitární. Potom platí

1. $\mathbb{A}\mathbb{B}$ je unitární,
2. $|\det \mathbb{A}| = 1$.

Důkaz 1. $(\mathbb{A}\mathbb{B})(\mathbb{A}\mathbb{B})^* = \mathbb{A}\mathbb{B}\mathbb{B}^*\mathbb{A}^* = \mathbb{A}\mathbb{B}\mathbb{B}^{-1}\mathbb{A}^{-1} = \mathbb{A}\mathbb{I}\mathbb{A}^{-1} = \mathbb{A}\mathbb{A}^{-1} = \mathbb{I} \implies (\mathbb{A}\mathbb{B})^* = (\mathbb{A}\mathbb{B})^{-1}$ □

Důkaz 2. Můžeme postupně psát

$$\det \mathbb{A} \overset{1}{=} \det \mathbb{A}^* \overset{2}{=} \det \mathbb{A}^{-1} = \frac{1}{\det \mathbb{A}},$$

kde první rovnost plyne z faktu, že transpozice ani komplexní sdružení nemají na hodnotu determinantu vliv, a druhá rovnost plyne z unitárnosti matice $\mathbb{A}$. Tato rovnost je v $\mathbb{R}$ naplněna pro $\det \mathbb{A} = \pm 1$ a v $\mathbb{C}$ pro $|\det \mathbb{A}| = 1$. □

Věta 1.32. Nechť $\mathbb{U} \in \mathbb{C}^{n,n}$ je unitární matice. Potom jsou její sloupce ortonormální.

Důkaz. $\delta_{ij} = \mathbb{I}_{ij} = [\mathbb{U}^*\mathbb{U}]_{ij} = \sum_{k=1}^{n} \mathbb{U}^*_{ik}\mathbb{U}_{kj} = \sum_{k=1}^{n} \overline{\mathbb{U}_{ki}}\mathbb{U}_{kj} = (\mathbb{U}_{\cdot i}, \mathbb{U}_{\cdot j})$. Z tohoto výrazu vyplývá, že součin dvou sloupců je 1 jen tehdy, když jde o součin sloupce se sebou samým, což odpovídá definici ortonormality. Analogie platí pro řádky. □

1.1.5 Trojúhelníkové matice

Definice 1.33. Nechť $\mathbb{A} \in \mathbb{C}^{n,n}$. Matici $\mathbb{A}$ nazvu **dolní trojíhelníkovou** právě tehdy když $\forall\, i, j \in \hat{n}$, $j > i$ platí $a_{ij} = 0$, tj.

$$\mathbb{A} = \begin{pmatrix} a_{11} & 0 & \dots & 0 \\ a_{21} & a_{22} & \ddots & \vdots \\ \vdots & \vdots & \ddots & 0 \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix}.$$

Naopak $\mathbb{A}$ nazvu **horní trojúhelníkovou** právě tehdy když $\forall\, i, j \in \hat{n}$, $j <\ i$ platí $a_{ij} = 0$, tj.

$$\mathbb{A} = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ 0 & a_{22} & \dots & a_{2n} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & a_{nn} \end{pmatrix}.$$

Věta 1.34 (součin trojúhelníkových matic)**.** Nechť $\mathbb{A}, \mathbb{B} \in \mathbb{C}^{n,n}$ jsou dolní, resp. horní trojíhelníkové matice. Potom matice $\mathbb{AB}$ je dolní, resp. horní trojúhelníková. Přitom má na diagonále součin odpovídajících diagonálních prvků.

Důkaz. Označme si matici $\mathbb{AB} := \mathbb{C}$. Chceme dokázat, že pro $j > i$ je nutně $\mathbb{C}_{ij} = 0$. Z definice maticového násobení platí

$$\mathbb{C}_{ij} = \sum_{k=1}^{n} \mathbb{A}_{ik}\mathbb{B}_{kj}$$

Víme, že $\mathbb{A}_{ik} = 0$ pro $i < k$ (tj. pro $k = i+1, ..., n$), protože $\mathbb{A}$ je dolní trojúhelníková. Stejně tak víme, že $\mathbb{B}_{kj} = 0$ pro $k < j$ (tj. pro $k = 1, ..., j-1$, což může nejhůře dopadnout jako $k = 1, ..., i$, protože hodnotu $\mathbb{C}_{ij}$ určujeme v bodě, kde $i < j$). Dohromady dostáváme, že $\mathbb{C}_{ij}$ bude 0 je-li $i < j$. Pro C_{ii} stejnými úvahami dojdeme k tomu, že

$$\mathbb{C}_{ii} = \sum_{k=1}^{n} \mathbb{A}_{ik}\mathbb{B}_{ki} = \mathbb{A}_{ii}\mathbb{B}_{ii}$$

□

Věta 1.35 (inverzní matice k trojúheníkové matici)**.** Nechť $\mathbb{A} \in \mathbb{C}^{n,n}$ je regulární dolní, resp. horní trojúhelníková matice. Potom $\mathbb{A}^{-1}$ je dolní, resp. horní trojúhelníková matice a její diagonální prvky jsou převrácené hodnoty původních diagonálních prvků.

Důkaz. Předpokládejme, že $\mathbb{A}$ je dolní trojúhelníková a $\mathbb{A}^{-1}$ je k ní inverzní. Pro spor budeme dále předpokládat, že $\mathbb{A}^{-1}$ není dolní trojíhelníková, tzn. $\exists\, i, j \in \hat{n}$, $i < j$ takové, že $\mathbb{A}^{-1}_{ij} \neq 0$. Volme $i, j \in \hat{n}$ tak, že pro dané i najdeme prvek $\mathbb{A}^{-1}_{ij}$ s co nejvyšším j. (Jinými slovy volme poslední nenulový prvek v i-tém řádku.)

$$\mathbb{I}_{ij} = \left[\mathbb{A}^{-1}\mathbb{A}\right]_{ij} = \sum_{k=1}^{n} \mathbb{A}^{-1}_{ik}\mathbb{A}_{kj} = \mathbb{A}^{-1}_{i1}\mathbb{A}_{1j} + \mathbb{A}^{-1}_{i2}\mathbb{A}_{2j} + \dots + \mathbb{A}^{-1}_{ij}\mathbb{A}_{jj} + \dots + \mathbb{A}^{-1}_{in}\mathbb{A}_{nj}$$

Nejdříve si uvědomme, že prvek $\mathbb{A}_{jj}$ je jistě nenulový. Kdyby byl nenulový, pak má trojúhelníková matice $\mathbb{A}$ na diagonále nulu, nemůže být regulární a nemůže tak existovat k ní inverzní. Prvek $\mathbb{A}^{-1}_{ij}$ jsme volili tak, aby byl

nenulový. Dohromady tedy $\mathbb{A}_{ij}^{-1}\mathbb{A}_{jj} \neq 0$. Protože $\mathbb{A}_{ij}^{-1}$ byl poslední nenulový v i-tém řádku matice $\mathbb{A}^{-1}$, členy sumy napravo od něj budou nulové, protože $\mathbb{A}_{ik}^{-1} = 0$ pro $k = j+1, ..., n$. Naopak $\mathbb{A}_{jj}$ je první nenulový v j-tém sloupci matice $\mathbb{A}$, (protože $\mathbb{A}$ je dolní trojúhelníková a $\mathbb{A}_{jj}$ je diagonální prvek). Členy sumy nalevo od něj tedy budou nulové. Dostáváme

$$\mathbb{I}_{ij} = 0 + \mathbb{A}_{ij}^{-1}\mathbb{A}_{jj} + 0 \neq 0$$

To je spor s vlastnostmi jednotkové matice $\mathbb{I}$. Druhá část věty pak vyplývá z věty 1.34 když se zaměříme na výraz $\mathbb{A}\mathbb{A}^{-1} = \mathbb{I}$. □

Trojúhelníkové matice jsou v numerice důležité z důvodu paměťové efektivity. Pokud víme, že budeme pracovat jen s trojúhelníkovými maticemi, můžeme ušetřit polovinu paměti. Nyní navíc víme, že operace jako součin a inverze trojúhelníkovost zachovávají.

1.1.6 Vlastní čísla a podobnost

Definice 1.36. Nechť je dána matice $\mathbb{A} \in \mathbb{C}^{n,n}$. Číslo $\lambda \in \mathbb{C}$ nazveme **vlastním číslem** *(eigenvalue)* matice $\mathbb{A}$, pokud existuje vektor $\vec{x} \in \mathbb{C}^n, \vec{x} \neq \vec{0}$, takový, že $\mathbb{A}\vec{x} = \lambda\vec{x}$. Vektor $\vec{x}$ nazveme **vlastním vektorem** matice $\mathbb{A}$ příslušným vlastnímu číslu λ. Množinu vlastních čísel matice $\mathbb{A}$ nazveme **spektrem** matice $\mathbb{A}$ a značíme $\sigma(\mathbb{A})$. Číslo $\rho(\mathbb{A})$ definované

$$\rho(\mathbb{A}) = \max_{\lambda \in \sigma(\mathbb{A})} |\lambda|$$

nazýváme **spektrálním poloměrem** matice $\mathbb{A}$.

Definice 1.37. Nechť $\mathbb{A} \in \mathbb{C}^{n,n}$. Zobrazení $p_{\mathbb{A}} : \mathbb{C} \to \mathbb{C}$ definované $\forall t \in \mathbb{C}$ jako $p_{\mathbb{A}}(t) = \det(\mathbb{A} - t\mathbb{I})$ nazýváme charakteristickým polynomem matice $\mathbb{A}$.

Věta 1.38 (vlastní čísla a charakteristický polynom)**.** Nechť $\mathbb{A} \in \mathbb{C}^{n,n}$. Pak $\lambda \in \sigma(\mathbb{A})$ právě tehdy když $p_{\mathbb{A}}(\lambda) = 0$. Jinými slovy, λ je vlastním číslem matice $\mathbb{A}$ právě tehdy když λ je kořenem charakteristického polynomu.

Důkaz. $\lambda \in \sigma(\mathbb{A}) \iff \exists\, \vec{x} \in \mathbb{C}^n, \vec{x} \neq \vec{0}$ takové, že $\mathbb{A}\vec{x} = \lambda\vec{x} \iff \exists\, \vec{x} \in \mathbb{C}^n, \vec{x} \neq \vec{0}$ takové, že $(\mathbb{A} - \lambda\mathbb{I})\vec{x} = \vec{0} \iff$ homogenní soustava s maticí $(\mathbb{A} - \lambda\mathbb{I})$ má netriviání řešení $\iff$ matice $(\mathbb{A} - \lambda\mathbb{I})$ je singulární $\iff \det(\mathbb{A} - \lambda\mathbb{I}) = 0 \iff p_{\mathbb{A}}(\lambda) = 0$. □

Věta 1.39 (vlastní čísla trojúhelníkové matice)**.** Nechť $\mathbb{A} \in \mathbb{C}^{n,n}$ je dolní, resp. horní trojúhelníková matice. Pak jsou její vlastní čísla rovna jejím diagonálním prvkům.

Důkaz. Plyne z věty 1.8 o determinantu trojúhelníkové matice a věty 1.38 o vztahu vlastních čísel a charakteristického polynomu. Nezkouší se. □

Definice 1.40. Nechť $\mathbb{A} \in \mathbb{C}^{n,n}$ a $\lambda \in \sigma(\mathbb{A})$. **Algebraickou násobností** $\nu_a(\lambda)$ vlastního čísla λ nazvme násobnost λ jakožto kořene charakteristického polynomu $p_{\mathbb{A}}$. **Geometrickou násobností** $\nu_g(\lambda)$ nazveme počet lineárně nezávislých vlastních vektorů příslušných k vlastnímu číslu λ.

Definice 1.41. Nechť $\mathbb{A}, \mathbb{B} \in \mathbb{C}^{n,n}$. Matici $\mathbb{A}$ nazvu **podobnou** matici $\mathbb{B}$ právě tehdy, když existuje regulární matice $\mathbb{X}$ řádu n taková, že $\mathbb{A} = \mathbb{X}^{-1}\mathbb{B}\mathbb{X}$. Tuto operaci nazýváme podobnostní transformací maticí $\mathbb{X}$.

Poznámka 1.42. Podobnost je ekvivalence na množině čtvercových matic řádu n, tj. pro $\mathbb{A}, \mathbb{B}, \mathbb{M} \in \mathbb{C}^{n,n}$ platí

1. Reflexivita: $\mathbb{A}$ je podobná sáma sobě,
2. Symetrie: je-li $\mathbb{A}$ podobná $\mathbb{B}$, potom je i $\mathbb{B}$ podobná $\mathbb{A}$,
3. Tranzitivita: je-li $\mathbb{A}$ podobná $\mathbb{B}$ a $\mathbb{B}$ podobná $\mathbb{M}$, je i $\mathbb{A}$ podobná $\mathbb{M}$.

Důkaz. Důkaz plyne jednoduše z definice podobnosti. Trvzení o symetrii se např. dokáže snadnými ekvivalentními úpravami: $\mathbb{A} = \mathbb{X}^{-1}\mathbb{B}\mathbb{X} \iff \mathbb{X}\mathbb{A} = \mathbb{B}\mathbb{X} \iff \mathbb{X}\mathbb{A}\mathbb{X}^{-1} = \mathbb{B}$. Tentokrát je maticí podobnostní transformace matice $\mathbb{X}^{-1}$. □

Věta 1.43 (vlastnosti podobných matic)**.** Nechť $\mathbb{A}, \mathbb{B} \in \mathbb{C}^{n,n}$, $\mathbb{A}$ je podobná $\mathbb{B}$. Potom platí

1. $p_{\mathbb{A}} = p_{\mathbb{B}}$, tudíž i $\sigma(\mathbb{A}) = \sigma(\mathbb{B})$ a $\nu_a^{\mathbb{A}}(\lambda) = \nu_a^{\mathbb{B}}(\lambda)$ pro každé $\lambda \in \sigma(\mathbb{A})$,
2. $\nu_g^{\mathbb{A}}(\lambda) = \nu_g^{\mathbb{B}}(\lambda)$ pro každé $\lambda \in \sigma(\mathbb{A})$,
3. $\det \mathbb{A} = \det \mathbb{B}$

Důkaz. Vlastnosti plynou triviálně z definic jednotlivých objektů, viz strana 47 [2]. Nezkouší se. □

Poznámka 1.44. Implikaci ve větě 1.43 o vlastnostech podobných matic nelze obrátit, tj. i přes to, že mají dvě matice stejné charakteristické polynomy, stejná spektra i stejné algebraické a geometrické násobnosti, nemusí existovat regulární matice $\mathbb{X}$ zprostředkovávající podobnostní transformaci.

Definice 1.45. Nechť $\mathbb{A} \in \mathbb{C}^{n,n}$. Matici $\mathbb{A}$ nazveme **diagonalizovatelnou**, pokud je podobná diagonální matici, tj. existuje diagonální matice $\mathbb{D}$ a regulární matice $\mathbb{X}$ takové, že $\mathbb{A} = \mathbb{X}\mathbb{D}\mathbb{X}^{-1}$.

Věta 1.46 (diagonalizovatelnost a násobnosti)**.** Nechť $\mathbb{A} \in \mathbb{C}^{n,n}$. Potom matice $\mathbb{A}$ je diagonalizovatelná právě tehdy když $\forall\, \lambda \in \sigma(\mathbb{A})$ platí $\nu_a(\lambda) = \nu_g(\lambda)$.

Důkaz. Důkaz tohoto tvrzení vyžaduje větší teoretickou přípravu. Lze ho zapsat jako důsledek vět 3.19 a 3.28 z [2]. Nezkouší se. □

Věta 1.47. Nechť $\mathbb{A} \in \mathbb{C}^{n,n}$ a $\lambda_1, ..., \lambda_p$ jsou všechna její vzájemně různá vlastní čísla. Pak existuje regulární matice $\mathbb{X}$ taková, že $\mathbb{A}$ je podobná matici $\mathbb{J}$ v Jordanově kanonickém tvaru, kde

$$\mathbb{J} = \begin{pmatrix} \mathbb{J}_1^{(1)} & & & & & & & & \\ & \ddots & & & & & & & \\ & & \mathbb{J}_{s_1}^{(1)} & & & & & & \\ & & & \mathbb{J}_1^{(2)} & & & & & \\ & & & & \ddots & & & & \\ & & & & & \mathbb{J}_{s_2}^{(2)} & & & \\ & & & & & & \ddots & & \\ & & & & & & & \mathbb{J}_1^{(p)} & \\ & & & & & & & & \ddots \\ & & & & & & & & & \mathbb{J}_{s_p}^{(p)} \end{pmatrix}.$$

Tzv. Jordanovy bloky $\mathbb{J}_i^k$ pro $k \in \hat{p}, i \in \hat{s}_k$ jsou tvaru

$$\mathbb{J}_i^k = \begin{pmatrix} \lambda_k & 1 & & \\ & \lambda_k & \ddots & \\ & & \ddots & 1 \\ & & & \lambda_k \end{pmatrix}.$$

Přitom matice $\mathbb{J}$ je až na pořadí bloků dána jednoznačně.

Poznámka 1.48. Věta o Jordanově kanonickém tvaru nám jistým způsobem rozšiřuje větu 1.46 o diagonalizovatelnosti a násobnostech. Ne každá čtvercová matice je podobná diagonální matici, každá čtvercová matice je ale podobná matici v Jordanově kanonickém tvaru. Je-li matice přímo podobná diagonální matici, pak jsou Jordanovy bloky jednoprvkové matice obsahující pouze vlastní čísla a vymizí jedičky nad diagonálou.

1.2 Rozklady matic

Věta 1.49 (o rozkladu na dolní a horní trojúhelníkovou matici)**.** Každou silně regulární matici $\mathbb{A}$ lze jednoznačným způsobem vyjádřit ve tvaru $\mathbb{A} = \mathbb{L}\mathbb{D}\mathbb{R}$, kde $\mathbb{L}$ je (levá) dolní trojúhelníková matice s 1 na diagonále, $\mathbb{R}$ je (pravá) horní trojúhelníková matice s 1 na diagonále a $\mathbb{D}$ je diagonální matice.

Důkaz. Budeme postupovat indukcí, nechť tedy $n = 1$. Potom $\mathbb{A} = \begin{pmatrix} a_{11} \end{pmatrix} = 1 \cdot a_{11} \cdot 1$, kde $\mathbb{L} = \begin{pmatrix} 1 \end{pmatrix}$, $\mathbb{D} = \begin{pmatrix} a_{11} \end{pmatrix}$ a $\mathbb{R} = \begin{pmatrix} 1 \end{pmatrix}$, což jsou matice splňující požadavky věty. Nyní provedeme indukční krok od $n-1 \to n$. Označme si

$$\mathbb{A} = \begin{pmatrix} \tilde{\mathbb{A}} & \vec{v} \\ \vec{u}^T & \alpha \end{pmatrix}$$

kde $\tilde{\mathbb{A}} \in \mathbb{C}^{n-1,n-1}$. Podle indukčního předpokladu umíme vytvořit rozklad $\tilde{\mathbb{A}} = \tilde{\mathbb{L}}\tilde{\mathbb{D}}\tilde{\mathbb{R}}$. Ten můžeme zapsat ve tvaru součinu matic

$$\mathbb{A} = \mathbb{L}\mathbb{D}\mathbb{R}$$

$$\begin{pmatrix} \tilde{\mathbb{A}} & \vec{v} \\ \vec{u}^T & \alpha \end{pmatrix} = \begin{pmatrix} \tilde{\mathbb{L}} & \vec{0} \\ \vec{l}^T & 1 \end{pmatrix} \begin{pmatrix} \tilde{\mathbb{D}} & \vec{0} \\ \vec{0}^T & \delta \end{pmatrix} \begin{pmatrix} \tilde{\mathbb{R}} & \vec{r} \\ \vec{0}^T & 1 \end{pmatrix} = \begin{pmatrix} \tilde{\mathbb{L}} & \vec{0} \\ \vec{l}^T & 1 \end{pmatrix} \begin{pmatrix} \tilde{\mathbb{D}}\tilde{\mathbb{R}} & \tilde{\mathbb{D}}\vec{r} \\ \vec{0}^T & \delta \end{pmatrix} = \begin{pmatrix} \tilde{\mathbb{L}}\tilde{\mathbb{D}}\tilde{\mathbb{R}} & \tilde{\mathbb{L}}\tilde{\mathbb{D}}\vec{r} \\ \vec{l}^T\tilde{\mathbb{D}}\tilde{\mathbb{R}} & \vec{l}^T\tilde{\mathbb{D}}\vec{r} + \delta \end{pmatrix},$$

kde $\vec{l}^T, \delta$ a $\vec{r}$ neznáme. Postupně si je tedy vyjádříme

$$\begin{aligned} \vec{v} = \tilde{\mathbb{L}}\tilde{\mathbb{D}}\vec{r} &\implies \vec{r} = \tilde{\mathbb{D}}^{-1}\tilde{\mathbb{L}}^{-1}\vec{v}, \\ \vec{u}^T = \vec{l}^T\tilde{\mathbb{D}}\tilde{\mathbb{R}} &\implies \vec{l}^T = \vec{u}^T\tilde{\mathbb{D}}^{-1}\tilde{\mathbb{R}}^{-1}, \\ \alpha = \vec{l}^T\tilde{\mathbb{D}}\vec{r} + \delta &\implies \delta = \alpha - \vec{l}^T\tilde{\mathbb{D}}\vec{r}. \end{aligned}$$

Vyjdeme-li z indukčního předpokladu, tj. že známe rozklad $\tilde{\mathbb{A}} = \tilde{\mathbb{L}}\tilde{\mathbb{D}}\tilde{\mathbb{R}}$, získáváme všechny neznámé a umíme spočítat rozklad $\mathbb{A} = \mathbb{L}\mathbb{D}\mathbb{R}$, což jsme chtěli dokázat. Takový rozklad tedy jistě existuje, nyní dokážeme jeho jednoznačnost. Nechť existují dva různé rozklady.

$$\begin{aligned} \mathbb{A} &= \mathbb{L}\mathbb{D}\mathbb{R} = \mathbb{L}'\mathbb{D}'\mathbb{R}' \\ &= \mathbb{L}'^{-1}\mathbb{L}\mathbb{D} = \mathbb{D}'\mathbb{R}'\mathbb{R}'^{-1} \end{aligned}$$

Je vhodné poznamenat, že inverzní matice určitě existují, protože jde o trojúhelníkové matice s 1 na diagonále, které jsou tudíž jistě regulární. Na levé straně rovnice máme dolní trojúhelníkové matice, na pravé straně horní trojúhelníkové matice. Protože se tyto strany musí rovnat, musí součiny na každé straně dát diagonální matici. Zároveň víme, že $\mathbb{L}'^{-1}, \mathbb{L}, \mathbb{R}'$ a $\mathbb{R}'^{-1}$ mají na diagonále jen jedničky, proto prvky na diagonále ovlivňují jen $\mathbb{D}$ a $\mathbb{D}'$. Dostáváme rovnost $\mathbb{D} = \mathbb{D}'$. Tento postup opakujeme.

$$\begin{aligned} \mathbb{A} &= \mathbb{L}\mathbb{D}\mathbb{R} = \mathbb{L}'\mathbb{D}'\mathbb{R}' \\ &= \mathbb{L}'^{-1}\mathbb{L}\mathbb{D} = \mathbb{D}'\mathbb{R}'\mathbb{R}'^{-1} \\ &= \mathbb{L}'^{-1}\mathbb{L} = \mathbb{D}'\mathbb{R}'\mathbb{R}'^{-1}\mathbb{D}^{-1} \end{aligned}$$

Stejnou úvahou jako výše zjistíme, že na obou stranách rovnice musíme mít diagonální matice. Protože navíc $\mathbb{L}'^{-1}$ i $\mathbb{L}$ mají na diagonále 1, platí $\mathbb{L}'^{-1}\mathbb{L} = \mathbb{I}$, odkud už zřejmě $\mathbb{L} = \mathbb{L}'$. Čtenář analogicky dokáže rovnost $\mathbb{R}$ a $\mathbb{R}'$. □

Poznámka 1.50. LDR rozklad není podobnostní transformace, tudíž na diagonále $\mathbb{D}$ **nejsou** vlastní čísla $\mathbb{A}$. Kdyby mělo jít o podobnostní transformaci, muselo by platit, že $\mathbb{R}$ je inverzní k $\mathbb{L}$, tedy že $\mathbb{L} = \mathbb{R}^{-1}$. Podle věty 1.35 by tak ale buď obě matice musely být horní trojúhelníkové, nebo obě dolní trojúhelníkové. Protože mají 1 na diagonále, muselo by jít o identitu a v takovém případě by nám věta jen říkala, že $\mathbb{A} = \mathbb{D}$, tzn. že $\mathbb{A}$ už je v

diagonálním tvaru, což není moc užitečná informace.

Definice 1.51. Householderovou reflekční maticí (elementární unitární maticí) nazeme každou matici $\mathbb{H}_{\vec{w}}$ tvaru

$$\mathbb{H}_{\vec{w}} = \mathbb{I} - 2\vec{w}\vec{w}^*,$$

kde $\vec{w}$ je **Householderův vektor**, pro který platí $||\vec{w}||_2 = \sqrt{(\vec{w},\vec{w})} = 1$.

Poznámka 1.52. Výraz $\vec{w}\vec{w}*$ se dá přepsat jako

$$\vec{w}\vec{w}^* = \begin{pmatrix} w_1 \\ \vdots \\ w_n \end{pmatrix} \begin{pmatrix} \overline{w_1} & \dots & \overline{w_n} \end{pmatrix} = \begin{pmatrix} w_1\overline{w}_1 & \dots & w_1\overline{w}_n \\ \vdots & \ddots & \vdots \\ w_n\overline{w}_1 & \dots & w_n\overline{w}_n \end{pmatrix}.$$

Pro ij-tý prvek tedy platí $(\vec{w}\vec{w}^*)_{ij} = w_i\overline{w_j}$.

Věta 1.53. Householderova reflekční matice je hermitovská a unitární.

Důkaz. Postupně dokážeme obě vlastnosti.

- *hermitovskost* $(\mathbb{H}_{\vec{w}} \stackrel{?}{=} \mathbb{H}^*_{\vec{w}})$: $\mathbb{H}^*_{\vec{w}} = \mathbb{I}^* - 2(\vec{w}\vec{w}^*)^* = \mathbb{I} - 2(\vec{w}^*)^*\vec{w}^* = \mathbb{I} - 2\vec{w}\vec{w}* = \mathbb{H}_{\vec{w}}$
- *unitarita* $(\mathbb{H}^{-1}_{\vec{w}} \stackrel{?}{=} \mathbb{H}^*_{\vec{w}})$: Dokážeme, že $\mathbb{I} = \mathbb{H}_{\vec{w}}\mathbb{H}^*_{\vec{w}} = \mathbb{H}^*_{\vec{w}}\mathbb{H}_{\vec{w}}$. Potom z toho, co víme o inverzních maticích bude jistě platit, že $\mathbb{H}^*_{\vec{w}} = \mathbb{H}^{-1}_{\vec{w}}$.

  $$\mathbb{H}_{\vec{w}}\mathbb{H}^*_{\vec{w}} \stackrel{*}{=} \mathbb{H}_{\vec{w}}\mathbb{H}_{\vec{w}} = (\mathbb{I} - 2\vec{w}\vec{w}^*)(\mathbb{I} - 2\vec{w}\vec{w}^*) = \mathbb{I} - 4\vec{w}\vec{w}^* + 4\vec{w}\vec{w}^*\vec{w}\vec{w}^* \stackrel{**}{=} \mathbb{I} - 4\vec{w}\vec{w}^* + 4\vec{w}\vec{w}^* = \mathbb{I},$$

  kde jsme v rovnosti označené * využili již dokázanou hermitovskost. Rovnost označená ** vyplývá z poznámky 1.52 resp. z definice Householderova vektoru a jeho normalizace. Zřejmě $\vec{w}^*\vec{w} = (\vec{w},\vec{w}) = 1$.

□

Věta 1.54. Nechť $\mathbb{U} \in \mathbb{C}^{n,n}$ je unitární matice. Pak $||\mathbb{U}\vec{x}||_2 = ||\vec{x}||_2$. Jinými slovy unitární matice zachovávají normu.

Důkaz. $||\mathbb{U}\vec{x}||_2^2 = (\mathbb{U}\vec{x},\mathbb{U}\vec{x}) = (\vec{x},\mathbb{U}^*\mathbb{U}\vec{x}) = (\vec{x},\mathbb{U}^{-1}\mathbb{U}\vec{x}) = (\vec{x},\mathbb{I}\vec{x}) = (\vec{x},\vec{x}) = ||\vec{x}||_2^2$ □

Věta 1.55 (význam Householderovy reflekční matice)**.** Nechť $\mathbb{H}_{\vec{w}}$ je Householderova reflekční matice a $\vec{v} \in \mathbb{C}^n$. Pak vektor $\mathbb{H}_{\vec{w}}\vec{v}$ je zrcadlový obraz vektoru $\vec{v}$ podle nadroviny $L \coloneqq \{\vec{x} \in \mathbb{C}^n \mid \vec{w}^*\vec{x} = (\vec{x},\vec{w}) = 0\}$ v tom smyslu, že platí

1. $||\mathbb{H}_{\vec{w}}\vec{v}||_2 = ||\vec{v}||_2$,
2. $(\mathbb{H}_{\vec{w}}\vec{v} + \vec{v}) \in L$,
3. $(\mathbb{H}_{\vec{w}}\vec{v} - \vec{v}) \perp L$.

Důkaz 1. Matice $\mathbb{H}_{\vec{w}}$ je unitární, tudíž jde o důsledek věty 1.54. □

Důkaz 2.

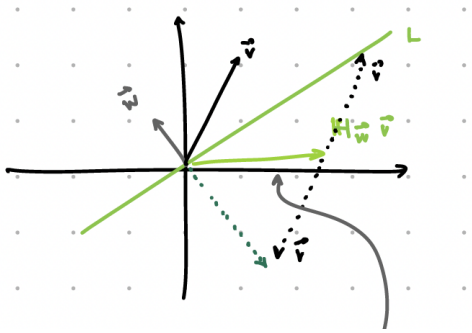
$$\begin{aligned}
(\mathbb{H}_{\vec{w}}\vec{v}+\vec{v}) \in L &\iff (\mathbb{H}_{\vec{w}}\vec{v}+\vec{v},\vec{w}) = 0,\\
((\mathbb{I}-2\vec{w}\vec{w}^*)\vec{v}+\vec{v},\vec{w}) &= 0,\\
(\vec{v}-2\vec{w}\vec{w}^*\vec{v}+\vec{v},\vec{w}) &= 0,\\
(2\vec{v}-2\vec{w}(\vec{v},\vec{w}),\vec{w}) &= 0,\\
2(\vec{v},\vec{w})-2(\vec{w},\vec{w})(\vec{v},\vec{w}) &= 0,\\
2(\vec{v},\vec{w})-2(\vec{v},\vec{w}) &= 0,\\
0 &= 0,
\end{aligned}$$

kde první ekvivalence vyplývá z definice nadroviny L. (Do té patří všechny vektory $\vec{x}$ kolmé na $\vec{w}$.) □

Důkaz 3.

$$\begin{aligned}
(\mathbb{H}_{\vec{w}}\vec{v}-\vec{v}) \perp L &\iff \forall\,\vec{x}\in L \text{ platí } (\mathbb{H}_{\vec{w}}\vec{v}-\vec{v},\vec{x}) = 0,\\
((\mathbb{I}-2\vec{w}\vec{w}^*)\vec{v}-\vec{v},\vec{x}) &= 0,\\
(\vec{v}-2\vec{w}\vec{w}^*\vec{v}-\vec{v},\vec{x}) &= 0,\\
-2(\vec{w}\vec{w}^*\vec{v},\vec{x}) &= 0,\\
-2\vec{x}^*\vec{w}\vec{w}^*\vec{v} &= 0,\\
-2(\vec{w},\vec{x})(\vec{v},\vec{w}) &= 0,\\
-2\cdot 0\cdot(\vec{v},\vec{w}) &= 0,\\
0 &= 0,
\end{aligned}$$

protože víme z definice nadroviny L, že $\forall\,\vec{x}\in L$ platí $(\vec{x},\vec{w}) = 0$. □



Obrázek 1.1: význam Householderovy reflekční matice

Poznámka 1.56. Protože platí $\mathbb{H}_{\vec{w}} = \mathbb{H}^*_{\vec{w}} = \mathbb{H}^{-1}_{\vec{w}}$, při druhé aplikace transformace dostáváme opět původní vektor ($\mathbb{H}_{\vec{w}}\mathbb{H}_{\vec{w}}\vec{v} = \mathbb{H}_{\vec{w}}\mathbb{H}^{-1}_{\vec{w}}\vec{v} = \mathbb{I}\vec{v} = \vec{v}$).

Věta 1.57. Nechť $\mathbb{A}\in\mathbb{C}^{n,n}$ a $\lambda\in\sigma(\mathbb{A})$. Potom $\exists\,\mathbb{H}_{\vec{w}}$ taková, že $\mathbb{H}_{\vec{w}}\mathbb{A}\mathbb{H}_{\vec{w}}\vec{e}_1 = \lambda\vec{e}_1$.

Důkaz. $\mathbb{H}_{\vec{w}}\mathbb{A}\mathbb{H}_{\vec{w}}\vec{e}_1 = \lambda\vec{e}_1 \implies \mathbb{A}\mathbb{H}_{\vec{w}}\vec{e}_1 = \lambda\mathbb{H}^{-1}_{\vec{w}}\vec{e}_1 = \lambda\mathbb{H}_{\vec{w}}\vec{e}_1 \implies \mathbb{H}_{\vec{w}}\vec{e}_1$ je vlastní vektor $\mathbb{A}$ příslušný vlastnímu číslu λ. Označme si jej $\vec{x} \coloneqq \mathbb{H}_{\vec{w}}\vec{e}_1$. Abychom dokázali existenci $\mathbb{H}_{\vec{w}}$ ukážeme, jak pro dané $\mathbb{A}$ a λ volit $\vec{w}$, přičemž pro dané λ známe $\vec{x}$. Z věty 1.55 o významu Householderovy reflekční matice víme, že $\forall\,\vec{y}\in\mathbb{C}^n$ platí

$(\mathbb{H}_{\vec{w}}\vec{y} - \vec{y}) \perp L$, tedy i $(\mathbb{H}_{\vec{w}}\vec{e}_1 - \vec{e}_1) \perp L$, resp. $(\vec{x} - \vec{e}_1) \perp L$. Volme tedy

$$\vec{w} \coloneqq \frac{\vec{x} - \vec{e}_1}{||\vec{x} - \vec{e}_1||_2}.$$

Takto zvolené $\vec{w}$ zřejmě splní podmínky z věty 1.55 a zároveň je normované na 1, což požaduje definice Householderova vektoru. □

Poznámka 1.58. Podívejme se nejdřív na to, co vlastně výraz $\mathbb{H}_{\vec{w}}\mathbb{A}\mathbb{H}_{\vec{w}}\vec{e}_1$ dělá. Označme si matici $\mathbb{H}_{\vec{w}}\mathbb{A}\mathbb{H}_{\vec{w}} \coloneqq \mathbb{A}'$. Všiměme si, že jde o podobnostní transformaci, jelikož z hermitovskosti a unitarity $\mathbb{H}_{\vec{w}}$ platí $\mathbb{H}_{\vec{w}} = \mathbb{H}^*_{\vec{w}} = \mathbb{H}^{-1}_{\vec{w}}$ a proto $\mathbb{A}' = \mathbb{H}_{\vec{w}}\mathbb{A}\mathbb{H}_{\vec{w}} = \mathbb{H}_{\vec{w}}\mathbb{A}\mathbb{H}^{-1}_{\vec{w}}$. Matice $\mathbb{A}'$ má tedy na diagonále vlastní čísla $\mathbb{A}$. Z definice maticového násobebí je dále zřejmé, že součin s prvním vektorem standardní báze „vyselektuje“ první sloupec matice, tzn. platí

$$\mathbb{A}'\vec{e}_1 = \begin{pmatrix} \lambda \\ 0 \\ \vdots \\ 0, \end{pmatrix}.$$

Pro součin $\mathbb{H}_{\vec{w}}\mathbb{A}\mathbb{H}_{\vec{w}}$ tím pádem ale musí platit, že výsledná matice má tvar

$$\mathbb{H}_{\vec{w}}\mathbb{A}\mathbb{H}_{\vec{w}} = \begin{pmatrix} \lambda & a'_{12} & \dots & a'_{1n} \\ 0 & a'_{22} & \dots & a'_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & a'_{n2} & \dots & a'_{nn} \end{pmatrix}.$$

Bez použití Gaussovy eliminace se nám povedlo zbavit se čísel v prvním sloupci původní matice $\mathbb{A}$. To je velmi užitečné a bude se nám v budoucnu hodit.

Věta 1.59 (Schurova)**.** Nechť $\mathbb{A} \in \mathbb{C}^{n,n}$. Potom existuje unitární matice $\mathbb{U} \in \mathbb{C}^{n,n}$ taková, že $\mathbb{A} = \mathbb{U}^*\mathbb{R}\mathbb{U}$, kde $\mathbb{R}$ je horní trojúhelníková.

Důkaz. $\mathbb{A} = \mathbb{U}^*\mathbb{R}\mathbb{U} \iff \mathbb{U}\mathbb{A} = \mathbb{R}\mathbb{U} \iff \mathbb{U}\mathbb{A}\mathbb{U}^* = \mathbb{R}$, kde již bez rozepsání mezikroků využíváme unitaritu $\mathbb{U}$, tj. že $\mathbb{U}^* = \mathbb{U}^{-1}$. Podle věty 1.57 víme, že $\exists\, \mathbb{H}_{\vec{w}1}$ taková, že $\mathbb{H}_{\vec{w}1}\mathbb{A}\mathbb{H}_{\vec{w}1}\vec{e}_1 = \lambda_1\vec{e}_1$ a následně podle poznámky 1.58, že

$$\mathbb{H}_{\vec{w}1}\mathbb{A}\mathbb{H}_{\vec{w}1} = \begin{pmatrix} \lambda_1 & \vec{v}_1^T \\ \vec{0} & \tilde{\mathbb{A}}_1 \end{pmatrix} \in \mathbb{C}^{n,n}, \quad \text{kde } \vec{v}_1 \in \mathbb{C}^{n-1} \text{ a } \tilde{\mathbb{A}}_1 \in \mathbb{C}^{n-1,n-1}$$

Na matici $\tilde{\mathbb{A}}_1$ můžeme větu 1.57 aplikovat znovu, tj. víme, že $\exists\, \tilde{\mathbb{H}}_{\vec{w}2}$ taková, že

$$\tilde{\mathbb{H}}_{\vec{w}2}\tilde{\mathbb{A}}_1\tilde{\mathbb{H}}_{\vec{w}2} = \begin{pmatrix} \lambda_2 & \vec{v}_2^T \\ \vec{0} & \tilde{\mathbb{A}}_2 \end{pmatrix} \in \mathbb{C}^{n-1,n-1}, \quad \text{kde } \vec{v}_2 \in \mathbb{C}^{n-2} \text{ a } \tilde{\mathbb{A}}_2 \in \mathbb{C}^{n-2,n-2}$$

Potom zadefinujeme-li si matici $\mathbb{H}_{\vec{w}2}$ jako

$$\mathbb{H}_{\vec{w}2} = \begin{pmatrix} 1 & \vec{0}^T \\ \vec{0} & \tilde{\mathbb{H}}_{\vec{w}2} \end{pmatrix} \in \mathbb{C}^{n,n},$$

můžeme analyzovat součin $\mathbb{H}_{\vec{w}2}\mathbb{H}_{\vec{w}1}\mathbb{A}\mathbb{H}_{\vec{w}1}\mathbb{H}_{\vec{w}2}$:

$$\mathbb{H}_{\vec{w}2}\mathbb{H}_{\vec{w}1}\mathbb{A}\mathbb{H}_{\vec{w}1}\mathbb{H}_{\vec{w}2} = \begin{pmatrix} 1 & \vec{0}^T \\ \vec{0} & \tilde{\mathbb{H}}_{\vec{w}2} \end{pmatrix} \begin{pmatrix} \lambda_1 & \vec{v}_1^T \\ \vec{0} & \tilde{\mathbb{A}}_1 \end{pmatrix} \begin{pmatrix} 1 & \vec{0}^T \\ \vec{0} & \tilde{\mathbb{H}}_{\vec{w}2} \end{pmatrix} = \begin{pmatrix} \lambda_1 & \vec{v}_1^T \\ \vec{0} & \tilde{\mathbb{H}}_{\vec{w}2}\tilde{\mathbb{A}}_1 \end{pmatrix} \begin{pmatrix} 1 & \vec{0}^T \\ \vec{0} & \tilde{\mathbb{H}}_{\vec{w}2} \end{pmatrix} =$$

$$= \begin{pmatrix} \lambda_1 & \vec{v}_1^T \\ \vec{0} & \tilde{\mathbb{H}}_{\vec{w}2}\tilde{\mathbb{A}}_1\tilde{\mathbb{H}}_{\vec{w}2} \end{pmatrix} = \begin{pmatrix} \lambda_1 & \vec{v}_1^T & \\ 0 & \lambda_2 & \vec{v}_2^T \\ \vec{0} & \vec{0} & \tilde{\mathbb{A}}_2 \end{pmatrix}.$$

Kdybychom nyní kroky opakovali, nakonec bychom dostali součin

$$\mathbb{H}_{\vec{w}n-1}\mathbb{H}_{\vec{w}n-2}\ldots\mathbb{H}_{\vec{w}2}\mathbb{H}_{\vec{w}1}\mathbb{A}\mathbb{H}_{\vec{w}1}\mathbb{H}_{\vec{w}2}\ldots\mathbb{H}_{\vec{w}n-2}\mathbb{H}_{\vec{w}n-1} \coloneqq \mathbb{R}.$$

Každá z matic $\mathbb{H}_{\vec{w}i}$ je unitární a podle věty 1.31 je i součin unitárních matic unitární. Proto můžeme jako naše $\mathbb{U}$ do věty volit

$$\mathbb{U} \coloneqq \mathbb{H}_{\vec{w}n-1}\ldots\mathbb{H}_{\vec{w}1}.$$

Stačí už jen dokázat, že $\mathbb{H}_{\vec{w}n-1}\ldots\mathbb{H}_{\vec{w}1}$ a $\mathbb{H}_{\vec{w}1}\ldots\mathbb{H}_{\vec{w}n-1}$ jsou vzájemě inverzní. To ale triviálně plyne opět z unitarity $\mathbb{H}_{\vec{w}i}$, protože

$$\mathbb{H}_{\vec{w}n-1}\ldots\mathbb{H}_{\vec{w}2}\mathbb{H}_{\vec{w}1}\mathbb{H}_{\vec{w}1}\mathbb{H}_{\vec{w}2}\ldots\mathbb{H}_{\vec{w}n-1} = \mathbb{H}_{\vec{w}n-1}\ldots\mathbb{H}_{\vec{w}2}\mathbb{I}\mathbb{H}_{\vec{w}1}\mathbb{H}_{\vec{w}2}\ldots\mathbb{H}_{\vec{w}n-1} =$$

$$= \mathbb{H}_{\vec{w}n-1}\ldots\mathbb{H}_{\vec{w}2}\mathbb{H}_{\vec{w}1}\mathbb{H}_{\vec{w}2}\ldots\mathbb{H}_{\vec{w}n-1} = \ldots = \mathbb{I}.$$

Dohromady tedy dostáváme $\mathbb{U}\mathbb{A}\mathbb{U}^{-1} = \mathbb{R}$ a na začátku důkazu jsme si ukázali, že je to ekvivalentní se zněním věty, tedy $\mathbb{A} = \mathbb{U}^*\mathbb{R}\mathbb{U}$. □

Poznámka 1.60. Je vhodné si všimnout, že tvrzení Schurovy věty je podobnostní transformace, protože $\mathbb{A} = \mathbb{U}^*\mathbb{R}\mathbb{U} = \mathbb{U}^{-1}\mathbb{R}\mathbb{U}$. Důležité je, že tato věta platí pro libovolné čtvercové matice, tzn. i ty, které nejsou diagonalizovatelné. Pokutou za to je to, že nedostáváme podobnost s diagonální maticí ale jen s horní trojúhelníkovou maticí.

Poznámka 1.61. Vzhledem k tomu, že jde o podobnostní transformaci, která podle věty 1.43 zachovává vlastní čísla, musí mít matice horní trojúhelníková matice $\mathbb{R}$ podle věty 1.39 na diagonále vlastní čísla $\mathbb{A}$.

Lemma 1.62. Nechť $\mathbb{A} \in \mathbb{C}^{n,n}$ je normální trojúhelníková matice. Potom $\mathbb{A}$ je nutně diagonální.

Důkaz. Budeme postupovat indkucí. Nechť tedy $n = 1$. Potom $\mathbb{A} = \begin{pmatrix} a_{11} \end{pmatrix}$, což je diagonální matice. Nyní provedeme indukční krok od $n-1 \to n$. Bez újmy na obecnosti nechť $\mathbb{A}$ je dolní trojúhelníková matici. Tu můžeme zapsat ve tvaru

$$\mathbb{A} = \begin{pmatrix} \tilde{\mathbb{A}} & \vec{0} \\ \vec{l}^T & \alpha \end{pmatrix}, \quad \text{kde } \tilde{\mathbb{A}} \in \mathbb{C}^{n-1,n-1} \text{ a } \vec{l} \in \mathbb{C}^{n-1}.$$

Kdybychom dokázali, že $\tilde{\mathbb{A}}$ je normální, byla by z indukčního předpokladu diagonální. Pak ještě musíme dokázat, že $\vec{l} = \vec{0}$ a celá $\mathbb{A}$ bude diagonální. Z předpokladů věty víme, že $\mathbb{A}$ je normální, tudíž musí platit $\mathbb{A}\mathbb{A}^* = \mathbb{A}^*\mathbb{A}$. Rozepišme si tyto dva součiny.

$$\mathbb{A}\mathbb{A}^* = \begin{pmatrix} \tilde{\mathbb{A}} & \vec{0} \\ \vec{l}^T & \alpha \end{pmatrix} \begin{pmatrix} \tilde{\mathbb{A}}^* & \bar{\vec{l}} \\ \vec{0}^T & \overline{\alpha} \end{pmatrix} = \begin{pmatrix} \tilde{\mathbb{A}}\tilde{\mathbb{A}}^* & \tilde{\mathbb{A}}\bar{\vec{l}} \\ \vec{l}^T\tilde{\mathbb{A}}^* & \vec{l}^T\bar{\vec{l}} + \alpha\overline{\alpha} \end{pmatrix}$$

$$\mathbb{A}^*\mathbb{A} = \begin{pmatrix} \tilde{\mathbb{A}}^* & \bar{\vec{l}} \\ \vec{0}^T & \overline{\alpha} \end{pmatrix} \begin{pmatrix} \tilde{\mathbb{A}} & \vec{0} \\ \vec{l}^T & \alpha \end{pmatrix} = \begin{pmatrix} \tilde{\mathbb{A}}^*\tilde{\mathbb{A}} + \bar{\vec{l}}\vec{l}^T & \alpha\bar{\vec{l}} \\ \overline{\alpha}\vec{l}^T & \alpha\overline{\alpha} \end{pmatrix}$$

Aby platila nomrálnost $\mathbb{A}$, musí platit rovnost $\vec{l}^T\bar{\vec{l}} + \alpha\overline{\alpha} = \alpha\overline{\alpha}$. Ta nastane jedině za podmínky, že $\vec{l} = \vec{0}$, což je jedna z věcí, které jsme potřebovali ukázat. Součiny si také můžeme dále upravit.

$$\mathbb{A}\mathbb{A}^* = \begin{pmatrix} \tilde{\mathbb{A}}\tilde{\mathbb{A}}^* + & \vec{0}^T \\ \vec{0} & \alpha\overline{\alpha} \end{pmatrix}$$

$$\mathbb{A}^*\mathbb{A} = \begin{pmatrix} \tilde{\mathbb{A}}^*\tilde{\mathbb{A}} + & \vec{0}^T \\ \vec{0} & \alpha\overline{\alpha} \end{pmatrix}$$

Odtud ale vidíme, že matice $\tilde{\mathbb{A}}$ je rovněž normální, a podle indukčního předpokladu tedy také diagonální. Matice $\mathbb{A}$ je tudíž také diagonální. (TODO: lépe vysvětlit logiku důkazu) □

Důsledek 1.63. Nechť $\mathbb{A} \in \mathbb{C}^{n,n}$ je normální. Potom existuje unitární matice $\mathbb{U} \in \mathbb{C}^{n,n}$ taková, že $\mathbb{A} = \mathbb{U}^*\mathbb{D}\mathbb{U}$, kde $\mathbb{D}$ je diagonální matice. Je-li navíc $\mathbb{A}$ hermitovská, pak má $\mathbb{D}$ na diagonále realná čísla.

Důkaz. Z Schurovy věty (1.59) již víme, že pro každou matici $\mathbb{A} \in \mathbb{C}^{n,n}$ existuje unitární matice $\mathbb{U}$ taková, že platí $\mathbb{A} = \mathbb{U}^*\mathbb{D}\mathbb{U}$. Schurova věta nám nicméně o $\mathbb{D}$ řekla jen to, že je horní trojúhelníková. My budeme chtít navíc aplikovat lemma 1.62, potřebujeme ale nejdřív ukázat, že za předpokladů normálnosti $\mathbb{A}$ je i $\mathbb{D}$ normální. Pro $\mathbb{D}$ můžeme psát $\mathbb{D} = \mathbb{U}\mathbb{A}\mathbb{U}^*$. Rozepišme si $\mathbb{D}^* = (\mathbb{U}\mathbb{A}\mathbb{U}^*)^* = (\mathbb{A}\mathbb{U}^*)^*\mathbb{U}^* = \mathbb{U}\mathbb{A}^*\mathbb{U}^*$. Potom

$$\mathbb{D}\mathbb{D}^* = \mathbb{U}\mathbb{A}\mathbb{U}^*\mathbb{U}\mathbb{A}^*\mathbb{U}^* = \mathbb{U}\mathbb{A}\mathbb{A}^*\mathbb{U}^*,$$

$$\mathbb{D}^*\mathbb{D} = \mathbb{U}\mathbb{A}^*\mathbb{U}^*\mathbb{U}\mathbb{A}\mathbb{U}^* = \mathbb{U}\mathbb{A}^*\mathbb{A}\mathbb{U}^* \overset{1}{=} \mathbb{U}\mathbb{A}\mathbb{A}^*\mathbb{U}^*.$$

Vidíme, že je-li $\mathbb{A}$ normální (tuto vlastnost jsme potřebovali v rovnosti označené 1), je i $\mathbb{D}$ normální. Potom kombinací tvrzení Schurovy věty ($\mathbb{D}$ je trojúhelníková) a předchozího lemmatu ($\mathbb{D}$ je normální) dostáváme, že $\mathbb{D}$ je nutně diagonální. Nechť je nyní $\mathbb{A}$ navíc hermitovská, tzn. platí $\mathbb{A} = \mathbb{A}^*$. Potom je hermitovská i $\mathbb{D}$, jelikož platí

$$\mathbb{D}^* = \mathbb{U}\mathbb{A}^*\mathbb{U}^* = \mathbb{U}\mathbb{A}\mathbb{U}^* = \mathbb{D}.$$

Diagonální hermitovská matice musí mít na diagonále realná čísla (aby na ně nemělo vliv opruhování). □

Definice 1.64. Nechť $\mathbb{A} \in \mathbb{C}^{n,n}$. Čtvercovou matici $\mathbb{A}$ nazvu **pozitivně definitní**, pokud $\forall \vec{x} \in \mathbb{C}^n, \vec{x} \neq \vec{0}$ platí, že $\vec{x}^*\mathbb{A}\vec{x} \in \mathbb{R}^+$. Značíme $\mathbb{A} > 0$.

Poznámka 1.65. Je-li pro $\mathbb{A}, \mathbb{B} \in \mathbb{C}^{n,n}$ matice $\mathbb{A} - \mathbb{B} > 0$, píšeme $\mathbb{A} > \mathbb{B}$. Tento zápis nemluví o hodnotách prvků těchto matic.

Poznámka 1.66. V námi používané definici skalárního součinu platí, že $\vec{x}^*\mathbb{A}\vec{x} = (\mathbb{A}\vec{x}, \vec{x})$.

Poznámka 1.67. Tato definice pozitivní definitnosti je obecnější než v [2], neboť je platná pro všechny čtvercové matice a nikoliv jen pro symetrické matice.

Věta 1.68. Nechť $\mathbb{A} \in \mathbb{C}^{n,n}$ je pozitivně definitní matice. Potom její vlastní čísla jsou kladná. Je-li naopak $\mathbb{A} \in \mathbb{C}^{n,n}$ hermitovská matice s kladnými vlastními čísly, pak je $\mathbb{A}$ pozitivně definitní.

Důkaz 1. Předpokládáme, že $\mathbb{A}$ je pozitivně definitní, neboli že $\forall \vec{x} \in \mathbb{C}^n, \vec{x} \neq \vec{0}$ platí $\vec{x}^*\mathbb{A}\vec{x} > 0$. Beru si libovolně pevně $\lambda \in \sigma(\mathbb{A})$ a k němu $\vec{x}_\lambda$ příslušný vlaastní vektor. Ten si normujeme na 1.

$$0 < \vec{x}_\lambda^*\mathbb{A}\vec{x}_\lambda = (\mathbb{A}\vec{x}_\lambda, \vec{x}_\lambda) = (\lambda\vec{x}_\lambda, \vec{x}_\lambda) = \lambda(\vec{x}_\lambda, \vec{x}_\lambda) = \lambda||\vec{x}_\lambda||^2 = \lambda$$

□

Důkaz 2. Nyní předpokládáme, že $\mathbb{A}$ je hermitovská (a tedy podle 1.28 i normální) a má kladná vlastní čísla. Díky důsledku Schurovy věty umíme zapsat rozklad $\mathbb{A} = \mathbb{U}^*\mathbb{D}\mathbb{U}$, kde $\mathbb{U}$ je regulární unitární matice a $\mathbb{D}$ je

diagonální s vlastními čísli $\mathbb{A}$ na diagonále. Díky předpokladům tedy víme, že $\forall\, i \in \hat{n}$ platí $d_{ii} > 0$. Podívejme se jak bude vypadat $(\mathbb{A}\vec{x}, \vec{x})$ pro libovolné pevné $\vec{x} \in \mathbb{C}^n, \vec{x} \neq \vec{0}$.

$$(\mathbb{A}\vec{x}, \vec{x}) = (\mathbb{U}^*\mathbb{D}\mathbb{U}\vec{x}, \vec{x}) = (\mathbb{D}\mathbb{U}\vec{x}, \mathbb{U}\vec{x}) = (\mathbb{D}\vec{y}, \vec{y}) = \sum_{i=1}^{n} d_{ii}|y_i|^2 > 0$$

□

Poznámka 1.69. Nechť $\mathbb{A} \in \mathbb{C}^{n,n}$ pozitivně definitní. Potom pokud za libovolné $\vec{x} \in \mathbb{C}^n$ vezmu i-tý vektor standardní báze $\vec{e}_i$, dostaneme $0 < \vec{e}_i^*\mathbb{A}\vec{e} = (\mathbb{A}\vec{e}_i, \vec{e}_i) = (\mathbb{A}_{\bullet i}, \vec{e}_i) = a_{ii}$, tzn. všechny diagonální prvky pozitivně definitní matice jsou kladné.

Definice 1.70. Nechť $\mathbb{A} \in \mathbb{C}^{n,n}$ je hermitovská a pozitivně definitní. Pro libovolné $p \in \mathbb{R}$ definujeme $\mathbb{A}^p = \mathbb{U}^*\mathbb{D}^p\mathbb{U}$.

Poznámka 1.71. Z Schurovy věty 1.59 a jejího důsledku 1.63 víme, že pro hermitovskou (a tedy i normální) matici $\mathbb{A}$ rozklad $\mathbb{A} = \mathbb{U}^*\mathbb{D}\mathbb{U}$ existuje, kde $\mathbb{D}$ je diagonální matice. Z poznámky 1.61 pak také víme, že na diagonále $\mathbb{D}$ leží vlastní čísla matice $\mathbb{A}$.

Poznámka 1.72. Můžeme si povšimnout, že definice 1.70 odpovídá pozorování pro celočíselné mocniny:

$$\mathbb{A}^2 = \mathbb{U}^*\mathbb{D}\mathbb{U}\mathbb{U}^*\mathbb{D}\mathbb{U} = \mathbb{U}^*\mathbb{D}\mathbb{I}\mathbb{D}\mathbb{U} = \mathbb{U}^*\mathbb{D}^2\mathbb{U}.$$

1.3 Posloupnosti vektorů a matic a normy

1.3.1 Posloupnosti

Definice 1.73. Nechť je dána **posloupnost vektorů**

$$\vec{x}^{(k)} = \begin{pmatrix} x_1^{(k)} \\ \vdots \\ x_n^{(k)} \end{pmatrix} \quad \text{pro } k \in \mathbb{N}.$$

Řekneme, že posloupnost $\left(\vec{x}^{(k)}\right)_{k=1}^{+\infty}$ **konverguje** k vektoru

$$\vec{x} = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}$$

právě tehdy když $\forall\, i \in \hat{n}$ platí $\lim\limits_{k\to+\infty} x_i^{(k)} = x_i$. Tuto vlastnost značíme $\lim\limits_{k\to+\infty} \vec{x}^{(k)} = \vec{x}$ nebo $\vec{x}^{(k)} \to \vec{x}$.

Definice 1.74. Analogicky řekneme, že **posloupnost matic**

$$\mathbb{A}^{(k)} = \begin{pmatrix} a_{11}^{(k)} & \dots & a_{1n}^{(k)} \\ \vdots & \ddots & \vdots \\ a_{m1}^{(k)} & \dots & a_{mn}^{(k)} \end{pmatrix} \in \mathbb{C}^{m,n} \quad \text{pro } k \in \mathbb{N}$$

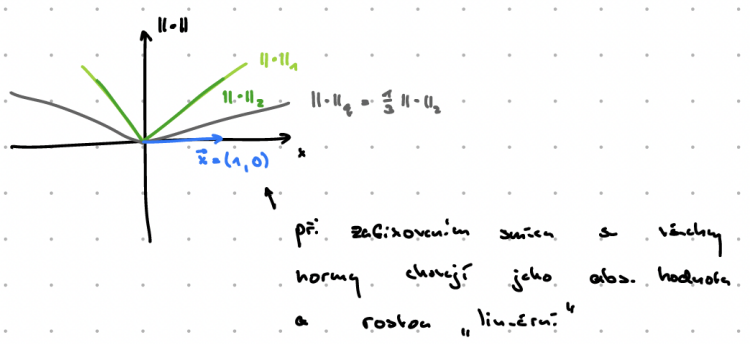
konverguje k matici $\mathbb{A}$ právě tehdy když $\forall\, i \in \hat{m}, j \in \hat{n}$ platí $\lim\limits_{k\to+\infty} a_{ij}^{(k)} = a_{ij}$.

Poznámka 1.75. Dokazování konvergence po prvcích je časově velmi náročně. K vyšetořování tedy budeme používat normy.

1.3.2 Normy

Definice 1.76. Zobrazení $||\cdot|| : \mathbb{C}^n \to \mathbb{R}$ nazvu **normou** na množině vektorů z $\mathbb{C}^n$, splňuje-li $\forall \vec{x}, \vec{y} \in \mathbb{C}^n$ a $\forall \alpha \in \mathbb{C}$ následující vlastnosti:

1. $||\vec{x}|| \geq 0 \wedge (||\vec{x}|| = 0 \iff \vec{x} = \vec{0})$,
2. $||\alpha\vec{x}|| = |\alpha|\,||\vec{x}||$,
3. $||\vec{x} + \vec{y}|| \leq ||\vec{x}|| + ||\vec{y}||$.



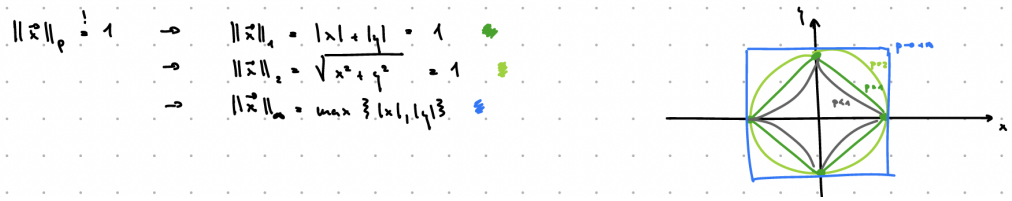
Obrázek 1.2: graf norem (TODO: vysvětlit)

Poznámka 1.77. Bude se nám často hodit fakt, že $||\vec{x} - \vec{y}|| = 0 \iff \vec{x} - \vec{y} = \vec{0} \iff \vec{x} = \vec{y}$.

Poznámka 1.78. Existuje nespočetně mnoho norem splňující tuto definici. Mezi některé známe patří:

- euklidovská norma : $||\vec{x}||_2 := \left(\sum_{i=1}^{n} |x_i|^2\right)^{\frac{1}{2}}$,
- součtová norma : $||\vec{x}||_1 = \sum_{i=1}^{n} |x_i|$,
- maximová norma : $||\vec{x}||_\infty = \max_{i \in \hat{n}} |x_i|$.

Lze ukázat, že všechny tyto normy jsou jen speciálním případem obecně definované normy $||\vec{x}||_p = \left(\sum_{i=1}^{n} |x_i|^p\right)^{\frac{1}{p}}$.



Obrázek 1.3: geometrická představa norem (TODO: vysvětlit)

Věta 1.79. Pro libovolné dvě normy $||\cdot||_\alpha, ||\cdot||_\beta : \mathbb{C}^n \to \mathbb{R}$ existují kladné konstanty γ_1, γ_2 splňující $\forall \vec{x} \in \mathbb{C}^n$

$$\gamma_1 ||\vec{x}||_\alpha \leq ||\vec{x}||_\beta \leq \gamma ||\vec{x}||_\alpha.$$

Důkaz. Nezkouší se. (TODO: doplnit odkaz na skripta ANA3 doc. Štampacha) □

Důsledek 1.80. Je-li pro nějaké $\vec{x} \in \mathbb{C}^n$ $||\vec{x}||_\alpha = 0$, potom nutně $||\vec{x}||_\beta = 0$.

Věta 1.81. Nechť je dána posloupnost vektorů $\left(\vec{x}^{(k)}\right)_{k=1}^{+\infty}$ kde $\vec{x}^{(k)} \in \mathbb{C}^n$ pro každé $k \in \mathbb{N}$, vektor $\vec{x} \in \mathbb{C}^n$ a libovolná norma $||\cdot||_\alpha : \mathbb{C}^n \to \mathbb{R}$. Potom $\vec{x}^{(k)} \to \vec{x}$ právě tehdy když $||\vec{x}^{(k)} - \vec{x}||_\alpha \to 0$.

Důkaz. Než se pustíme do důkazu jednotlivých stran ekvivalence, pojďme si obecně říct, co znamená, že $\vec{x}^{(k)} \to \vec{x}$ v řeči norem.

$$\vec{x}^{(k)} \to \vec{x} \iff x_i^{(k)} \to x_i \overset{1}{\iff} |x_i^{(k)} - x_i| \to 0 \overset{2}{\iff} ||\vec{x}^{(k)} - \vec{x}||_\infty \to 0$$

Ekvivalence označená 1 je fakt plynoucí z definice limity číselné posloupnosti. Proč platí ekvivalence označená 2 je vidět z definice maximové normy.

$$||\vec{x}^{(k)} - \vec{x}||_\infty \to 0 \iff \max_{i \in \hat{n}} |x_i^{(k)} - x_i| \to 0$$

Protože musejí k nule konvergovat všechny prvky vektoru $\vec{x}$, musí konvergovat i ten, kde je rozdíl $|x_i^{(k)} - x_i|$ maximální. Celkem jsme tedy dostali vztah, že $\vec{x}^{(k)} \to \vec{x} \iff ||\vec{x}^{(k)} - \vec{x}||_\infty \to 0$. Nyní si do věty 1.79 vezmeme libovolnou normu $||\cdot||_\alpha$ a normu $||\cdot||_\infty$. Potom víme, že existují konstanty $\gamma_1, \gamma_2 \in \mathbb{R}^+$ takové, že

$$\gamma_1 ||\vec{x}^{(k)} - \vec{x}||_\alpha \leq ||\vec{x}^{(k)} - \vec{x}||_\infty \leq \gamma_2 ||\vec{x}^{(k)} - \vec{x}||_\alpha$$

Nyní už se můžeme pustit do důkazu jednotlivých implikací. V prvním případě (důkaz $\Rightarrow$) předpokládáme, že $\vec{x}^{(k)} \to \vec{x}$. To jsme řekli, že je ekvivalentní s výrokem $||\vec{x}^{(k)} - \vec{x}||_\infty \to 0$. Z nerovnosti $||\vec{x}^{(k)} - \vec{x}||_\infty \leq \gamma_2 ||\vec{x}^{(k)} - \vec{x}||_\alpha$ pak už dostáváme, že pro libovolnou normu $||\cdot||_\alpha$ platí $||\vec{x}^{(k)} - \vec{x}||_\alpha$. V druhém případě (důkaz $\Leftarrow$) předpokládáme, že $||\vec{x}^{(k)} - \vec{x}||_\alpha \to 0$. Z nerovnosti $\gamma_1 ||\vec{x}^{(k)} - \vec{x}||_\alpha \leq ||\vec{x}^{(k)} - \vec{x}||_\infty$ dostáváme, že za takového předpokladu i $||\vec{x}^{(k)} - \vec{x}||_\infty \to 0$, což je ekvivalentní s výrokem $\vec{x}^{(k)} \to \vec{x}$. □

Definice 1.82. Zobrazení $||\cdot|| : \mathbb{C}^n \to \mathbb{R}$ nazvu **normou** na množině čtvercových matic řádu n, splňuje-li $\forall\, \mathbb{A}, \mathbb{B} \in \mathbb{C}^{n,n}$ a $\forall\, \alpha \in \mathbb{C}$ následující vlastnosti:

1. $||\mathbb{A}|| \geq 0 \wedge (||\mathbb{A}|| = 0 \iff \mathbb{A} = \mathbb{O})$,
2. $||\alpha \mathbb{A}|| = |\alpha|\, ||\mathbb{A}||$,
3. $||\mathbb{A} + \mathbb{B}|| \leq ||\mathbb{A}|| + ||\mathbb{B}||$,
4. $||\mathbb{A}\mathbb{B}|| \leq ||\mathbb{A}||\, ||\mathbb{B}||$.

Poznámka 1.83. Na maticovou normu se lze dívat jako na vektorovou normu aplikovanou na vektory z $\mathbb{C}^{n^2}$. (TODO: doplnit vysvětlení, proč maticová norma splňuje větu o konvergenci vektorů v normě)

Poznámka 1.84. Příkladem užitečné maticové normy je Schurova (v anglických textech ozn. jako Frobeniova) norma, definovaná $||\mathbb{A}||_S = \left(\sum_{i,j=1}^n |a_{ij}|^2 \right)^{\frac{1}{2}}$.

Důkaz. TODO: důkaz splnění definice □

Definice 1.85. Nechť $\mathbb{A} \in \mathbb{C}^{n,n}, \vec{x} \in \mathbb{C}^n$. Maticovou normu nazvu **souhlasnou** s vektorovou normou právě tehdy když platí

$$||\mathbb{A}\vec{x}|| \leq ||\mathbb{A}||\, ||\vec{x}||.$$

Poznámka 1.86. Schurova maticová norma je souhlasná s euklidovskou normou.

Důkaz. TODO: doplnit důkaz □

Definice 1.87. Nechť $\mathbb{A} \in \mathbb{C}^{n,n}, \vec{x} \in \mathbb{C}^n$. Maticová norma **indukovaná** vektorovou normou je dána vztahem

$$||\mathbb{A}|| = \max_{||\vec{x}||=1} ||\mathbb{A}\vec{x}||.$$

Korektnost. Je třeba ověřit, zda takto definovaná maticová norma splňuje definici normy. TODO: doplnit □

Poznámka 1.88. Maximum existuje vždy, neboť $\{\vec{x} \in \mathbb{C}^n | ||\vec{x}|| = 1\}$ je kompaktní množina a norma je na ní spojitá. (Obdoba funkce spojité na uzavřeném intervalu — ta na něm maximum vždy nabývá.) Obráceně toto ale neplatí – ne pro každou maticovou normu existuje vektorová norma, která ji indukuje.

Věta 1.89 (ekvivalentní definice maticové normy)**.** Nechť $\mathbb{A} \in \mathbb{C}^{n,n}, \vec{x} \in \mathbb{C}^n$. Potom

$$||\mathbb{A}|| = \max_{||\vec{x}||=1} ||\mathbb{A}\vec{x}|| \iff ||\mathbb{A}|| = \max \frac{||\mathbb{A}\vec{x}||}{||\vec{x}||}$$

Důkaz. Výrok je zřejmý, místo toho abych dělal maximum jen přes $||\vec{x}|| = 1$ naopak normuji $||\mathbb{A}\vec{x}||$ na 1. □

Poznámka 1.90. Indukovaná maticová norma je souhlasná s vektorovou normou, která ji indukuje.

Důkaz. Chceme, aby platil výraz $||\mathbb{A}\vec{x}|| \leq ||\mathbb{A}||\,||\vec{x}||$, přičemž za $||\mathbb{A}||$ „dosazuji“ $\max_{||\vec{x}||=1} ||\mathbb{A}\vec{x}||$. Pro směr $\vec{x}$ kde je maximum nabýváno tak zřejmě bude platit rovnost, jinak bude $||\mathbb{A}\vec{x}||$ menší než své maximum. □

Věta 1.91. Nechť $\mathbb{A} \in \mathbb{C}^{n,n}, \vec{x} \in \mathbb{C}^n$. Potom při značení

(i) $||\mathbb{A}||_\infty = \max_{||\vec{x}||_\infty=1} ||\mathbb{A}\vec{x}||_\infty$,

(ii) $||\mathbb{A}||_1 = \max_{||\vec{x}||_1=1} ||\mathbb{A}\vec{x}||_1$,

(iii) $||\mathbb{A}||_2 = \max_{||\vec{x}||_2=1} ||\mathbb{A}\vec{x}||_2$.

platí následující vztahy:

1. $||\mathbb{A}||_\infty = \max_{i\in\hat{n}} \sum_{j=1}^{n} |a_{ij}|$ *(řádková norma – odpovídá maximu součtu přes jednotlivé řádky)*,
2. $||\mathbb{A}||_1 = \max_{j\in\hat{n}} \sum_{i=1}^{n} |a_{ij}|$ *(sloupcová norma – odpovídá maximu součtu přes jednotlivé sloupce)*,
3. $||\mathbb{A}||_2 = \sqrt{\rho(\mathbb{A}^*\mathbb{A})}$.

Důkaz 1. Chceme najít alternativní způsob výpočtu výrazu $||\mathbb{A}||_\infty = \max_{||\vec{x}||_\infty=1} ||\mathbb{A}\vec{x}||_\infty$. Z definice maximové normy si můžeme rozepsat $||\vec{x}||_\infty = 1$. To platí, jsou-li všechna $|x_i| \leq 1$ kde $i \in \hat{n}$. Pro jednoduchost zatím předpokládejme, že $\mathbb{A} \in \mathbb{R}^{n,n}$ a $\vec{x} \in \mathbb{R}^n$. Potom $\forall\, i \in \hat{n}$ platí $x_i \in [-1, 1]$. Označme si $\vec{y} \coloneqq \mathbb{A}\vec{x}$. Tento výraz si můžeme představit tak, že jednotlivými složkami $\vec{x}$ násobíme sloupce matice $\mathbb{A}$, tj.

$$\vec{y} = x_1\mathbb{A}_{\cdot 1} + x_2\mathbb{A}_{\cdot 2} + ... + x_2\mathbb{A}_{\cdot 2} = x_1 \begin{pmatrix} a_{11} \\ \vdots \\ a_{n1} \end{pmatrix} + x_2 \begin{pmatrix} a_{12} \\ \vdots \\ a_{n2} \end{pmatrix} + ... + x_n \begin{pmatrix} a_{1n} \\ \vdots \\ a_{nn} \end{pmatrix}$$

My hledáme co největší $||\vec{y}||_\infty$. To v kontextu maximové normy znamená, že koeficienty $x_1, ..., x_n$ chceme nastavit tak, aby libovolná složka vektoru $\vec{y}$ byla co největší. Dívám-li se tedy na i-tý prvek vektoru $\vec{y}$, zajímá mě v tu chvíli jen rovnice

$$y_i = x_1a_{i1} + ...x_na_{in} = \sum_{j=1}^{n} x_ja_{ij}.$$

Je zřejmé, že aby y_i bylo maximální, budu brát

$$x_j \coloneqq \left\{ \begin{matrix} 1 \text{ pro } a_{ij} > 0 \\ -1 \text{ pro } a_{ij} < 0 \end{matrix} \right\} = \operatorname{sgn} a_{ij}$$

Dosadím-li toto nové vyjádření do mého výrazu pro y_i, dostáváme

$$y_i = \sum_{j=1}^{n} x_j^{(i)} a_{ij} = \sum_{j=1}^{n} \operatorname{sgn}(a_{ij}) \cdot a_{ij} = \sum_{j=1}^{n} |a_{ij}|.$$

(Horní index (i) u x_j je doplněn kvůli tomu, že daná volba hodnot x_j tak, aby posléze platilo $\operatorname{sgn}(a_{ij}) \cdot a_{ij} = |a_{ij}|$ je platná jen pro konkrétní zkoumaný i-tý řádek.) Chceme maximální hodnotu $||\mathbb{A}\vec{x}||_\infty$, tedy maximální hodnotu $||\vec{y}||_\infty$. Tu podle toho, co jsme si řekli snadno spočítám tím, že projdu všechny řádky a určím, kdy byla hodnota y_i největší, což je znění věty.

$$||\vec{y}||_\infty = \max_{i \in \hat{n}} |y_i| = \max_{i \in \hat{n}} \sum_{j=1}^{n} |a_{ij}|$$

Na začátku jsme se omezili jen na reálná čísla. Upravme tedy nyní naši volbu $x_j^{(i)}$ tak, aby stále platilo $\operatorname{sgn}(a_{ij}) \cdot a_{ij} = |a_{ij}|$ i nad komplexními čísli. Z vlastností komlexních čísel je zřejmé, že taková volba bude mít tvar

$$x_j^{(i)} := \frac{\overline{a_{ij}}}{|a_{ij}|}.$$

□

Důkaz 2. Postup bude v tomto případě analogický. Chceme najít alternativní způsob výpočtu výrazu $||\mathbb{A}||_1 = \max_{||\vec{x}||_1 = 1} ||\mathbb{A}\vec{x}||_1$. Z definice jednotkové normy si můžeme rozepsat $||\vec{x}||_1 = 1$. To platí, je-li $\sum_{j=1}^{n} |x_j| = 1$. Příkladem vektorů splňujících tuto vlastnost jsou

$$\vec{v} = \begin{pmatrix} \frac{1}{n} \\ \vdots \\ \frac{1}{n} \end{pmatrix}, \quad \text{nebo} \quad \vec{w} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}.$$

Opět si označím $\vec{y} := \mathbb{A}\vec{x}$ a opět si ho představím tak, že jednotlivými složkami $\vec{x}$ násobíme sloupce matice $\mathbb{A}$, tj.

$$\vec{y} = x_1 \mathbb{A}_{\cdot 1} + x_2 \mathbb{A}_{\cdot 2} + \ldots + x_2 \mathbb{A}_{\cdot 2} = x_1 \begin{pmatrix} a_{11} \\ \vdots \\ a_{n1} \end{pmatrix} + x_2 \begin{pmatrix} a_{12} \\ \vdots \\ a_{n2} \end{pmatrix} + \ldots + x_n \begin{pmatrix} a_{1n} \\ \vdots \\ a_{nn} \end{pmatrix}$$

Tentokrát se snažíme maximalizovat součet složek vektoru $\vec{y}$. Je třeba si rozmyslet, že toho dosáhnu tím, že najdu sloupec matice $\mathbb{A}$ s největším součtem složek, u něj nastavím x_j na 1, a zbytku dám nulu. K tomu můžeme dojít například představou, že jednotlivé sloupcové vektory představují bankovní účty lidí, přičemž každá složka je účet v jiné měně (ale pro potřeby sčítání převedená na koruny). Já chci ukrást co nejvíc peněz, ale smím ukrást jen tolik, aby podíl mého kradení byl roven jedné. To znamená, že buď můžu jednomu člověku ukrást vše, nebo dvěma polovinu, nebo např. deseti lidem desetinu jejich majetku. Zde už je lépe vidět fakt, že nejvíce peněz ukradu tím, že si najdu nejbohatšího člověka, a tomu ukradnu vše. Matematicky tuto úvahu formulujeme tím, že volíme $x_j = 1$ pro $j \in \hat{n}$ kde je výraz $\sum_{i=1}^{n} |a_{ij}|$ maximální, a $x_j = 0$ jinak. Potom vektor $\vec{y}$ odpovídá tomuto sloupci matice $\mathbb{A}$. $||\vec{y}||_1$ tedy bude maximální, je-li maximální $\sum_{i=1}^{n} |y_i|$, což odpovídá $\max_{j \in \hat{n}} \sum_{i=1}^{n} |a_{ij}|$. □

Důkaz 3. Tentokrát chceme najít alternativní vyjádření pro výraz $||\mathbb{A}||_2 = \max_{||\vec{x}||_2 = 1} ||\mathbb{A}\vec{x}||_2$. Zaměříme se na to, jak vypadá dvojková norma $\mathbb{A}\vec{x}$. Budeme pracovat s kvadráty, abychom se nemuseli zaobírat odmocninami. Kvadráty nám zároveň nezmění maximum.

$$||\mathbb{A}\vec{x}||_2^2 = (\mathbb{A}\vec{x}, \mathbb{A}\vec{x}) = (\vec{x}, \mathbb{A}^*\mathbb{A}\vec{x})$$

Matice $\mathbb{A}^*\mathbb{A}$ je hermitovská, protože platí $(\mathbb{A}^*\mathbb{A})^* = \mathbb{A}^*(\mathbb{A}^*)^* = \mathbb{A}^*\mathbb{A}$. Podle poznámky 1.28 je tedy i normální,

proto můžeme aplikovat Shurovu větu, respektive její důsledek 1.63, který nám říká, že umíme vytvořit rozklad

$$\mathbb{A}^*\mathbb{A} = \mathbb{U}^*\mathbb{D}\mathbb{U},$$

kde $\mathbb{D}$ je diagonální matice se spektrem $\mathbb{A}^*\mathbb{A}$ na diagonále. Pokračujme dále ve výrazu pro kvadrát dvojkové normy $\mathbb{A}\vec{x}$.

$$||\mathbb{A}\vec{x}||_2^2 = (\vec{x}, \mathbb{A}^*\mathbb{A}\vec{x}) = (\vec{x}, \mathbb{U}^*\mathbb{D}\mathbb{U}\vec{x}) = (\mathbb{U}\vec{x}, \mathbb{D}\mathbb{U}\vec{x}) = (\vec{y}, \mathbb{D}\vec{y}) \quad \text{kde } \vec{y} \coloneqq \mathbb{U}\vec{x}$$

Protože $\mathbb{U}$ je dle použitého důsledku Shurovy věty 1.63 regulární, vztah mezi $\vec{y}$ a $\vec{x}$ je jednoznačný. Z toho, že $\mathbb{U}$ je unitární, dále dle 1.54 víme, že zachovává normu. Při hledání maxima volíme $||\vec{x}||_2 = 1$, proto

$$||\vec{y}||_2 = ||\mathbb{U}\vec{x}||_2 = ||\vec{x}||_2 = 1.$$

Nyní se vraťme k výrazu, který chceme upravit a dosaďme do něj to, co jsme zatím získali. Jak již bylo zmíněno, kvadráty nám nezmění hodnotu maxima, proto s nimi můžeme nadále pracovat.

$$||\mathbb{A}||_2^2 = \max_{||\vec{x}||_2=1} ||\mathbb{A}\vec{x}||_2^2 = \max_{||\vec{y}||_2=1} (\vec{y}, \mathbb{D}\vec{y}) = \max_{||\vec{y}||_2=1} \sum_{i=1}^{n} d_{ii}|y_i|^2$$

Poznamenejme, že při výpočtu maxima jsme z procházení $||\vec{x}||_2 = 1$ mohli přejít k procházení $||\vec{y}||_2 = 1$ jen díky tomu, že jejich přiřazení je jednoznačné a je zachována norma. Výraz $||\vec{y}||_2^2 = 1$ dále z definice dvojkové normy odpovídá výrazu $\sum_{i=1}^{n} |y_i|^2$. Abych tedy maximalizoval $||\mathbb{A}||_2^2$ volím stejným argumentem jako v důkazu bodu 2 $y_i = 1$ pro maximální d_{ii} a $y_i = 0$ jinak. Maximalizaci $||\mathbb{A}||_2^2$ jsme tedy převedli na hledání maximálního d_{ii} Všiměme si, že $\mathbb{A}^*\mathbb{A}$ je pozitivně definitní matice, protože platí $(\vec{x}, \mathbb{A}^*\mathbb{A}\vec{x}) = (\mathbb{A}\vec{x}, \mathbb{A}\vec{x}) > 0$ pro každé $\vec{x} \neq \vec{0}$. Z věty 1.68 víme, že pozitivně definitní matice má kladná vlastní čísla. Již dříve jsme ukázali, že d_{ii} (diagonální prvky matice $\mathbb{D}$) jsou vlastní čísla $\mathbb{A}^*\mathbb{A}$, proto $d_{ii} > 0$. Hledáme maximální d_{ii}, která jsou všechna kladná, tudíž

$$\max_{i\in\hat{n}} d_{ii} = \max_{i\in\hat{n}} |d_{ii}| = \max_{\lambda\in\sigma(\mathbb{A}^*\mathbb{A})} |\lambda| = \rho(\mathbb{A}^*\mathbb{A}),$$

což je pod odmocninou znění věty. □

Definice 1.92. Nechť $\mathbb{A} \in \mathbb{C}^{n,n}$ je hermitovská a pozitivně definitní matice. Energenickou vektorovou a maticovou normu definujeme $\forall\, \vec{x} \in \mathbb{C}^n$ a $\forall\, \mathbb{B} \in \mathbb{C}^{n,n}$ vztahy

$$||\vec{x}||_A \coloneqq ||\mathbb{A}^{\frac{1}{2}}\vec{x}||_2,$$
$$||\mathbb{B}||_A \coloneqq ||\mathbb{A}^{\frac{1}{2}}\mathbb{B}\mathbb{A}^{-\frac{1}{2}}||_2.$$

1.3.3 Geometrické posloupnosti

Věta 1.93. Nechť $\mathbb{A} \in \mathbb{C}^{n,n}$. Potom platí $\mathbb{A}^k \to \mathbb{O} \iff \rho(\mathbb{A}) < 1$.

Důkaz. Nechť nejdříve $\mathbb{A}$ diagonalizovatelná, tzn. existují $\mathbb{D}$ diagonální a $\mathbb{X}$ regulární takové, že $\mathbb{A} = \mathbb{X}\mathbb{D}\mathbb{X}^{-1}$. Jak jsme již dříve ukázali, pro $k \in \mathbb{N}$ platí $\mathbb{A}^k = \mathbb{X}\mathbb{D}^k\mathbb{X}^{-1}$. (To lze ukázat např. pro $k = 2$, kdy $\mathbb{A}^2 = \mathbb{X}\mathbb{D}\mathbb{X}^{-1}\mathbb{X}\mathbb{D}\mathbb{X}^{-1} = \mathbb{X}\mathbb{D}\mathbb{I}\mathbb{D}\mathbb{X}^{-1} = \mathbb{X}\mathbb{D}^2\mathbb{X}^{-1}$, a indukcí pro libovolné k.) Pak zřejmě $\mathbb{A}^k \to \mathbb{O} \iff \mathbb{D}^k \to \mathbb{O}$. Z příslušných definic plyne, že

$$\mathbb{D}^k = \begin{pmatrix} \lambda_1^k & 0 & \cdots & 0 \\ 0 & \lambda_2^k & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_{nn}^k \end{pmatrix} \to \mathbb{O} \iff (\forall\, i \in \hat{n})(\lambda_i^k \to 0) \iff (\forall\, i \in \hat{n})(|\lambda_i| < 1) \iff \rho(\mathbb{A}) < 1.$$

kde λ_i je i-té vlastní číslo matice $\mathbb{A}$. Nechť nyní $\mathbb{A}$ není diagonalizovatelná. Použijeme Jordanovu větu 1.47, která říká, že každá matice $\mathbb{A}$ je podobná matici $\mathbb{J}$ v Jordannově kanonickém tvaru, tzn. existuje regulární

matice $\mathbb{X}$ taková, že $\mathbb{A} = \mathbb{X}^{-1}\mathbb{J}\mathbb{X}$. Potom opět analogickým arguemntem jako výše $\mathbb{A}^k = \mathbb{X}^{-1}\mathbb{J}^k\mathbb{X}$ a tedy $A^k \to \mathbb{O} \iff \mathbb{J}^k \to \mathbb{O}$. Matice $\mathbb{J}$ má blokově diagonální tvar, tzn. je tvořena Jordanovými bloky $\mathbb{J}_i$. Potom

$$\mathbb{J}^k = \begin{pmatrix} \mathbb{J}_1^k & 0 & \cdots & 0 \\ 0 & \mathbb{J}_2^k & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbb{J}_m^k \end{pmatrix} \to \mathbb{O} \iff (\forall\, i \in \hat{m})(\mathbb{J}_i^k \to \mathbb{O})$$

□

Každý blok $\mathbb{J}_i$ má na diagonále i-té vlastní číslo λ_i a na nad diagonálou samé jedničky. Tedy

$$\mathbb{J}_i = \begin{pmatrix} \lambda_i & 1 & 0 & \cdots & 0 \\ 0 & \lambda_i & 1 & \cdots & 0 \\ 0 & 0 & \lambda_i & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \lambda_i \end{pmatrix}$$

Pro zjednodušení zápisu označme $\mathbb{T} \coloneqq \mathbb{J}_i$, $t \coloneqq \lambda_i$. Dokážeme nyní, že pro ij-tý prvek matice $\mathbb{T}^k$ platí

$$\mathbb{T}^k = \begin{cases} \binom{k}{j-i} t^{k-(j-i)} & \text{pro } j > i \text{ a } k \geq j-i, \\ t^k & \text{pro } j = i \\ 0 & \text{jinak.} \end{cases}$$

Postupujme indukcí, nechť $k = 1$. Potom tvrzení přejde na

$$\mathbb{T}^1 = \begin{cases} \binom{1}{j-i} t^{1-(j-i)} & \text{pro } j > i \text{ a } 1 \geq j-i, \\ t^1 & \text{pro } j = i, \\ 0 & \text{jinak,} \end{cases}$$

ale podmínky $j > i$ a $1 \geq j - i$ jsou zároveň splněny jen pokud $j = i + 1$ a $\binom{1}{j-i}t^0 = \binom{1}{1} = 1$. Dostáváme tedy opravdu správný Jordanův kanonický tvar

$$\mathbb{T}^1 = \begin{pmatrix} t & 1 & 0 & \cdots & 0 \\ 0 & t & 1 & \cdots & 0 \\ 0 & 0 & t & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & t \end{pmatrix}.$$

Nyní předpokládejme, že tvrzení platí pro $k-1$ a ukážeme, že platí i pro k. Z definice maticového násobení platí

$$(\mathbb{T}^k)_{ij} = \sum_{l=1}^{n} (\mathbb{T}^{k-1})_{il} \cdot \mathbb{T}_{lj} = (\mathbb{T}^{k-1})_{ij}\lambda + (\mathbb{T}^{k-1})_{i,j-1} \cdot 1$$

kde jsme využili ověřených vlastností pro $k = 1$, tedy že pro $l = j$ je $\mathbb{T}_{lj} = t$ a pro $l = j-1$ je $\mathbb{T}_{lj} = 1$. Dosadíme do vztahu z indukčního předpokladu:

$$(\mathbb{T}^k)_{ij} = \binom{k-1}{j-i} t^{k-1-(j-i)} \cdot t + \binom{k-1}{j-1-i} t^{k-1-(j-1-i)} \cdot 1 =$$

$$= t^{k-(j-i)} \left(\binom{k-1}{j-i} + \binom{k-1}{j-1-i} \right) = \binom{k}{j-i} t^{k-(j-i)}.$$

Máme ověřenou platnost tvrzení pro k, zbývá se tedy vrátit k původní větě a ukázat, že libovolné $(\mathbb{T}^k)_{ij}$ konverguje k nule, pokud $|t| < 1$. Podívejme se na obecné kombinační číslo:

$$\binom{n}{m} = \frac{n!}{m!(n-m)!} = \frac{n(n-1)(n-2)\cdots(n-m+1)}{m!} \leq \frac{n^m}{m!},$$

kde jsme využili toho, že v čitateli je m činitelů, které jsou všechny menší nebo rovny n. Nyní tedy můžeme zapsat

$$\binom{k}{j-i} t^{k-(j-i)} \leq \frac{k^{j-i}}{(j-i)!} t^{k-(j-i)}.$$

Zde $c \coloneqq j - i$ je konstanta (díváme se na konkrétní prvek matice $\mathbb{T}^k$), takže pro $|t| < 1$ a s využitím toho, že t^k konverguje rychleji než t^c pak už zřejmě dostaneme hledanou konvergenci k nule.

Poznámka 1.94. Všiměme si, že věta je realnou analogií vztahu $a^k \to 0 \iff |a| < 1$ kde $a \in \mathbb{R}$.

Věta 1.95. Nechť $\mathbb{A} \in \mathbb{C}^{n,n}$. Existuje-li norma $||\cdot||_\alpha$ taková, že $||\mathbb{A}||_\alpha < 1$, potom $\mathbb{A}^k \to \mathbb{O}$.

Důkaz. Čtvrtá vlastnost maticové normy říká, že pro každé $\mathbb{A}, \mathbb{B} \in \mathbb{C}^{n,n}$ je $||\mathbb{AB}|| \leq ||\mathbb{A}||||\mathbb{B}||$. Odtud jednoduše indukcí pro každé $k \in \mathbb{N}$ platí $||A^k|| \leq ||A||^k$ a tedy $||\mathbb{A}^k|| \to 0 \iff ||A|| < 1$. □

Důsledek 1.96. Pro libovolnou matici $\mathbb{A} \in \mathbb{C}^{n,n}$ a normu $||\cdot||_\alpha$ platí, že $\rho(\mathbb{A}) \leq ||\mathbb{A}||_\alpha$.

Důkaz. Nechť $\mathbb{A} \in \mathbb{C}^{n,n}$ a $\epsilon > 0$. Definujeme

$$\mathbb{B} = \frac{1}{||\mathbb{A}|| + \epsilon}\mathbb{A},$$

odkud zřejmě $||\mathbb{B}|| < 1$. Potom z předchozích tvrzení $||\mathbb{B}||^k \to 0 \iff \rho(\mathbb{B}) < 1$. Vezměme si libovolné vlastní číslo $\lambda \in \sigma(\mathbb{A})$ a příslušný vlastní vektor $\vec{x}$, tedy $\mathbb{A}\vec{x} = \lambda\vec{x}$. Pak také

$$\mathbb{B}\vec{x} = \frac{1}{||\mathbb{A}|| + \epsilon}\mathbb{A}\vec{x} = \frac{\lambda}{||\mathbb{A}|| + \epsilon}\vec{x}$$

Proto $\frac{\lambda}{||\mathbb{A}||+\epsilon} \in \sigma(\mathbb{B})$, tzn. je vlastní číslo matice $\mathbb{B}$. My ale už víme, že $\rho(\mathbb{B}) < 1$, tedy

$$\frac{\lambda}{||\mathbb{A}|| + \epsilon} < 1 \Rightarrow \lambda < ||\mathbb{A}|| + \epsilon.$$

Protože jsme $\epsilon > 0$ a $\lambda \in \sigma(\mathbb{A})$ brali libovolně, dostáváme tvrzení věty. □

Věta 1.97. Nechť $\mathbb{A} \in \mathbb{C}^{n,n}$. Řada $\sum_{k=1}^{\infty} \mathbb{A}^k = \mathbb{I} + \mathbb{A} + \mathbb{A}^2 + ...$ konverguje k $(\mathbb{I} - \mathbb{A})^{-1}$ právě tehdy, když $\rho(\mathbb{A}) < 1$.

Poznámka 1.98. Opět jde o analogii realného tvrzení, kdy $\sum_{k=0}^{\infty} a^k = \frac{1}{1-a}$ právě tehdy, když $|a| < 1$.

Důkaz. Zabývajme se nejdříve samotnou existencí matice $(\mathbb{I} - \mathbb{A})^{-1}$. Z Schurovy věty 1.59 víme, že umíme napsat $\mathbb{A} = \mathbb{U}^*\mathbb{R}\mathbb{U}$, kde $\mathbb{U}$ je unitární a $\mathbb{R}$ je horní trojúhelníková s vlastními čísli matice $\mathbb{A}$ na diagonále. Pak opět z Schurovy věty

$$\mathbb{I} - \mathbb{A} = \mathbb{I} - \mathbb{U}^*\mathbb{R}\mathbb{U} = \mathbb{U}^*\mathbb{I}\mathbb{U} - \mathbb{U}^*\mathbb{R}\mathbb{U} = \mathbb{U}^*(\mathbb{I} - \mathbb{R})\mathbb{U}.$$

Předpoklad $\rho(\mathbb{A}) < 1$ nám říká, že $\forall\, \lambda \in \sigma(\mathbb{A})$ je $|\lambda| < 1$. Protože měla $\mathbb{R}$ vlastní čísla na diagonále, jsou diagonální prvky $\mathbb{I} - \mathbb{R}$ nenulové. Determinant trojúhelníkové matice je součin jejích diagonálních prvků, tedy je v tomto případě nenulový a matice $\mathbb{I} - \mathbb{R}$ je regulární. Protože $\mathbb{U}$ je unitární, je i $\mathbb{I} - \mathbb{A}$ regulární, a tedy existuje matice $(\mathbb{I} - \mathbb{A})^{-1}$. Označme si částečný součet $S_k = \mathbb{I} + \mathbb{A} + \mathbb{A}^2 + ...\mathbb{A}^k$. Pak

$$\mathbb{S}_k(\mathbb{I} - \mathbb{A}) = \mathbb{I} + \mathbb{A} + \mathbb{A}^2 + ... + \mathbb{A}^k - \mathbb{A} - \mathbb{A}^2 - ... - \mathbb{A}^{k+1} = \mathbb{I} - \mathbb{A}^{k+1}.$$

Když obě strany rovnosti vynásobíme maticí $(\mathbb{I}-\mathbb{A})^{-1}$, dostáváme

$$\mathbb{S}_k = (\mathbb{I}-\mathbb{A})^{-1}(\mathbb{I}-\mathbb{A}^{k+1}) = (\mathbb{I}-\mathbb{A})^{-1} - (\mathbb{I}-\mathbb{A})^{-1}\mathbb{A}^{k+1}.$$

Pro $k \to \infty$ tento výraz nyní konverguje k $(\mathbb{I}-\mathbb{A})^{-1}$, pokud $\mathbb{A}^{k+1} \to 0$, což z předchozí věty 1.93 platí právě tehdy, když $\rho(\mathbb{A}) < 1$. □

Věta 1.99. Nechť $\mathbb{A} \in \mathbb{C}^{n,n}$ taková, že $||\mathbb{A}|| < 1$. Potom

$$||(\mathbb{I}-\mathbb{A})^{-1} - \sum_{j=0}^{k} \mathbb{A}^j|| \leq \frac{||\mathbb{A}||^{k+1}}{1-||\mathbb{A}||}.$$

Poznámka 1.100. Z věty 1.97 víme, že $\sum_{k=0}^{+infty}$ konverguje za daných předpokladů k $(\mathbb{I}-\mathbb{A})^{-1}$. Všiměme si, že nynější věta nám tedy dává horní odhad pro chybu aproximace nekonečné řady $\sum_{k=0}^{\infty} \mathbb{A}^k$ jen konečnou po nějaké $k \in \mathbb{N}$.

Důkaz. Nechť $\mathbb{A} \in \mathbb{C}^{n,n}$ a podívejme se na výraz v normě.

$$||(\mathbb{I}-\mathbb{A})^{-1} - \sum_{j=0}^{k} \mathbb{A}^j|| = ||\sum_{j=0}^{+\infty} \mathbb{A}^j - \sum_{j=0}^{k} \mathbb{A}^j|| = ||\sum_{j=k+1}^{+\infty} \mathbb{A}^j|| = ||\mathbb{A}^{k+1}\sum_{j=0}^{+\infty} \mathbb{A}^j||$$

$$\leq ||\mathbb{A}^{k+1}|| \cdot ||\sum_{j=0}^{+\infty} \mathbb{A}^j|| \leq ||\mathbb{A}^{k+1}|| \sum_{j=0}^{+\infty} ||\mathbb{A}^j|| = \frac{||\mathbb{A}^{k+1}||}{1-||\mathbb{A}||}.$$

□

1.4 Otázky

- Pozitivně definitní matice a jejich vlastnosti.
- Konvergence geometrických maticových posloupností.
- Generované maticové normy a jejich alternativní vyjádření.

Kapitola 2

Úvod do numerických výpočtů

Tato kapitola se věnuje základním principům, na kterých fungují numerické výpočty. Klíčové je pochopení, jak jsou čísla reprezentována v počítači a jaké důsledky z toho plynou, zejména co se týče přesnosti a chyb ve výpočtech.

2.1 Reprezentace čísel v počítači

Definice 2.1 (Poziční zápis čísla)**.** Buď $\beta \in \mathbb{N}, \beta \geq 2$ základ číselné soustavy. Libovolné reálné číslo x s konečným počtem cifer x_k $(0 \leq x_k < \beta)$ lze zapsat v poziční soustavě jako

$$x_\beta = (-1)^s [x_n x_{n-1} \dots x_1 x_0 . x_{-1} x_{-2} \dots x_{-m}], \tag{2.1}$$

kde $s \in \{0, 1\}$ určuje znaménko a typicky požadujeme $x_n \neq 0$. Alternativně lze tento zápis vyjádřit pomocí součtu mocnin základu β:

$$x_\beta = (-1)^s \left(\sum_{k=-m}^{n} x_k \beta^k \right). \tag{2.2}$$

Věta 2.2. Libovolné reálné číslo $x \in \mathbb{R}$ lze s libovolnou přesností aproximovat reálným číslem x_β, jehož zápis v soustavě o základě β má konečný počet cifer.

Důkaz. Stačí volit počet cifer m dostatečně velký. □

2.1.1 Čísla s pohyblivou desetinnou čárkou

Poziční zápis naráží na praktické problémy při ukládání velmi velkých nebo velmi malých čísel, protože by vyžadoval ukládání velkého množství nevýznamových nul. Proto se v počítačích používá reprezentace čísel v pohyblivé desetinné čárce (floating-point).

Definice 2.3 (Zápis v pohyblivé desetinné čárce)**.** Číslo x zapisujeme ve tvaru

$$x = (-1)^s \cdot m \cdot \beta^{e-t}, \tag{2.3}$$

kde:

- $s \in \{0, 1\}$ je **znaménko**,
- β je **základ** soustavy (typicky 2 nebo 10),
- $t \in \mathbb{N}$ je **přesnost**, tedy maximální počet povolených cifer v mantise,
- $m = (a_1 a_2 \dots a_t)_\beta$ je **mantisa**, celé číslo tvořené t ciframi $0 \leq a_i < \beta$,

- $e \in \mathbb{Z}$ je **exponent**.

Příklad 2.4. Ukážeme si, jak zapsat číslo 928.371 v systému s pohyblivou desetinnou čárkou. Budeme vycházet z definice $x = (-1)^s \cdot (0.a_1a_2 \dots a_t) \cdot \beta^e$.

1. **Číslo k reprezentaci:**
$$x = 928.371$$
2. **Volba základu:** Pro jednoduchost zvolíme desítkovou soustavu, takže základ $\beta = 10$.
3. **Normalizace:** Musíme číslo upravit tak, aby bylo ve tvaru 0.něco. Toho dosáhneme posunutím desetinné čárky o tři místa doleva.
$$928.371 = 0.928371 \times 1000 = 0.928371 \cdot 10^3$$
4. **Identifikace jednotlivých hodnot:** Z normalizovaného tvaru $0.928371 \cdot 10^3$ můžeme vyčíst všechny potřebné hodnoty:
   - **Znaménko (s):** Číslo je kladné, takže $s = 0$.
   - **Základ (β):** $\beta = 10$.
   - **Mantisa:** Je tvořena ciframi $a_1 = 9, a_2 = 2, a_3 = 8, a_4 = 3, a_5 = 7, a_6 = 1$.
   - **Přesnost (t):** Počet cifer v mantise je 6, takže $t = 6$.
   - **Exponent (e):** Exponent je $e = 3$.

Výsledný zápis čísla 928.371 v normalizovaném tvaru s pohyblivou desetinnou čárkou (při základu 10) je tedy $\mathbf{0.928371 \cdot 10^3}$.

Definice 2.5 (Normalizovaný zápis)**.** Pro jednoznačnost zápisu požadujeme, aby první cifra mantisy byla nenulová, tj. $a_1 \neq 0$. Takovému zápisu se říká **normalizovaný**. Bez této podmínky by například platilo $1 = 0.100 \cdot 10^1 = 0.010 \cdot 10^2$.

Poznámka 2.6. Všimněme si, že normalizovaný zápis neumožňuje zapsat číslo nula. Proto do naší množiny povolených čísel musíme nulu přidat explicitně.

Definice 2.7 (Množina čísel s pohyblivou desetinnou čárkou)**.** Množinu všech čísel, která lze v daném systému s pohyblivou desetinnou čárkou reprezentovat, definujeme jako

$$\mathbb{F}(\beta, t, L, U) = \{0\} \cup \left\{ x \in \mathbb{R} \mid x = (-1)^s \beta^e \sum_{i=1}^{t} a_i \beta^{-i}, a_1 \neq 0, L \leq e \leq U \right\}, \tag{2.4}$$

kde L a U jsou minimální a maximální povolená hodnota exponentu. Číslo nula není možné zapsat v normalizovaném tvaru a musí být definováno speciálně.

2.1.2 Standard IEEE 754

Pro sjednocení reprezentace čísel v počítačích byl zaveden standard IEEE 754, který definuje formáty s různou přesností. Nejběžnější jsou:

- **Jednoduchá přesnost (single precision):** Používá 32 bitů (4 bajty).
  - 1 bit pro znaménko
  - 8 bitů pro exponent (rozsah $L = -126, U = 127$ – dohromady 254 hodnot, dvě zbývající (8 bitů nám umožňuje zapsat 256 hodnot) jsou vyhrazeny pro speciální případy, viz dále)
  - 23 bitů pro mantisu (rozah 0-8388608)
- **Dvojitá přesnost (double precision):** Používá 64 bitů (8 bajtů).

- 1 bit pro znaménko
- 11 bitů pro exponent (rozsah $L = -1022, U = 1023$ – 2 bity opět vyhrazeny pro speciální hodnoty)
- 52 bitů pro mantisu (rozsah až 15 cifer v desítkové soustavě)

Poznámka 2.8 (Speciální hodnoty)**.** Standard IEEE 754 dále definuje speciální hodnoty pomocí rezervovaných hodnot exponentu:

Hodnota	**Exponent**	**Mantisa**
± 0	$L-1$	0
$\pm\infty$	$U+1$	0
NaN (Not a Number)	$U+1$	$\neq 0$

Hodnota NaN vzniká při nedefinovaných operacích, jako je dělení nulou nebo odmocnina ze záporného čísla.

2.1.3 Zaokrouhlovací chyby

Množina strojových čísel $\mathbb{F}(\beta, t, L, U)$ má dvě zásadní vlastnosti: je konečná a její prvky nejsou na reálné ose rozloženy rovnoměrně. S rostoucí velikostí čísla se zvětšují i mezery mezi sousedními reprezentovatelnými čísly.

Příklad 2.9 (Rozložení strojových čísel)**.** Uvažujme jednoduchý systém $\mathbb{F}(10, 1, -2, 2)$. Množina kladných reprezentovatelných čísel v tomto systému je:

- pro $e = -2$: $\{0.01, 0.02, \ldots, 0.09\}$ (rozestup 0.01)
- pro $e = -1$: $\{0.1, 0.2, \ldots, 0.9\}$ (rozestup 0.1)
- pro $e = 0$: $\{1, 2, \ldots, 9\}$ (rozestup 1)
- pro $e = 1$: $\{10, 20, \ldots, 90\}$ (rozestup 10)
- pro $e = 2$: $\{100, 200, \ldots, 900\}$ (rozestup 100)

Z tohoto příkladu je patrné, že čím dále jsme od nuly, tím "řidčeji"jsou čísla na reálné ose rozložena.

Důsledkem těchto mezer je, že množina $\mathbb{F}$ není uzavřená vůči základním aritmetickým operacím. Výsledek operace dvou strojových čísel může snadno padnout do mezery mezi nimi a sám tedy do množiny $\mathbb{F}$ patřit nemusí.

Příklad 2.10. V systému $\mathbb{F}(10, 1, -2, 2)$ jsou čísla $x_1 = 1$ a $x_2 = 0.1$ reprezentovatelná. Jejich součet $x_1 + x_2 = 1.1$ ale do této množiny nepatří, protože by k jeho zápisu byly potřeba dvě platné cifry v mantise.

Poznámka 2.11 (Zaokrouhlování)**.** Abychom mohli výsledek aritmetické operace v počítači uložit, musíme ho **zaokrouhlit** na nejbližší číslo z množiny $\mathbb{F}$. Tímto procesem vznikají **zaokrouhlovací chyby**, které jsou fundamentálním zdrojem nepřesností v numerických výpočtech. Aritmetické operace provedené v pohyblivé desetinné čárce (tedy s následným zaokrouhlením) budeme značit s indexem fl, například $+_{fl}, \cdot_{fl}$ atd..

Tento proces zaokrouhlování může vést k neintuitivním výsledkům a ztrátě přesnosti.

Příklad 2.12 (Vliv zaokrouhlení na sčítání)**.** Mějme systém $\mathbb{F}(10, 2, -5, 5)$ a počítejme součet deseti čísel 0.1 a jednoho čísla 10. Pokud začneme sčítat od velkého čísla:

$$10 +_{fl} 0.1 = 10$$

Již první operace $10 + 0.1 = 10.1$ musí být zaokrouhlena. V systému s přesností $t = 2$ (dvě platné cifry v mantise) se 10.1 zapíše jako $0.10 \cdot 10^2$. Cifra 1 na pozici desetin se ztratí. Každé další přičtení 0.1 tedy nepřinese žádnou změnu a konečný výsledek bude 10. Pokud ale sečteme stejná čísla v opačném pořadí:

$$(\ldots((0.1 +_{fl} 0.1) +_{fl} \ldots) +_{fl} 10)$$

Nejprve sečteme desetkrát 0.1. Mezivýsledky $(0.2, \ldots, 1.0)$ jsou přesně reprezentovatelné. Po deseti krocích

dostaneme 1.0. Až nakonec provedeme operaci $1.0 +_{fl} 10 = 11$. Výsledek je 11, což je správná hodnota.

Poznámka 2.13. Z předchozích příkladů plyne důležitý poznatek: při sčítání čísel, jejichž řády se výrazně liší, může mantisa menšího čísla „spadnout“ mimo rozsah přesnosti většího čísla. Menší číslo je pak při součtu zcela "pohlceno"a výsledek je nepřesný. Je tedy numericky výhodnější sčítat čísla od nejmenšího po největší. Chtělo by se tedy říct, že bychom si před jakýmkoliv sčítáním měli seřadit čísla podle velikosti. To ale není vždy možné, hlavně třeba proto, že řazení je výpočetně náročné.

Poznámka 2.14. Pro většinu inženýrských úloh je jednoduchá přesnost (single) nedostatečná právě kvůli těmto efektům. Dvojitá přesnost (double) se pro většinu reálných problémů ukazuje jako dostatečná, neboť se obvykle pracuje s daty, která jsou řádově srovnatelná.

2.2 Stabilita úlohy a podmíněnost matic

Při výpočtech se nevyhneme chybám – ať už zaokrouhlovacím, nebo chybám ve vstupních datech, které jsou často zatíženy nepřesností měření. Je proto klíčové zkoumat, jak se tyto malé nepřesnosti projeví na výsledku. Obecnou úlohu můžeme formulovat jako hledání řešení $\vec{x}$ rovnice $F(\vec{x}, \vec{d}) = \vec{0}$, kde $\vec{d}$ jsou vstupní data (parametry) úlohy.

Příklad 2.15. Příklady formulace úloh:

- **Řešení lineární soustavy:** $F(\vec{x}, \vec{d}) = \mathbb{A}\vec{x} - \vec{b} = \vec{0}$, kde data jsou $\vec{d} = \{\mathbb{A}, \vec{b}\}$ a řešení je $\vec{x}$.
- **Výpočet inverzní matice:** $F(\mathbb{X}, \vec{d}) = \mathbb{A}\mathbb{X} - \mathbb{I} = \mathbb{O}$, kde data jsou $\vec{d} = \{\mathbb{A}\}$ a řešení je $\mathbb{X} = \mathbb{A}^{-1}$.
- **Výpočet vlastních čísel:** $F(\vec{\lambda}, \vec{d}) = \det(\mathbb{A} - \lambda_i \mathbb{I}) = 0$ pro $i = 1, \ldots, n$, kde data jsou $\vec{d} = \{\mathbb{A}\}$ a řešení je vektor vlastních čísel $\vec{\lambda}$.

Definice 2.16 (Stabilita úlohy)**.** Řekneme, že úloha je **dobře položená** (well-posed) nebo **stabilní**, pokud má jednoznačné řešení, které spojitě závisí na vstupních datech. To znamená, že malá změna vstupních dat $\delta\vec{d}$ způsobí pouze malou změnu v řešení $\delta\vec{x}$. Pokud úloha tuto podmínku nesplňuje, nazývá se **špatně položená** (ill-posed) nebo **nestabilní**.

Pro úlohu řešení soustavy lineárních rovnic je míra stability popsána tzv. číslem podmíněnosti matice.

Definice 2.17 (Číslo podmíněnosti matice)**.** Pro regulární čtvercovou matici $\mathbb{A} \in \mathbb{C}^{n,n}$ definujeme **číslo podmíněnosti** jako

$$\kappa(\mathbb{A}) = ||\mathbb{A}|| \cdot ||\mathbb{A}^{-1}||. \tag{2.5}$$

Je-li $\kappa(\mathbb{A})$ „malé“ (blízké 1), říkáme, že matice je **dobře podmíněná**. Je-li $\kappa(\mathbb{A})$ velké, matice je **špatně podmíněná**.

Příklad 2.18 (Špatně podmíněná soustava)**.** Uvažujme soustavu lineárních rovnic $\mathbb{A}\vec{x} = \vec{b}$:

$$\begin{pmatrix} 20 & 16 & 9 \\ 14 & 15 & 11 \\ 13 & 17 & 14 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} 45 \\ 40 \\ 44 \end{pmatrix}.$$

Přesným řešením této soustavy je vektor $\vec{x} = (1, 1, 1)^T$. Podívejme se, jak se řešení změní při malé změně (perturbaci) vektoru pravé strany $\vec{b}$:

- Pokud pravou stranu změníme o 0.1 na $\vec{b} = (45.1, 39.9, 44.1)^T$, řešením je $\vec{x} = (-12.5, 32, -24.1)^T$.
- Pokud pravou stranu změníme pouze o 0.01 na $\vec{b} = (45.01, 39.99, 44.01)^T$, řešením je $\vec{x} = (-0.25, 4.1, -1.51)^T$.

Vidíme, že nepatrná změna ve vstupních datech vedla k obrovské změně ve výsledném řešení. Taková úloha je špatně položená a matice soustavy je špatně podmíněná.

Věta 2.19 (Odhad chyby řešení)**.** Nechť $\mathbb{A} \in \mathbb{C}^{n,n}$ je regulární matice a nechť $\vec{x}$ je řešením soustavy $\mathbb{A}\vec{x} = \vec{b}$. Pokud řešíme soustavu s porušenou pravou stranou $\mathbb{A}(\vec{x} + \delta\vec{x}) = \vec{b} + \delta\vec{b}$, pak pro relativní chybu řešení platí

odhad:

$$\frac{||\delta\vec{x}||}{||\vec{x}||} \leq \kappa(\mathbb{A})\frac{||\delta\vec{b}||}{||\vec{b}||}. \tag{2.6}$$

Jde-li o indukovanou maticovou normu, pak existuje nenulový vektor $\vec{b}$ a nenulová perturbace $\delta\vec{b}$ taková, že nastává rovnost. Tento vztah ukazuje, že číslo podmíněnosti $\kappa(\mathbb{A})$ funguje jako "zesilovač"relativní chyby vstupních dat. Pro dobře podmíněné matice (s $\kappa(\mathbb{A}) \approx 1$) bude relativní chyba řešení srovnatelná s relativní chybou dat. Pro špatně podmíněné matice (s velkým $\kappa(\mathbb{A})$) může být relativní chyba řešení mnohonásobně větší. (TODO: přepsat odstavec lepšími slovy)

Důkaz. Mějme soustavu LAR $\mathbb{A}\vec{x} = \vec{b}$. Použitím souhlasné maticové a vektorové normy dostáváme

$$||\vec{b}|| = ||\mathbb{A}\vec{x}|| \leq ||\mathbb{A}|| \cdot ||\vec{x}||. \tag{2.7}$$

Z předpokladů také víme, že

$$\begin{aligned} \mathbb{A}(\vec{x} + \delta\vec{x}) &= \vec{b} + \delta\vec{b} \\ \mathbb{A}\vec{x} + \mathbb{A}\delta\vec{x} &= \vec{b} + \delta\vec{b} \\ \mathbb{A}\delta\vec{x} &= \delta\vec{b} \\ \delta\vec{x} &= \mathbb{A}^{-1}\delta\vec{b}. \end{aligned} \tag{2.8}$$

Potom opět s využitím souhlasné maticové a vektorové normy platí:

$$||\delta\vec{x}|| = ||\mathbb{A}^{-1}\delta\vec{b}|| \leq ||\mathbb{A}^{-1}|| \cdot ||\delta\vec{b}||. \tag{2.9}$$

Potom když jednotlivé nerovnosti pronásobíme, dostáváme:

$$||\vec{b}|| \cdot ||\delta\vec{x}|| \leq ||\mathbb{A}|| \cdot ||\mathbb{A}^{-1}|| \cdot ||\delta\vec{b}|| \cdot ||\vec{x}|| = \kappa(\mathbb{A})||\delta\vec{b}|| \cdot ||\vec{x}||. \tag{2.10}$$

Nyní už jen stačí nerovnost přeuspořádat a dostáváme první část tvrzení:

$$\frac{||\delta\vec{x}||}{||\vec{x}||} \leq \kappa(\mathbb{A})\frac{||\delta\vec{b}||}{||\vec{b}||}. \tag{2.11}$$

Nechť nyní navíc $||\bullet||$ indukovaná maticová norma. Potom protože je indukovaná vektorovou normou, platí

$$||\mathbb{A}|| = \max_{||\vec{x}||=1} ||\mathbb{A}\vec{x}|| \tag{2.12}$$

Proto $\exists\,\vec{x}$ takové, že $||\vec{x}|| = 1$ a $||\mathbb{A}|| = ||\mathbb{A}\vec{x}||$, tzn. lze pro tento vektor psát

$$||\mathbb{A}\vec{x}|| = ||\mathbb{A}|| = ||\mathbb{A}|| \cdot 1 = ||\mathbb{A}|| \cdot ||\vec{x}||. \tag{2.13}$$

Potom pokud se vrátíme k původní soustavě $\mathbb{A}\vec{x} = \vec{b}$, můžeme psát

$$||\vec{b}|| = ||\mathbb{A}\vec{x}|| = ||\mathbb{A}|| \cdot ||\vec{x}|| \tag{2.14}$$

Dále provedeme analogii pro perturbaci $\delta\vec{b}$, přičemž vyjdeme z rovnice $\delta\vec{x} = \mathbb{A}^{-1}\delta\vec{b}$. Z definice indukované normy tedy plyne, že $\exists\,\delta\vec{b}$ takové, že $||\delta\vec{b}|| = 1$ a

$$||\mathbb{A}^{-1}|| = ||\mathbb{A}^{-1}\delta\vec{b}|| = ||\mathbb{A}^{-1}|| \cdot 1 = ||\mathbb{A}^{-1}|| \cdot ||\delta\vec{b}||. \tag{2.15}$$

Odtud tedy

$$||\delta\vec{x}|| = ||\mathbb{A}^{-1}\delta\vec{b}|| = ||\mathbb{A}^{-1}|| \cdot ||\delta\vec{b}||. \tag{2.16}$$

Stejně jako v prvním případě pronásobíme dohromady rovnosti:

$$||\delta\vec{x}|| \cdot ||\vec{b}|| = ||\mathbb{A}|| \cdot ||\mathbb{A}^{-1}|| \cdot ||\delta\vec{b}|| \cdot ||\vec{x}|| = \kappa(\mathbb{A})||\delta\vec{b}|| \cdot ||\vec{x}||. \tag{2.17}$$

Přeuspořádáním dostaneme druhou část tvrzení:

$$\frac{||\delta\vec{x}||}{||\vec{x}||} = \kappa(\mathbb{A})\frac{||\delta\vec{b}||}{||\vec{b}||}. \tag{2.18}$$

□

2.3 Klasifikace numerických metod

Definice 2.20 (Stabilita numerické metody)**.** Řekneme, že numerická metoda je **stabilní**, pokud při její aplikaci na stabilní (dobře položenou) úlohu způsobí malá změna vstupních parametrů jen malou změnu ve výsledku. Tyto malé změny jsou v praxi často způsobeny zaokrouhlovacími chybami počítačové aritmetiky.

Základní dělení numerických metod je na přímé a iterační.

2.3.1 Přímé metody

- Po konečném počtu kroků dávají teoreticky přesné řešení (pokud by se počítalo v exaktní aritmetice). V praxi je výsledek ovlivněn zaokrouhlovacími chybami.
- Řešení je k dispozici až na samém konci výpočtu; neposkytují průběžné aproximace.
- Jsou obecně robustnější, tj. fungují pro širší třídu úloh.
- Z těchto důvodů jsou upřednostňovány v některých kritických průmyslových aplikacích.

2.3.2 Iterační metody

- Generují posloupnost aproximací $\{\vec{x}^{(k)}\}_{k=1}^{\infty}$, která (v ideálním případě) konverguje k přesnému řešení.
- Jedna iterace by měla být výpočetně výrazně levnější než kompletní přímá metoda.
- Jsou často snazší na implementaci.
- Mohou být výrazně rychlejší, pokud je k dispozici dobrý počáteční odhad řešení.
- Většina numerických metod pro složité problémy je iteračních.

2.4 Základní pojmy analýzy iteračních metod

Pro iterační metody je klíčové umět posoudit, jak blízko je aktuální aproximace $\vec{x}^{(k)}$ skutečnému řešení $\vec{x}$, abychom věděli, kdy výpočet ukončit.

Definice 2.21 (Reziduum)**.** Nechť je dána úloha $F(\vec{x}, \vec{d}) = \vec{0}$. Pro k-tou aproximaci řešení $\vec{x}^{(k)}$ nazýváme **reziduem** číslo

$$r^{(k)} = ||F(\vec{x}^{(k)}, \vec{d})||, \tag{2.19}$$

kde $||\cdot||$ je nějaká vhodná vektorová norma. Reziduum tedy měří, jak dobře aproximace splňuje původní rovnici. Pro soustavu $\mathbb{A}\vec{x} = \vec{b}$ je reziduum $r^{(k)} = ||\mathbb{A}\vec{x}^{(k)} - \vec{b}||$.

Poznámka 2.22. Malá hodnota rezidua bohužel nemusí nutně znamenat malou chybu řešení $||\vec{x}^{(k)} - \vec{x}||$. Pro špatně podmíněné úlohy můžeme mít velmi malé reziduum, a přesto být daleko od skutečného řešení. V praxi je ale reziduum často jediným dostupným ukazatelem kvality aproximace. (TODO rozepsané je to dále u iteračních metod, sjednotit)

2.4.1 Apriorní a aposteriorní odhady

Odhady chyb dělíme na dva typy:

- **Apriorní odhady:** Lze je stanovit před zahájením vlastního výpočtu, pouze na základě znalosti úlohy (funkce F a dat $\vec{d}$)[1]. Mají spíše teoretický význam a často definují **řád metody** $r \in \mathbb{N}^+$, který v odhadu $||\vec{x}^{(k)} - \vec{x}|| \leq C||\vec{x}^{(k-1)} - \vec{x}||^r$ určuje rychlost konvergence[2].
- **Aposteriorní odhady:** Využívají kromě znalosti úlohy (funkce F a dat $\vec{d}$) navíc znalosti[3] napočítaných aproximací $\vec{x}^{(k)}$. Měly by být přesnější a slouží jako základ pro praktická ukončovací kritéria a pro adaptivní metody, které přizpůsobují svůj běh konkrétní řešené úloze. Odhad bývá ve tvaru $||\vec{x}^{(k)} - \vec{x}|| \leq C||\vec{x}^{(k)} - \vec{x}^{(k-1)}||^r$.

2.5 Zdroje chyb v numerických simulacích

Při řešení reálných problémů pomocí počítačových simulací se setkáváme se čtyřmi hlavními zdroji chyb:

1. **Chyba modelu:** Vzniká při zjednodušení reálného problému do podoby matematického modelu (např. zanedbání tření).
2. **Chyba měření:** Vstupní data jsou často získána experimentálně a jsou zatížena chybou měření.
3. **Chyba diskretizace (metody):** Vzniká nahrazením spojitého matematického modelu (např. diferenciální rovnice) diskrétním modelem (např. soustavou lineárních rovnic).
4. **Zaokrouhlovací chyba:** Vzniká z důvodu použití aritmetiky s konečnou přesností v počítači.

2.6 Otázky

- Reprezentace čísel s pohyblivou desetinnou čárkou
- Jednoduchá a dvojitá přesnost
- Zaokrouhlovací chyby
- Podmíněnost matice
- Řád metody

[1] Zde se projeví prostřednictvím konstanty $C(F, \vec{d}) > 0$, v pozdějších kapitolách uvidíme příklady konkrétních tvarů.

[2] V praxi se ukazuje, že výhodné jsou např. metody druhého řádů, protože stačí napočítat polovinu iterací k získání stejně přesné aproximace v porovnání s metodou prvního řádu. Metody vyšších řádů než dva bývají bohužel často velmi nestabilní, a např. špatný počáteční odhad může zpsůobit, že metoda ani nezkonverguje.

[3] Proto $C(F, \vec{d}\vec{x}^{(k)})$ už není konstanta, jelikož se mění s $\vec{x}^{(k)}$.

Kapitola 3

Přímé metody řešení soustav lineárních rovnic

3.1 Gaussova eliminační metoda

Jde o přímou metodu pro řešení soustavy lineárních algebraických rovnic nebo pro výpočet inverzní matice. Její základní myšlenkou je postupné zjednodušování soustavy rovnic pomocí elementárních řádkových operací, které neovlivňují řešení soustavy. V celé této části budeme předpokládat, že matice $\mathbb{A} \in \mathbb{C}^{n,n}$ jsou regulární. Samotná metoda se skládá ze dvou fází:

- **Přímý chod** : spočívá v převedení úlohy $\mathbb{A}\vec{x} = \vec{b}$ na úlohu $\mathbb{U}\vec{x} = \vec{d}$ se stejným řešením $\vec{x}$, kde $\mathbb{U}$ je ale horní trojúhelníková matice získaná elementárními řádkovými operacemi a $\vec{d}$ stejně modifikovaná pravá strana.
- **Zpětný chod** : spočívá v řešení soustavy $\mathbb{U}\vec{x} = \vec{d}$ pomocí zpětné substituce, tedy postupného dosazování do rovnic od poslední k první.

Přímý chod

Mějme matici $\mathbb{A} \in \mathbb{C}^{n,n}$ a vektor $\vec{b} \in \mathbb{C}^n$. Cílem přímého chodu je převést soustavu rovnic $\mathbb{A}\vec{x} = \vec{b}$ na soustavu rovnic $\mathbb{U}\vec{x} = \vec{d}$, kde $\mathbb{U}$ je horní trojúhelníková matice a $\vec{d}$ je modifikovaný vektor pravých stran.

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix} \tag{3.1}$$

Předpokládejme, že první diagonální prvek a_{11} je nenulový. Nazveme ho hlavním prvkem nebo pivotem a podělíme první řádek soustavy číslem a_{11}. Dostáváme

$$\begin{pmatrix} 1 & u_{12} & \cdots & u_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} d_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}, \tag{3.2}$$

kde

$$u_{1j} = \frac{a_{1j}}{a_{11}}$$

pro $j = 2, \ldots, n$ a $d_1 = \frac{b_1}{a_{11}}$. Nyní odečteme a_{i1} násobek prvního řádku od každého následujícího řádku, čímž vynulujeme první sloupeček od pro $i = 2, ...n$. Výsledný tvar nové soustavy bude

$$\begin{pmatrix} 1 & u_{12} & \cdots & u_{1n} \\ 0 & a_{22}^{(1)} & \cdots & a_{2n}^{(1)} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & a_{n2}^{(1)} & \cdots & a_{nn}^{(1)} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} d_1 \\ b_2^{(1)} \\ \vdots \\ b_n^{(1)} \end{pmatrix}, \tag{3.3}$$

kde $a_{ij}^{(1)} = a_{ij} - a_{i1}u_{1j}$ pro $i = 2, \ldots, n$ a $b_i^{(1)} = b_i - a_{i1}d_1$ pro $i = 2, \ldots, n$. Horní index (1) značí, že se jedná o první krok přímého chodu[1]. Nyní budeme tento postup opakovat $n-1$-krát, postupně s pivotem v druhém řádku, třetím, atd. až po poslední řádek. Na konci bude soustava vypadat takto:

$$\begin{pmatrix} 1 & u_{12} & u_{13} & \cdots & u_{1n} \\ 0 & 1 & u_{23} & \cdots & u_{2n} \\ \vdots & \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & u_{n-1,n} \\ 0 & 0 & \cdots & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_{n-1} \\ x_n \end{pmatrix} = \begin{pmatrix} d_1 \\ d_2 \\ \vdots \\ d_{n-1} \\ d_n \end{pmatrix}. \tag{3.4}$$

kde v k-tém kroku počítáme pro $i, j = k+1, ..., n$

$$\begin{aligned} u_{kj} &= \frac{a_{kj}^{(k-1)}}{a_{kk}^{(k-1)}}, d_k = \frac{b_k^{(k-1)}}{a_{kk}^{(k-1)}} \quad \text{je dělení } k\text{-tého řádku pivotem } a_{kk}^{k-1} \\ a_{ij}^{(k)} &= a_{ij}^{(k-1)} - a_{ik}^{(k-1)}u_{kj}, \quad \text{je eliminace prvků pod diagonálou v } k\text{-tém sloupci} \\ b_i^{(k)} &= b_i^{(k-1)} - a_{ik}^{(k-1)}d_k. \end{aligned} \tag{3.5}$$

Zpětný chod

Nyní řešíme soustavu rovnic $\mathbb{U}\vec{x} = \vec{d}$, kde $\mathbb{U}$ je horní trojúhelníková matice získaná v přímém chodu a $\vec{d}$ je modifikovaný vektor pravých stran. Z poslední rovnice je snadno vidět, že $x_n = d_n$. Z předposlední rovnice s touto znalostí pak snadno dostaneme $x_{n-1} = d_{n-1} - u_{n-1,n}x_n$. Obacně pro $k = n, ...1$

$$x_k = d_k - \sum_{i=k+1}^{n} u_{ki}x_i. \tag{3.6}$$

3.1.1 Implementace

TODO: Implementace Gaussovy eliminační metody v Pythonu.

3.1.2 Analýza složitosti

TODO: Rozepsat podrobně, z implementace je ale zřejmé, že přímý chod používá tři for cykly závislé na n, tedy $O(n^3)$, zpětný chod používá dva for cykly závislé na n, tedy $O(n^2)$. Celková složitost je tedy $O(n^3)$.

3.1.3 Numerická analýza

Abychom mohli lépe analyzovat numerické vlastnosti Gaussovy eliminační metody, převedeme její průběh do maticového zápisu. K tomu budeme nejdříve potřebovat několik pojmů.

Definice 3.1. Elementární úpravou provedenou v obdélníkové matici $\mathbb{A} \in \mathbb{C}^{n,m}$ nazveme:

- násobení všech prvků zvoleného i-tého řádku, resp. sloupce, nenulovou konstantou $\alpha \in \mathbb{C}$,

[1] Často budeme používat také značení $a_{ij}^{(0)} = a_{ij}$, tedy prvky, které byly v původní soustavě.

- pro $\alpha \in \mathbb{C}$ přičtení α-násobku prvků zvoleného i-tého řádku, resp. sloupce, k prvkům zvoleného j-tého řádku, resp. sloupce.

Poznámka 3.2. Násobení řádku, resp. sloupce, matice $\mathbb{A} \in \mathbb{C}^{n,m}$ konstantou $\alpha \in \mathbb{C}$ je ekvivalentní násobení matice $\mathbb{A}$ zleva, resp. zprava, maticí

$$\begin{pmatrix} 1 & 0 & & & & & \\ 0 & \ddots & 0 & & & & \\ & \ddots & 1 & \ddots & & & \\ & & 0 & \alpha & 0 & & \\ & & & \ddots & 1 & \ddots & \\ & & & & \ddots & \ddots & 0 \\ & & & & & 0 & 1 \end{pmatrix} \tag{3.7}$$

s číslem α na diagonále v i-tém řádku, resp. sloupci.

Poznámka 3.3. Přičtení α-násobku i-tého řádku, resp. sloupce, k j-tému řádku, resp. sloupci, je ekvivalentní násobení matice $\mathbb{A} \in \mathbb{C}^{n,m}$ zleva, resp. zprava, maticí

$$\begin{pmatrix} 1 & 0 & & & & & \\ 0 & \ddots & 0 & & & & \\ & \ddots & 1 & \ddots & & & \\ & & \ddots & \ddots & \ddots & & \\ & & \alpha & \ddots & 1 & \ddots & \\ & & & & \ddots & \ddots & 0 \\ & & & & & 0 & 1 \end{pmatrix} \tag{3.8}$$

s jedničkami na diagonále a číslem α na pozici (j, i).

Důkaz. Poctivý čtenář snadno ověří tyto vlastnosti z definice maticového násobení, důkaz je také k nalezení na str. TODO [2]. □

Příklad 3.4. TODO: doplnit ukázkový příklad

Poznámka 3.5. Provedeme-li na řádky, resp. sloupce, matice $\mathbb{A} \in \mathbb{C}^{n,m}$ konečný počet elementárních úprav, je výsledek stejný jako když matici $\mathbb{A}$ vynásobíme zleva, resp. zprava, maticí, která je součinem matic odpovídajících jednotlivým elementárním úpravám. (TODO: Doladit znění podle [2])

Důkaz. Důkaz je technický a je třeba ho rozložit na několik případů. Na přednášce neprobíráno, k nalezení je na str. TODO [2]. □

Definice 3.6. Rozšířenou maticí soustavy nazýváme matici $\mathbb{P} \in \mathbb{C}^{n,n+1}$, která vznikne připojením vektoru pravých stran $\vec{b} \in \mathbb{C}^n$ k matici koeficientů $\mathbb{A} \in \mathbb{C}^{n,n}$, tedy

$$\mathbb{P} = (\mathbb{A} \,|\, \vec{b}) = \left(\begin{array}{cccc|c} a_{11} & a_{12} & \cdots & a_{1n} & b_1 \\ a_{21} & a_{22} & \cdots & a_{2n} & b_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} & b_n \end{array}\right) \tag{3.9}$$

Nyní máme vše pro to, abychom mohli zapsat průběh Gaussovy eliminační metody v maticovém zápisu.

Přímý chod

Nechť $\mathbb{P} \in \mathbb{C}^{n,n+1}$ je rozšířená matice soustavy $\mathbb{A}\vec{x} = \vec{b}$. Na konci prvního kroku přímého chodu bude matice $\mathbb{P}^{(1)} \in \mathbb{C}^{n,n+1}$ vypadat takto:

$$\mathbb{P}^{(1)} = \left(\begin{array}{cccc|c} 1 & u_{12} & \cdots & u_{1n} & d_1 \\ 0 & a_{22}^{(1)} & \cdots & u_{2n}^{(1)} & b_2^{(1)} \\ \vdots & \vdots & \ddots & \vdots & \\ 0 & a_{n2}^{(1)} & \cdots & u_{nn}^{(1)} & b_n^{(1)} \end{array}\right), \tag{3.10}$$

čehož bylo dosaženo vhodnými elementárními úpravami matice $\mathbb{P}$. Dělení pivotem odpovídalo vynásobení matice $\mathbb{P}$ zleva maticí

$$\mathbb{M}_1^{(1)} = \begin{pmatrix} \frac{1}{a_{11}} & & & \\ & 1 & & \\ & & \ddots & \\ & & & 1 \end{pmatrix}, \tag{3.11}$$

přičtení $-a_{21}$-násobku prvního řádku k druhému řádku odpovídalo vynásobení matice $\mathbb{P}$ zleva maticí

$$\mathbb{M}_2^{(1)} = \begin{pmatrix} 1 & & & \\ -a_{21} & 1 & & \\ & & \ddots & \\ & & & 1 \end{pmatrix}. \tag{3.12}$$

Dohromady

$$\mathbb{M}^{(1)} = \mathbb{M}_n^{(1)} \cdots \mathbb{M}_1^{(1)} = \begin{pmatrix} \frac{1}{a_{11}} & & & \\ -\frac{a_{21}}{a_{11}} & 1 & & \\ & & \ddots & \\ -\frac{a_{n1}}{a_{11}} & & & 1 \end{pmatrix} \tag{3.13}$$

Tvar matice $\mathbb{M}^{(1)}$ lze samozřejmě ověřit z definice maticového násobení. Intuitivně je ale lepší dívat se postupně na to, jak matice $\mathbb{M}_k^{(1)}$ mění jednotkovou matici $\mathbb{I}$, tzn. například pokud bychom první řádek dělili 2 a odečítali třínásobek prvního řádku od druhého, dostaneme:

$$\mathbb{M}_2\mathbb{M}_1\mathbb{I} = \begin{pmatrix} \frac{1}{2} & 0 & 0 & 0 \\ -3/2 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{3.14}$$

Obecně před zahájením k-tého kroku přímého chodu bude matice $\mathbb{P}^{(k-1)}$ vypadat takto:

$$\mathbb{P}^{(k-1)} = \left(\begin{array}{ccccccc|c} 1 & u_{12} & \ldots & \ldots & u_{1k} & \ldots & u_{1n} & d_1 \\ & 1 & \ldots & \ldots & u_{2k} & \ldots & u_{2n} & d_2 \\ & & \ddots & & \vdots & & \vdots & \vdots \\ & & & 1 & u_{k-1,k} & \ldots & u_{k-1,n} & d_{k-1} \\ & & & & a_{kk}^{(k-1)} & \ldots & a_{kn}^{(k-1)} & b_k^{(k-1)} \\ & & & & \vdots & & \vdots & \vdots \\ & & & & a_{nk}^{(k-1)} & \ldots & a_{nn}^{(k-1)} & b_n^{(k-1)} \end{array}\right) \tag{3.15}$$

na konci k-tého kroku přímého chodu bude matice $\mathbb{P}^{(k)}$ vypadat takto:

$$\mathbb{P}^{(k)} = \left(\begin{array}{ccccccccc|c} 1 & u_{12} & \dots & \dots & u_{1k} & u_{1,k+1} & \dots & u_{1n} & d_1 \\ & 1 & \dots & \dots & u_{2k} & u_{2,k+1} & \dots & u_{2n} & d_2 \\ & & \ddots & & \vdots & \vdots & & \vdots & \vdots \\ & & & 1 & u_{k-1,k} & u_{k-1,k+1} & \dots & u_{k-1,n} & d_{k-1} \\ & & & & 1 & u_{k,k+1} & \dots & u_{kn} & d_k \\ & & & & & a^{(k)}_{k+1,k+1} & \dots & a^{(k)}_{k+1,n} & b^{(k)}_{k+1} \\ & & & & & \vdots & & \vdots & \vdots \\ & & & & & a^{(k)}_{n,k+1} & \dots & a^{(k)}_{nn} & b^{(k)}_n \end{array}\right) \tag{3.16}$$

a matice zprostředkovávající tento krok bude mít tvar:

$$\mathbb{M}^{(k)} = \begin{pmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & & & \\ & & & \frac{1}{a^{(k-1)}_{kk}} & & \\ & & & -\frac{a^{(k-1)}_{k+1,k}}{a^{(k-1)}_{kk}} & 1 & \\ & & & \vdots & & \ddots \\ & & & -\frac{a^{(k-1)}_{nk}}{a^{(k-1)}_{kk}} & & 1 \end{pmatrix} \tag{3.17}$$

Na konci přímého chodu bude matice $\mathbb{P}^{(n)}$ vypadat takto:

$$\mathbb{P}^{(n)} = \left(\begin{array}{ccccc|c} 1 & u_{12} & u_{13} & \cdots & u_{1n} & d_1 \\ 0 & 1 & u_{23} & \cdots & u_{2n} & d_2 \\ \vdots & & & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & u_{n-1,n} & d_{n-1} \\ 0 & 0 & \cdots & 0 & 1 & d_n \end{array}\right), \tag{3.18}$$

a přitom

$$\mathbb{P}^{(n)} = \mathbb{M}^{(n)}\mathbb{M}^{(n-1)} \cdots \mathbb{M}^{(1)}\mathbb{P} = \mathbb{M}\mathbb{P} \tag{3.19}$$

Protože všechny matice $\mathbb{M}^{(k)}$ byly dolní trojúhelníkové (tak jsme si je zavedli), ze znalosti faktu, že součin trojúhelníkových matic je trojúhelníková matice plyne, že i $\mathbb{M}$ je dolní trojúhelníková. V blokovém zápisu, kde $\mathbb{P} = (\mathbb{A} \,|\, \vec{b})$, resp. $\mathbb{P}^{(n)} = (\mathbb{U}, \vec{d})$ pak $\mathbb{M}\mathbb{P} = (\mathbb{M}\mathbb{A} \,|\, \mathbb{M}\vec{b})$, tzn.

$$\mathbb{U} = \mathbb{M}\mathbb{A}, \quad \Rightarrow \quad \mathbb{A} = \mathbb{M}^{-1}\mathbb{U}. \tag{3.20}$$

Samozřejmě poslední implikace platí pouze za předpokladu invertibility matice $\mathbb{M}$. Nyní předpokládejme, že invertibilní je, a více viz dále Připomeňme, že $\mathbb{A}$ jsme uvažovali regulární čtvercovou, $\mathbb{M}$ je dolní trojúhelníková (a její inverze je tudíž také dolní trojúhelníková, viz TODO) a $\mathbb{U}$ je horní trojúhelníková s jedničkami na diagonále. Vidíme, že GEM nějakým způsobem souvisí s rozkladem regulární čtvercové matice $\mathbb{A}$ na součin dolní a horní trojúhelníkové matice. Podívejme se na chvíli na diagonálu matice $\mathbb{M}$. Z TODO víme, že diagonální prvky součinu trojúhelníkových matic odpovídají jednoduše součinu diagonálních prvků původních matic. Z tvaru k-té matice $\mathbb{M}^{(k)}$ tedy zřejmě

$$\operatorname{diag} \mathbb{M} = \left(\frac{1}{a_{11}}, \frac{1}{a^{(1)}_{22}}, ..., \frac{1}{a^{n-1}_{nn}}\right) \tag{3.21}$$

Dále také z TODO víme, že diagonální prvky inverze k trojúhelníkové matici jsou inverze diagonálních prvků původní matice, proto

$$\operatorname{diag} \mathbb{M}^{-1} = \left(a_{11}, a_{22}^{(1)}, ..., a_{nn}^{(n-1)}\right) \tag{3.22}$$

Definujme matici $\mathbb{D} \in \mathbb{C}^{n,n}$ předpisem $\mathbb{D} = \operatorname{diag}\left(a_{11}, a_{22}^{(1)}, ..., a_{nn}^{(n-1)}\right)$. Pak lze psát

$$\mathbb{A} = \mathbb{M}^{-1}\mathbb{U} = \mathbb{M}^{-1}\mathbb{D}^{-1}\mathbb{D}\mathbb{U} = \mathbb{L}\mathbb{D}\mathbb{R} \tag{3.23}$$

kde jsme opět využili vět o práci s trojúhelníkovými maticemi, odkud $\mathbb{M}^{-1}\mathbb{D}^{-1}$ je dolní trojúhleníková s jedničkami na diagonále (ozn. $\mathbb{L}$). Horní trojúhelníkovou matici s jedničkami na diagonále $\mathbb{U}$ jsme už jen přejmenovali na $\mathbb{R}$, abychom dostali dříve ojevený LDR rozklad, viz (1.49). My jsme ale dříve ukázali, že LDR rozklad existuje jen pro silně regulární matice[2]. Nabízí se tedy otázka, zda GEM můžeme provádět pro libovolnou matici, už výše jsme narazili na potřebu invertibility matice $\mathbb{M}$. Ukáže se, že podmínka silné regularity bude splněna poměrně intuitivně, tedy pokud v rámci přímého chodu nenarazíme na nulového pivota (potřebujeme jimi dělit).

Definice 3.7. Řekneme, že *lze provést GEM* právě tehdy, když žádný z pivotů není nulový.

Věta 3.8. Základní GEM *lze provést* právě tehdy, když matice lineární soustavy $\mathbb{A} \in \mathbb{C}^{n,n}$ je silně regulární.

Důkaz. Uvažujme, že pro nějakou matici soustavy $\mathbb{A} \in \mathbb{C}^{n,n}$ provedli $k-1$ kroků přímého chodu, tzn. $\mathbb{A}^{(k)} = \mathbb{M}^{(k-1)} \cdot ... \cdot \mathbb{M}^{(1)}\mathbb{A}$.

$$\mathbb{A}^{(k)} = \begin{pmatrix} 1 & u_{12} & u_{13} & & & & & \\ 0 & 1 & u_{23} & & & & & \\ & 0 & 1 & & & & & \\ & & \ddots & \ddots & & & & \\ & & & 0 & 1 & & & \\ & & & & 0 & a_{kk}^{(k-1)} & \dots & a_{kn}^{(k-1)} \\ & & & & & \vdots & & \vdots \\ & & & & & a_{nk}^{(k-1)} & \dots & a_{nn}^{(k-1)} \end{pmatrix} = \begin{pmatrix} \mathbb{A}_{11}^{(k)} & \mathbb{A}_{12}^{(k)} \\ \mathbb{O} & \mathbb{A}_{22}^{(k)} \end{pmatrix}, \tag{3.24}$$

Předpokládáme, že těchto prvních $k-1$ kroků se povedlo, tzn. nenarazili jsme na žádný nulový pivot. O k-tém pivotu ale zatím nic nevíme, proto v blokovém tvaru uvažujeme $\mathbb{A}_{11}^{(k)} \in \mathbb{C}^{k,k}$, tedy včetně $a_{kk}^{(k-1)}$. Tento člen potřebujeme zahrnout ještě do matice $\mathbb{A}_{11}^{(k)}$, protože v k-tém kroku přímého chodu provedeme dělení k-tého řádku $a_{kk}^{(k-1)}$. (TODO: tady ale pak nesouhlasním s tím, že ten blok označený $\mathbb{O}$ je nulová matice... podle mě má v posledním sloupci potenciálně nenulové prvky) Matici $\mathbb{M}$, kterou jsme dosud sestavili před zahájením k-tého kroku (před dělení potenciálně nulovým $a_{kk}^{(k-1)}$) přímého chodu, můžeme schématicky[3] zapsat jako

$$\mathbb{M} = \begin{pmatrix} \frac{1}{a_{11}} & 0 & & & & & & \\ * & \frac{1}{a_{22}^{(1)}} & 0 & & & & & \\ & * & \frac{1}{a_{33}^{(2)}} & 0 & & & & \\ & & \ddots & \ddots & \ddots & & & \\ & & & * & \frac{1}{a_{k-1,k-1}^{(k-2)}} & 0 & & \\ & & & \vdots & * & 1 & 0 & \\ & & & \vdots & \vdots & & \ddots & 0 \\ & & & * & * & & & 1 \end{pmatrix} \tag{3.25}$$

Je to tedy dolní trojúhelníková matice, a na diagonále jsou v prvních prvních $k-2$ sloupcích převrácené

[2] viz definice 1.17: takové jejíž všechny hlavní subdeterminanty jsou nenulové

[3] Nezajímá nás co přesně je pod diagonálou, jen * naznačíme, kde se nachází nenulové prvky.

hodnoty pivotů a dále jedničky. Proto je také invertibilní, tudíž $\exists \mathbb{M}^{-1}$, kterou označíme $\mathbb{L}$. Odtud $\mathbb{A} = \mathbb{L}\mathbb{A}^{(k)}$ kde diagonální prvky $\mathbb{L}$ jsou převracené hodnoty diagonálních prvků $\mathbb{M}$, tzn.

$$\operatorname{diag} \mathbb{L} = \left(a_{11}, a_{22}^{(1)}, \ldots, a_{nn}^{(n-1)}, 1, \ldots, 1\right) \tag{3.26}$$

Přepíšeme si vztah pro $\mathbb{A}$ opět blokově:

$$\begin{pmatrix} \mathbb{A}_{11} & \mathbb{A}_{12} \\ \mathbb{A}_{21} & \mathbb{A}_{22} \end{pmatrix} = \begin{pmatrix} \mathbb{L}_{11} & \mathbb{O} \\ \mathbb{L}_{21} & \mathbb{L}_{22} \end{pmatrix} \begin{pmatrix} \mathbb{A}_{11}^{(k)} & \mathbb{A}_{12}^{(k)} \\ \mathbb{O} & \mathbb{A}_{22}^{(k)} \end{pmatrix} \tag{3.27}$$

kde $\mathbb{A}_{11}^{(k)}, \mathbb{L}_{11}, \mathbb{A}_{11} \in \mathbb{C}^{k,k}$ (TODO: já si myslím, že by to mělo být $k-1, k-1$...). Odtud

$$\begin{aligned} \mathbb{A}_{11} = \mathbb{L}_{11}\mathbb{A}_{11}^{(k)} \quad \wedge \quad \det \mathbb{A}_{11} &= \det \mathbb{L}_{11} \cdot \det \mathbb{A}_{11}^{(k)} \\ &= \left(a_{11} \cdot a_{22}^{(1)} \cdot \ldots \cdot a_{k-1,k-1}^{(k-2)}\right) \cdot \left(1 \cdot 1 \cdot \ldots \cdot 1 \cdot a_{kk}^{(k-1)}\right) \end{aligned} \tag{3.28}$$

Determinant $\mathbb{A}_{11}$ je tedy součinem prvních k pivotů a tento vztah můžeme použít pro všechna $k \in \hat{n}$. Pokud je matice $\mathbb{A}$ silně regulární, tak $\forall k$ bude $\det \mathbb{A}_{11} \neq 0$ a tudíž všechny pivoty budou nenulové. Předpoklad silné regularity je tedy zřejmě postačující podmínka pro to, aby bylo možné provést GEM. Naopak pokud matice $\mathbb{A}$ není silně regulární, tedy $\exists k$ takové, že $\det \mathbb{A}_{11} = 0$, je součin prvních k pivotů nula. Bez újmy na obecnosti nechť k je první takové, pro které $\det \mathbb{A}_{11} = 0$. Pak $a_{kk}^{(k-1)} = 0$, a zřejmě GEM nelze provést (nemůžeme dělit nulou). Silná regularita tedy představuje také nutnou podmínku. □

3.1.4 Modifikovaná Gaussova eliminační metoda

Pokud matice soustavy není silně regulární, nutně musí nastat situace, že se objeví nulový pivot $a_{kk}^{(k-1)}$, viz důkaz věty (3.8). V takovém případě je třeba Gaussovu eliminaci modifikovat a zvolit za vedoucí prvek některý jiný nenulový z submatice

$$\begin{pmatrix} a_{kk}^{(k-1)} & \cdots & a_{kn}^{(k-1)} \\ \vdots & \ddots & \vdots \\ a_{n,k}^{(k-1)} & \cdots & a_{nn}^{(k-1)} \end{pmatrix}. \tag{3.29}$$

Z regularity matice $\mathbb{A}$ plyne, že nenulový prvek v submatici určitě existovat musí (to lze dokázat např. výpočtem determinantu matice $\mathbb{A}$ rozvojem tak, abychom se dostali k determinantu této submatice). Otázkou ale je, jak vybrat nejlepší nenulový prvek, resp. nejlepšího pivota. Ukáže se, že nejlepší volbou je prvek s největší absolutní hodnotou. Podívejme se nejdříve na to, co by se stalo v aritmetice s konečnou[4] přesností $\mathbb{F}(10, 2, -5, 5)$, kdybychom vzali malého pivota. Mějme následující soustavu, kde vezmeme pivot 10^{-5}:

$$\begin{aligned} \left(\begin{array}{ccc|c} 10^{-5} & 1 & 2 & 1 \\ 3 & 2 & 1 & 1 \\ 2 & 3 & 3 & 3 \end{array}\right) &\sim \left(\begin{array}{ccc|c} 1 & 10^5 & 2 \cdot 10^5 & 1 \cdot 10^5 \\ 3 & 2 & 1 & 1 \\ 2 & 3 & 3 & 3 \end{array}\right) \\ &\sim \left(\begin{array}{ccc|c} 1 & 10^5 & 2 \cdot 10^5 & 1 \cdot 10^5 \\ 0 & 2 - 3 \cdot 10^5 & 1 - 6 \cdot 10^5 & 1 - 3 \cdot 10^5 \\ 0 & 3 - 2 \cdot 10^5 & 3 - 4 \cdot 10^5 & 3 - 2 \cdot 10^5 \end{array}\right) \\ &\sim \left(\begin{array}{ccc|c} 1 & 10^5 & 2 \cdot 10^5 & 1 \cdot 10^5 \\ 0 & -3 \cdot 10^5 & -6 \cdot 10^5 & -3 \cdot 10^5 \\ 0 & -2 \cdot 10^5 & -4 \cdot 10^5 & -2 \cdot 10^5 \end{array}\right). \end{aligned} \tag{3.30}$$

Pak v prvním kroku přímého chodu jsme vydělili první řádek 10^{-5} a dále jsme přičítali -3 násobek prvního řádku k druhému a -2 násobek prvního řádku k třetímu. Všimněme si, že v druhém řádku jsme dostali číslo

[4] Značení $\mathbb{F}(10, 2, -5, 5)$ říká, že počítáme v desítkové soustavě, umíme si pamatovat pouze 2 cifry a exponent se umí pohybovat od -5 do 5.

$1 - 3 \cdot 10^5 = -29999$, což je ale číslo s 5 ciframi, takže ho musíme zaokrouhlit na nejvýše dvě, a tedy v naší aritmetice $1 - 3 \cdot 10^5 = -3 \cdot 10^5$. Poslední matice tedy odpovídá stavu po zaokrouhlení. Vidíme ale, že poslední dva řádky jsou lineárně závislé (a dostáváme singulární matici), k čemuž došlo právě proto, že dělením malým pivotem vznikl extrémně velký řádek, který pak přebyl všechny následující. Mějme nyní naopak následující soustavu, kde vezmeme velký pivot 10^5:

$$\begin{aligned}
\left(\begin{array}{ccc|c} 10^5 & 1 & 2 & 1 \\ 3 & 2 & 1 & 1 \\ 2 & 3 & 3 & 3 \end{array}\right) &\sim \left(\begin{array}{ccc|c} 1 & 10^{-5} & 2 \cdot 10^{-5} & 1 \cdot 10^{-5} \\ 3 & 2 & 1 & 1 \\ 2 & 3 & 3 & 3 \end{array}\right) \\
&\sim \left(\begin{array}{ccc|c} 1 & 10^{-5} & 2 \cdot 10^{-5} & 1 \cdot 10^{-5} \\ 0 & 2 - 3 \cdot 10^{-5} & 1 - 6 \cdot 10^{-5} & 1 - 3 \cdot 10^{-5} \\ 0 & 3 - 2 \cdot 10^{-5} & 3 - 4 \cdot 10^{-5} & 3 - 2 \cdot 10^{-5} \end{array}\right) \\
&\sim \left(\begin{array}{ccc|c} 1 & 10^5 & 2 \cdot 10^5 & 1 \cdot 10^5 \\ 0 & 2 & 1 & 1 \\ 0 & 3 & 3 & 3 \end{array}\right).
\end{aligned} \tag{3.31}$$

Tentokrát jsme po vidělení sice dostali velký řádek, ten ale při následných úpravách zbytek matice ovlivnil jen mále. Ač tedy došlo také k výpočetním chybám způsobených zaokrouhlováním, ty příliš neovlivní další výpočet. Takovouto úvahou jsme tedy ověřili, že pokud chceme minimalizovat chyby způsobené zaokrouhlováním, měli bychom vybírat za pivot prvek s největší absolutní hodnotou ze submatice

$$\begin{pmatrix} a_{kk}^{(k-1)} & \cdots & a_{kn}^{(k-1)} \\ \vdots & \ddots & \vdots \\ a_{n,k}^{(k-1)} & \cdots & a_{nn}^{(k-1)} \end{pmatrix}. \tag{3.32}$$

V praxi se ukazuje, že ani není třeba volit z celé submatice, ale stačí vybírat z k-tého sloupce, protože jiná volba by způosbila nutnost prohazovat sloupce, což je nakonec algoritmicky dražší. Je ale zřejmé, že kdybychom obecně vybírali prvek s největší absolutní hodnotou z k-tého sloupce, resp. k-tého řádku, a pak provedli odpovídající permutaci sloupců, resp. řádků, postupně tím převádíme matici na silně regulární a tedy aplikujeme základní GEM.

3.1.5 Otázky

- Odvození GEM včetně algoritmu
- Pro jaké matice lze GEM použít
- Vysvětlit modifikovanou GEM

3.2 LU rozklad

Z praxe víme, že přímý chod Gaussovy eliminační metody lze provést s několika pravými stranami současně, aniž by se výpočet výrazně zpomalil. Zpětný chod pak sice musíme provést pro každou pravou stranu zvlášť, ten je ale řádově rychlejší, a tak nakonec nepředstavuje významné zpomalení. V praxi ale bohužel často narazíme na situaci, kdy až vyřešení soustavy $\mathbb{A}\vec{x}^{(k)} = \vec{b}^{(k)}$ nám umožní určit další pravou stranu $\vec{b}^{(k+1)}$. Ukážeme si tedy metodu, která umí novou pravou stranu rychle převést na tvar vhodný pro zpětný chod, tedy *LU rozklad*. [TODO lépe popsat, jak se odseparuje přímý chod od závislosti na pravé straně]. Již jsme v (3.20) ukázali, že výsledkem GEMu je rozklad matice $\mathbb{A}$ na horní a dolní trojúhelníkovou, tzn. $\mathbb{A} = \mathbb{M}^-1\mathbb{U} = \mathbb{L}\mathbb{U}$ (matice $\mathbb{M}$ byla dolní trojúhelníková s jedničkami na diagonále, proto matice $\mathbb{M}^{-1}$ je rovněž dolní trojúhelníková s jedničkami na diagonále). Dostáváme tak

$$\mathbb{A}\vec{x} = \vec{b} \iff \mathbb{L}\mathbb{U}\vec{x} = \vec{b}. \tag{3.33}$$

Označíme-li

$$\mathbb{U}\vec{x} = \vec{d}, \tag{3.34}$$

dostaneme

$$\mathbb{L}\vec{d} = \vec{b}. \tag{3.35}$$

V soustavě $\mathbb{L}\vec{d} = \vec{b}$ neznáme pouze vektor $\vec{d}$, pokud tedy tuto soustavu vyřešíme, můžeme pokračovat k $\mathbb{U}\vec{x} = \vec{d}$ a najít $\vec{x}$. Matice $\mathbb{U}$ je horní trojúhelníková s jedničkami na diagonále, a ukáže se, že celá soustava $\mathbb{U}\vec{x} = \vec{d}$ odpovídá zpětnému chodu. Soustavu $\mathbb{L}\vec{d} = \vec{b}$ můžeme zároveň řešit podobně jako zpětný chod, jelikož jediný rozdíl je, že $\mathbb{L}$ je tentokrát horní trojúhelníková. Také ukážeme, že složitost LU rozkladu je n^3, takže pokud bychom metodu aplikovali na soustavu s jednou pravou stranou, nijak si oproti GEMu nepomůžeme. Naopak v případě, kdy máme více pravých stran, můžeme si jednou spočítat LU rozklad se složitostí n^3, a dále napočítávat řešení pro libovolnou pravou stranu se složitostí n^2.

3.2.1 LU rozklad pomocí Gaussovy eliminační metody

Předpokládejme pro jednoduchost, že $\mathbb{A}$ je silně regulární. V případě, že by nebyla, vhodnými sloupcovými nebo řádkovými úpravami ji na silně regulární v každém kroku GEMu umíme převést. Zajímá nás, jak vypadá matice $\mathbb{L} \coloneqq \mathbb{M}^{-1}$. Víme, že matice $\mathbb{M}$ se rovná součinu $\mathbb{M}^{(k)}$, které reprezentují všechny elementární úpravy provedené v k-tém kroku přímého chodu, tj.

$$\mathbb{M} = \mathbb{M}^{(n)}\mathbb{M}^{(n-1)}...\mathbb{M}^{(1)}, \tag{3.36}$$

kde $\mathbb{M}^{(1)}$ reprezentuje první krok přímého chodu, tedy nastavení prvního prvku prvního řádku na 1 a vynulování všech prvků ve sloupci pod ním. Dále

$$\mathbb{M}^{-1} = (\mathbb{M}^{(1)})^{-1}(\mathbb{M}^{(2)})^{-1}...(\mathbb{M}^{(n)})^{-1}. \tag{3.37}$$

Nyní nás proto zajíma jak získat inverzní matici $(\mathbb{M}^{(k)})^{-1}$ pro k-tý krok přímého chodu. Pro tu ale známe předpis:

$$\mathbb{M}^{(k)} = \mathbb{M}_n^{(k)}...\mathbb{M}_k^{(k)}, \tag{3.38}$$

kde matice ze součinu jsou jednotlivé eliminační kroky k-tého kroku, tedy $\mathbb{M}_k^{(k)}$ je nastavení prvního prvku na jedničku a ostatní matice odečítání patřičných násobků od dalších řádků, aby se postupně prvky pod diagonálou vynulovaly. Proto

$$(\mathbb{M}^{(k)})^{-1} = (\mathbb{M}_k^{(k)})^{-1}...(\mathbb{M}_n^{(k)})^{-1}. \tag{3.39}$$

Nyní tedy zbývá určit tyto inverze. My ale známe tvar jednotlivých matic.

$$\mathbb{M}_k^{(k)} = \begin{pmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ & & 1 & & & & \\ & & & \frac{1}{a_{kk}^{(k-1)}} & & & \\ & & & & 1 & & \\ & & & & & \ddots & \\ & & & & & & 1 \end{pmatrix} \tag{3.40}$$

Inverze této matice ji musí převést na jednotkovou, tudíž k-tý řádek musí být jednoduše vynásoben prvek $a_{kk}^{(k-1)}$.

$$(\mathbb{M}_k^{(k)})^{-1} = \begin{pmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ & & 1 & & & & \\ & & & a_{kk}^{(k-1)} & & & \\ & & & & 1 & & \\ & & & & & \ddots & \\ & & & & & & 1 \end{pmatrix} \tag{3.41}$$

Dále pro $i > k$ (odečtení $a_{ik}^{(k-1)}$ násobku k-tého řádku od i-tého):

$$\mathbb{M}_i^{(k)} = \begin{pmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ & & 1 & & & & \\ & & & \ddots & & & \\ & & -a_{ik}^{(k-1)} & & 1 & & \\ & & & & & \ddots & \\ & & & & & & 1 \end{pmatrix} \tag{3.42}$$

Abych dostal jednotkovou matici tentokrát, musím udělat inverzní operaci, tzn. přičtení, proto:

$$(\mathbb{M}_i^{(k)})^{-1} = \begin{pmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ & & 1 & & & & \\ & & & \ddots & & & \\ & & a_{ik}^{(k-1)} & & 1 & & \\ & & & & & \ddots & \\ & & & & & & 1 \end{pmatrix} \tag{3.43}$$

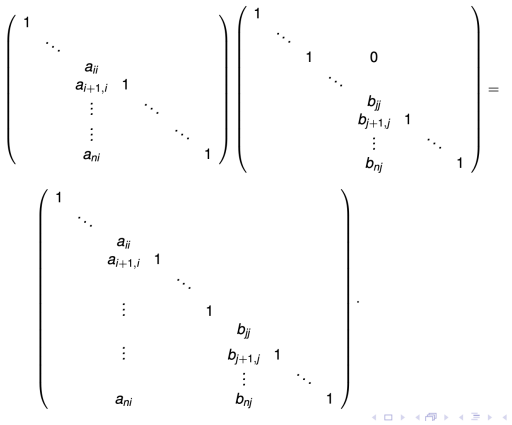
Odtud

$$(\mathbb{M}^{(k)})^{-1} = (\mathbb{M}_k^{(k)})^{-1} ... (\mathbb{M}_n^{(k)})^{-1} = \begin{pmatrix} 1 & & & & \\ & \ddots & & & \\ & & a_{k,k}^{(k-1)} & & \\ & & a_{k+1,k}^{(k-1)} & 1 & \\ & & \vdots & & \ddots \\ & & a_{n,k}^{(k-1)} & & 1 \end{pmatrix}, \tag{3.44}$$

což lze jednoduše odvodit např tak, že si představíme, jak jednotlivé matice ze součinu působí na jednotkovou matici. Nyní bychom potřebovali vědět co se stane, když takto vypadající matice pronásobíme mezi sebou. K tomu poslouží následující lemma.

Důsledek 3.9. Pro $i < j$ platí (TODO viz obr)

Důkaz. Důkaz intutivní [TBD]. □



Obrázek 3.1: TODO Přetexovat

Z lemmatu dostáváme, že

$$\mathbb{M}^{-1} = \begin{pmatrix} a_{11} & & & & & & \\ a_{21} & a_{22}^{(1)} & & & & & \\ a_{31} & a_{32}^{(2)} & a_{33}^{(2)} & & & & \\ \vdots & \vdots & & \ddots & & & \\ a_{k1} & a_{k2}^{(1)} & \dots & a_{k,k-1}^{(k-2)} & a_{kk}^{(k-1)} & & \\ \vdots & \vdots & & & & \ddots & \\ a_{n1} & a_{n2}^{(1)} & \dots & \dots & \dots & \dots & a_{nn}^{(n-1)} \end{pmatrix} \tag{3.45}$$

Všimněme si, že jde o prvky, které jsme v původní matici nulovali. Vše si tedy můžeme ukládat in-place do stejné paměti, kde máme původní matici $\mathbb{A}$. Nad diagonálou bude vznikat matice $\mathbb{U}$, která bude horní trojúhelníková. O ní také víme, že bude mít na diagonále jedničky, což není třeba ukládat, protože to nutně plyne z teorie. Nakonec diagonála a prvky pod diagonálou dají hledanou matici $\mathbb{L}$. V literatuře se často mšto o GEM mluví pouze o LU rozkladu, protože jde v zásadě o ekvivalentní úlohy, jen LU rozklad má větší potenciál, protože poté umožňuje řešit libovolnou pravou stranu $\vec{b}$.

3.2.2 Kompaktní schéma pro LU faktorizaci

Další způsob jak získat LU rozklad je pomocí kompaktního schématu, které je také známé jakou Croutova nebo Doolittleova faktorizace. Ukážeme si tu první, rozdíl v nich je pouze takový, že zatímco Croutova faktorizace má jedničky na diagonále $\mathbb{U}$, Doolittleova je má na diagonále $\mathbb{L}$. Vyjdeme ze vztahu pro násobení matic $\mathbb{L}$ a $\mathbb{U}$:

$$a_{ij} = \sum_{k=1}^{\min(i,j)} l_{ik}u_{kj}. \tag{3.46}$$

Horní mez $\min(i,j)$ plyne z toho, že $\mathbb{L}$ je dolní trojúhelníková matice, a tedy $l_{ik} = 0$ pro $k > i$, a $\mathbb{U}$ je horní trojúhelníková matice, a tedy $u_{kj} = 0$ pro $k > j$. Proto jakmile se dostaneme do $k > i$ nebo $k > j$, v součtu už máme jen nulové členy. Tento vztah platí pro libovolné $i, j \in \hat{n}$, čímž dostáváme n^2 rovnic pro neznámé prvky l_{ik} a u_{kj}. Těch je také n^2, jelikož z matice $\mathbb{U}$ neznáme jen to co je nad diagonálou (je horní trojúhelníková s jedničkami na diagonále), a z matice $\mathbb{L}$ neznáme diagonálu a to co je pod ní. Soustava n^2 rovnic a n^2 neznámých je tedy zřejmě řešitelná, a to dokonce jednoznačně. Ukážeme si, jak ji řešit v postupných krocích. Začněme s tím, že si zafixujeme první sloupec $j = 1$. Potom

$$a_{i1} = \sum_{k=1}^{\min(i,1)} l_{ik}u_{k1} = l_{i1}u_{11} = l_{i1}, \tag{3.47}$$

kde jsme dále využili toho, že $u_{11} = 1$ (jednička na diagonále $\mathbb{U}$). Proto $l_{i1} = a_{i1}$ pro každé $i \in \hat{n}$. Tím jsme tedy určili první sloupec matice $\mathbb{L}$. Nyní naopak zafixujeme první řádek $i = 1$. Potom

$$a_{1j} = \sum_{k=1}^{\min(1,j)} l_{1k}u_{kj} = l_{11}u_{1j}, \tag{3.48}$$

tedy dostáváme $u_{1j} = \frac{a_{1j}}{l_{11}}$ pro každé $j \in \hat{n}$, kde ale l_{11} známe. Tím jsme tedy určili první řádek matice $\mathbb{U}$. Pokračujeme dále, nyní zafixujeme druhý sloupec $j = 2$. Potom

$$a_{i2} = \sum_{k=1}^{\min(i,2)} l_{ik}u_{k2} = l_{i1}u_{12} + l_{i2}u_{22}. \tag{3.49}$$

Z tohoto vztahu známe l_{i1} (první sloupec $\mathbb{L}$ jsme určili), u_{12} (první řádek $\mathbb{U}$ jsme určili) a $u_{22} = 1$ (jednička na diagonále $\mathbb{U}$). Proto můžeme vyjádřit l_{i2} jako

$$l_{i2} = a_{i2} - l_{i1}u_{12}. \tag{3.50}$$

Tím jsme tedy určili druhý sloupec matice $\mathbb{L}$ a takto bychom pokračovali dále s druhým řádkem $\mathbb{U}$. Obecně tedy jsme-li v s-tém kroku (známe $s-1$ sloupců $\mathbb{L}$ a $s-1$ řádků $\mathbb{U}$ [TODO respektive tím, že znám řádky U znám i některé sloupce U]), zafixujeme s-tý sloupec, tzn. $j = s$ a potom

$$a_{is} = \sum_{k=1}^{\min(i,s)} l_{ik}u_{ks} = \sum_{k=1}^{s-1} l_{ik}u_{ks} + l_{is}u_{ss} = \sum_{k=1}^{s-1} l_{ik}u_{ks} + l_{is}, \tag{3.51}$$

a tedy

$$l_{is} = a_{is} - \sum_{k=1}^{s-1} l_{ik}u_{ks}. \tag{3.52}$$

Dále pro zafixovaný s-tý řádek (známe s sloupců $\mathbb{L}$ a $s-1$ řádků $\mathbb{U}$), tzn. $i = s$, máme

$$a_{sj} = \sum_{k=1}^{\min(s,j)} l_{sk}u_{kj} = \sum_{k=1}^{s-1} l_{sk}u_{kj} + l_{ss}u_{sj}, \tag{3.53}$$

a tedy

$$u_{sj} = \frac{a_{sj} - \sum_{k=1}^{s-1} l_{sk}u_{kj}}{l_{ss}}. \tag{3.54}$$

Je zřejmé, že opět lze algoritmus provádět in-place, jelikož pro získání prvků l_{is} a u_{sj} potřebujeme z matice $\mathbb{A}$ pouze a_{is}, resp. a_{sj}. V matici $\mathbb{A}$ tedy vždy postupně nahradíme sloupce od diagonálního prvku až dolů, a poté řádky od diagonálního prvku (tentokrát ne včetně) až doprava. Tím získáme matice $\mathbb{L}$ a $\mathbb{U}$. Je důležité zmínit, že algoritmus lze provést pouze pokud jsou diagonální prvky matice $\mathbb{L}$ nenulové, abychom nedělili nulou. Tato matice je ale z jednoznačnosti LU rozkladu stejná, jako matice $\mathbb{L}$ z GEM, a ta má na diagonále pivoty z GEM. Proto předpoklad silné regularity je pro obě metody stejný. Dále je vidět, že prvku matic $\mathbb{L}$ a $\mathbb{U}$ z LU rozkladu jsou spojitou funkcí prvků matice $\mathbb{A}$, což plyne přímo ze vztahů pro l_{ij} a u_{ij}.

3.2.2.1 Implementace

TODO

3.2.2.2 Analýza složitosti

Z implementace opět $O(n^3)$.

3.2.3 Choleského rozklad

Je-li matice $\mathbb{A}$ hermitovská a pozitivně definitní, pak existuje její hermitovský symetrický rozklad, tzv. Choleského:

$$\mathbb{A} = \mathbb{S}^*\mathbb{S}, \tag{3.55}$$

kde $\mathbb{S}$ je horní trojúhelníková, tzn. pokud $\mathbb{A} = \mathbb{A}^*$, pak $\mathbb{L} = \mathbb{U}^*$. Tento rozklad nám umožňuje ukládat informace o celém rozkladu jen v dolním trojúhelníkové matici $\mathbb{L}$, protože $\mathbb{U}$ pak umíme dopočítat. To odpovídá intuitivní představě, kde LU (resp. Choleského) rozklad symetrické matice v jistém smyslu zachovává symetrii. Existenci takového rozkladu je třeba dokázat, mějme tedy pozitivně definitní hermitovskou matici $\mathbb{A}$. Ze Sylvestrova kritéria víme, že všechny hlavní subdeterminanty $\mathbb{A}$ jsou kladné, tedy $\mathbb{A}$ je silně regulární. Proto existuje LDR rozklad

$$\mathbb{A} = \mathbb{L}\mathbb{D}\mathbb{R} \iff \mathbb{A} = \mathbb{A}^* = \mathbb{R}^*\mathbb{D}^*\mathbb{L}^*. \tag{3.56}$$

Z jednoznačnosti LDR rozkladu $\mathbb{L} = \mathbb{R}^*$ (TODO: někde později to rozebírám detailněji, tak sjednotit na jedno místo), proto

$$\mathbb{A} = \mathbb{R}^*\mathbb{D}^*\mathbb{R} \tag{3.57}$$

Kdyby se nám podařilo rozložti $\mathbb{D} = \tilde{\mathbb{D}}^*\tilde{\mathbb{D}}$, potom bychom mohli napsat

$$\mathbb{A} = \mathbb{R}^*\tilde{\mathbb{D}}^*\tilde{\mathbb{D}}\mathbb{R} = (\tilde{\mathbb{D}}\mathbb{R})^*(\tilde{\mathbb{D}}\mathbb{R}) = \mathbb{S}^*\mathbb{S}. \tag{3.58}$$

Z maticového násobení $\forall\, i \in \hat{n}$ platí $d_{ii} = \overline{\tilde{d}}_{ii}\tilde{d}_{ii} = |\tilde{d_{ii}}|^2$. Tento rozklad je tedy možný pouze tehdy, když jsou všechny prvky na diagonále $\mathbb{D}$ kladné. Rozepišme si LDR rozklad blokově:

$$\mathbb{A} = \begin{pmatrix} \mathbb{A}_{11} & \mathbb{A}_{12} \\ \mathbb{A}_{21} & \mathbb{A}_{22} \end{pmatrix} = \begin{pmatrix} \mathbb{L}_{11} & \mathbb{O} \\ \mathbb{L}_{21} & \mathbb{L}_{22} \end{pmatrix} \begin{pmatrix} \mathbb{D}_{11} & 0 \\ 0 & \mathbb{D}_{22} \end{pmatrix} \begin{pmatrix} \mathbb{R}_{11} & \mathbb{R}_{12} \\ 0 & \mathbb{R}_{22} \end{pmatrix}, \tag{3.59}$$

kde $\mathbb{A}_{11}, \mathbb{L}_{11}, \mathbb{D}_{11}, \mathbb{R}^{11} \in \mathbb{C}^{k,k}$. Potom

$$\mathbb{A}_{11} = \mathbb{L}_{11}\mathbb{D}_{11}\mathbb{R}_{11} \qquad \Longrightarrow \qquad \det \mathbb{A}_{11} = \det \mathbb{L}_{11} \det \mathbb{D}_{11} \det \mathbb{R}_{11} = \det \mathbb{D}_{11} = \prod_{i=1}^{k} d_{ii}, \tag{3.60}$$

protože $\det \mathbb{L}_{11} = 1$ a $\det \mathbb{R}_{11} = 1$ (jsou trojúhelníkové s jedničkami na diagonále). Z pozitivní definitnosti, resp. Sylvestrova kritéria ale $\forall\, k \in \hat{n}$ je $0 < \det \mathbb{A}_{11} = \prod_{i=1}^{k} d_{ii}$ a tedy $\forall\, i \in \hat{n}$ je $d_{ii} > 0$.

3.2.4 Otázky

- K čemu slouží a jak napočítat LU rozklad?
- K čemu slouží a jak napočítat Choleského rozklad?

3.3 Modifikace Gaussovy eliminační metody

3.3.1 Thomasův algoritmus

Mějme soustavu se silně regulární, tzv. trigiadiagonální maticí $\mathbb{A}$:

$$\begin{pmatrix} a_1 & b_1 & & & & \\ c_2 & a_2 & b_2 & & & \\ & c_3 & a_3 & b_3 & & \\ & & \ddots & \ddots & \ddots & \\ & & & c_{n-1} & a_{n-1} & b_{n-1} \\ & & & & c_n & a_n \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_{n-1} \\ x_n \end{pmatrix} = \begin{pmatrix} d_1 \\ d_2 \\ d_3 \\ \vdots \\ d_{n-1} \\ d_n \end{pmatrix}. \tag{3.61}$$

Když si představíme, jak by vypadala Gaussova eliminační metoda, zjistíme, že v každém kroku bychom provedli pouze jednu eliminaci (pod diagonálou je vždy jen jeden prvek). Také je zřejmé, že nad diagonálou s prvky b_i nemůže vzniknout žádný nenulový prvek. V rámci prvního kroku přímého chodu tedy nejdříve dostáváme

$$\begin{pmatrix} 1 & \mu_1 & & & & & \\ c_2 & a_2 & b_2 & & & & \\ & c_3 & a_3 & b_3 & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & c_{n-1} & a_{n-1} & b_{n-1} \\ & & & & c_n & a_n \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_{n-1} \\ x_n \end{pmatrix} = \begin{pmatrix} \rho_1 \\ d_2 \\ d_3 \\ \vdots \\ d_{n-1} \\ d_n \end{pmatrix}, \tag{3.62}$$

kde $\mu_1 = \frac{c_2}{a_1}$ a $\rho_1 = \frac{d_1}{a_1}$, a odečteme c_2 násobek prvního řádku od druhého.

$$\begin{pmatrix} 1 & \mu_1 & & & & & \\ 0 & a_2 - c_2\mu_1 & b_2 & & & & \\ & c_3 & a_3 & b_3 & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & c_{n-1} & a_{n-1} & b_{n-1} \\ & & & & c_n & a_n \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_{n-1} \\ x_n \end{pmatrix} = \begin{pmatrix} \rho_1 \\ d_2 - c_2\rho_1 \\ d_3 \\ \vdots \\ d_{n-1} \\ d_n \end{pmatrix}. \tag{3.63}$$

V dalším kroku bychom dělili druhý řádek pivotem $a_2 - c_2\mu_1$, a dostali

$$\begin{pmatrix} 1 & \mu_1 & & & & & \\ 0 & 1 & \frac{b_2}{a_2 - c_2\mu_1} & & & & \\ & c_3 & a_3 & b_3 & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & c_{n-1} & a_{n-1} & b_{n-1} \\ & & & & c_n & a_n \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_{n-1} \\ x_n \end{pmatrix} = \begin{pmatrix} \rho_1 \\ \frac{d_2 - c_2\rho_1}{a_2 - c_2\mu_1} \\ d_3 \\ \vdots \\ d_{n-1} \\ d_n \end{pmatrix}, \tag{3.64}$$

a tedy $\mu_2 = \frac{b}{a_2 - c_2\mu_1}$ a $\rho_2 = \frac{d_2 - c_2\rho_1}{a_2 - c_2\mu_1}$. V k-tém kroku máme na začátku soustavu tvaru

$$\begin{pmatrix} 1 & \mu_1 & & & & & & \\ & \ddots & \ddots & & & & & \\ & & 1 & \mu_{k-1} & b_k & & & \\ & & c_k & a_k & b_{k+1} & & & \\ & & & c_{k+1} & a_{k+1} & \ddots & & \\ & & & & \ddots & \ddots & b_{n-1} \\ & & & & & c_n & a_n \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_{k-1} \\ x_k \\ x_{k+1} \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} \rho_1 \\ \vdots \\ \rho_{k-1} \\ d_k \\ d_{k+1} \\ \vdots \\ d_n \end{pmatrix} \tag{3.65}$$

Odečteme c_k násobek $k-1$-tého řádku od k-tého:

$$\begin{pmatrix} 1 & \mu_1 & & & & & & \\ & \ddots & \ddots & & & & & \\ & & 1 & \mu_{k-1} & b_k & & & \\ & & & a_k - c_k\mu_{k-1} & b_{k+1} & & & \\ & & & c_{k+1} & a_{k+1} & \ddots & & \\ & & & & \ddots & \ddots & b_{n-1} \\ & & & & & c_n & a_n \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_{k-1} \\ x_k \\ x_{k+1} \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} \rho_1 \\ \vdots \\ \rho_{k-1} \\ d_k - c_k\rho_{k-1} \\ d_{k+1} \\ \vdots \\ d_n \end{pmatrix} \tag{3.66}$$

Nyní podělíme k-tý řádek pivotem $a_k - c_k\mu_{k-1}$ a dostáváme:

$$\mu_k = \frac{b_k}{a_k - c_k\mu_{k-1}}, \qquad \rho_k = \frac{d_k - c_k\rho_{k-1}}{a_k - c_k\mu_{k-1}} \tag{3.67}$$

Výsledná soustava má po doběhnutí přímého chodu tvar

$$\begin{pmatrix} 1 & \mu_1 & & & & \\ & 1 & \mu_2 & & & \\ & & 1 & \mu_3 & & \\ & & & \ddots & \ddots & \\ & & & & 1 & \mu_{n-1} \\ & & & & & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_{n-1} \\ x_n \end{pmatrix} = \begin{pmatrix} \rho_1 \\ \rho_2 \\ \rho_3 \\ \vdots \\ \rho_{n-1} \\ \rho_n \end{pmatrix}, \tag{3.68}$$

a snadno odvodíme zpětný chod:

$$\begin{aligned} x_n &= \rho_n, \\ x_{n-1} &= \rho_{n-1} - \mu_{n-1}x_n, \\ x_{n-2} &= \rho_{n-2} - \mu_{n-2}x_{n-1}, \\ &\vdots \\ x_2 &= \rho_2 - \mu_2 x_3, \\ x_1 &= \rho_1 - \mu_1 x_2. \end{aligned} \tag{3.69}$$

3.3.1.1 Význam

Když si běh algoritmu shrneme, postupně provádíme následující kroky:

1. Položíme $\mu_1 = \dfrac{b_1}{a_1},\ \rho_1 = \dfrac{d_1}{a_1}$,
2. a pro $k = 2, \ldots, n-1$ napočítáme:

$$\mu_k = \frac{b_k}{a_k - c_k\mu_{k-1}}, \quad \rho_k = \frac{d_k - c_k\rho_{k-1}}{a_k - c_k\mu_{k-1}} \tag{3.70}$$

3. Nakonec položíme $\rho_n = \dfrac{d_n - c_n\rho_{n-1}}{a_n - c_n\mu_{n-1}}$
4. Dále položíme $x_n = \rho_n$,
5. a pro $k = n-1, \ldots, 1$ napočítáme:

$$x_k = \rho_k - \mu_k x_{k+1} \tag{3.71}$$

Počet opakování kroků 2 a 5 je závislý na n, tudíž jejich složitost je $O(n)$. Zbylé kroky algoritmu jsou prováděny v konstatním čase, a tedy celková složitost Thomasova algoritmu je $O(n)$. Oproti složitosti Gaussovy eliminační metody $O(n^3)$ je to tedy výrazné zlepšení.

3.3.2 Schurův doplněk

Mějme nyní soustavu se silně regulární maticí $\mathbb{A} \in \mathbb{C}^{n,n}$, která má následující blokovou strukturu:

$$\begin{pmatrix} \mathbb{A}_1 & & & & \mathbb{C}_1 \\ & \mathbb{A}_2 & & & \mathbb{C}_2 \\ & & \ddots & & \vdots \\ & & & \mathbb{A}_{r-1} & \mathbb{C}_{r-1} \\ \mathbb{B}_1 & \mathbb{B}_2 & \ldots & \mathbb{B}_{r-1} & \mathbb{A}_r \end{pmatrix} \begin{pmatrix} \vec{x}_1 \\ \vec{x}_2 \\ \vdots \\ \vec{x}_{r-1} \\ \vec{x}_r \end{pmatrix} = \begin{pmatrix} \vec{d}_1 \\ \vec{d}_2 \\ \vdots \\ \vec{d}_{r-1} \\ \vec{d}_r \end{pmatrix}. \tag{3.72}$$

Rozměry bloků mohou být různé, ale požadujeme, aby $\mathbb{A}_i, \mathbb{C}_i, \vec{x}_i, \vec{d}_i$ měly stejný počet řádků pro $i \in (r \hat{-} 1)$, aby $\mathbb{B}_1, \mathbb{B}_{r-1}, \mathbb{A}_r$ měly stejný počet řádků, $\mathbb{B}_i, \mathbb{A}_i$ měly stejný počet sloupců pro $i \in (r \hat{-} 1)$ a $\mathbb{C}_1, ...\mathbb{C}_{r-1}, \mathbb{A}_r$ měly stejný počet sloupců. Gaussovu eliminační metodu provedeme v blokovém tvaru. Dělení pivotem v tomto případě představuje vynásobení i-tého řádku maticí $\mathbb{A}_i^{-1}$ pro $i \in r \hat{-} 1$, tzn.

$$\begin{pmatrix} \mathbb{I} & & & & \mathbb{A}_1^{-1}\mathbb{C}_1 \\ & \mathbb{I} & & & \mathbb{A}_2^{-1}\mathbb{C}_2 \\ & & \ddots & & \vdots \\ & & & \mathbb{I} & \mathbb{A}_{r-1}^{-1}\mathbb{C}_{r-1} \\ \mathbb{B}_1 & \mathbb{B}_2 & \dots & \mathbb{B}_{r-1} & \mathbb{A}_r \end{pmatrix} \begin{pmatrix} \vec{x}_1 \\ \vec{x}_2 \\ \vdots \\ \vec{x}_{r-1} \\ \vec{x}_r \end{pmatrix} = \begin{pmatrix} \mathbb{A}_1^{-1}\vec{d}_1 \\ \mathbb{A}_2^{-1}\vec{d}_2 \\ \vdots \\ \mathbb{A}_{r-1}^{-1}\vec{d}_{r-1} \\ \vec{d}_r \end{pmatrix}, \tag{3.73}$$

a od posledního řádku postupně odečteme B_i násobek i-tého řádku:

$$\begin{pmatrix} \mathbb{I} & & & \mathbb{A}_1^{-1}\mathbb{C}_1 \\ & \mathbb{I} & & \mathbb{A}_2^{-1}\mathbb{C}_2 \\ & & \ddots & \vdots \\ & & \mathbb{I} & \mathbb{A}_{r-1}^{-1}\mathbb{C}_{r-1} \\ & & & \mathbb{S} \end{pmatrix} \begin{pmatrix} \vec{x}_1 \\ \vec{x}_2 \\ \vdots \\ \vec{x}_{r-1} \\ \vec{x}_r \end{pmatrix} = \begin{pmatrix} \mathbb{A}_1^{-1}\vec{d}_1 \\ \mathbb{A}_2^{-1}\vec{d}_2 \\ \vdots \\ \mathbb{A}_{r-1}^{-1}\vec{d}_{r-1} \\ \vec{s} \end{pmatrix}, \tag{3.74}$$

kde

$$\begin{aligned} \mathbb{S} &= \mathbb{A}_r - \mathbb{B}_1\mathbb{A}_1^{-1}\mathbb{C}_1 - \mathbb{B}_2\mathbb{A}_2^{-1}\mathbb{C}_2 - \ldots - \mathbb{B}_{r-1}\mathbb{A}_{r-1}^{-1}\mathbb{C}_{r-1}, \\ \vec{s} &= \vec{d}_r - \mathbb{B}_1\mathbb{A}_1^{-1}\vec{d}_1 - \mathbb{B}_2\mathbb{A}_2^{-1}\vec{d}_2 - \ldots - \mathbb{B}_{r-1}\mathbb{A}_{r-1}^{-1}\vec{d}_{r-1}. \end{aligned} \tag{3.75}$$

Právě matice $\mathbb{S}$ se nazývá Schurův doplněk matice $\mathbb{A}$. Snadno vidíme, že pak řešení soustavy lze získat jako:

$$\begin{aligned} \vec{x}_r &= \mathbb{S}^{-1}\vec{s}, \\ \vec{x}_{r-1} &= \mathbb{A}_{r-1}^{-1}\vec{d}_{r-1} - \mathbb{A}_{r-1}^{-1}\mathbb{C}_{r-1}\vec{x}_r, \\ &\vdots \\ \vec{x}_1 &= \mathbb{A}_1^{-1}\vec{d}_1 - \mathbb{A}_1^{-1}\mathbb{C}_1\vec{x}_r. \end{aligned} \tag{3.76}$$

Protože je v blokovém tvaru výsledné matice nenulový pouze poslední sloupce, v každém kroku zpětného chodu využijeme pouze vektor $\vec{x}_r$, není třeba znát předchozí vektory $\vec{x}_i$ pro $i < r$.

3.3.2.1 Význam

V tom je velký význam Schurova doplňku – každý krok zpětného chodu lze řešit paralelně, stejně tak hledání inverzí $\mathbb{A}_i^{-1}$ na začátku metody lze provádět paralelně.

3.3.2.2 Speciální případ pro $r = 2$

V soustavě s maticí, pro kterou nemáme žádnou efektivní metodu řešení, můžeme např. položit $r = 2$. Dostáváme soustavu s maticí tvaru

$$\begin{pmatrix} \mathbb{A}_1 & \mathbb{C}_1 \\ \mathbb{B}_1 & \mathbb{A}_2 \end{pmatrix} \begin{pmatrix} \vec{x}_1 \\ \vec{x}_2 \end{pmatrix} = \begin{pmatrix} \vec{d}_1 \\ \vec{d}_2 \end{pmatrix}. \tag{3.77}$$

Nyní pokud je nově vzniklá $\mathbb{A}_1$ např. tridiagonální nebo symetrické nebo pozitivně definitní, můžeme k výpočtu inverze použít nějakou efektivnější metodu a tuto "část"původní úlohy tak vyřešit efektivněji.

3.3.2.3 GEM a blokový tvar

Z výše popsaného postupu je také vidět, že jak GEM, tak např. Thomasův algoritmus lze provádět také v blokovém tvaru.

3.3.3 Otázky

- Řešení soustav s tridiagonální maticí.
- Co je to Schurův doplněk?
- Blokové modifikace maticových algoritmů

Kapitola 4

Iterativní metody řešení soustav lineárních rovnic

V této kapitole se budeme zabývat iterativími metodami pro řešení úlohy $\mathbb{A}\vec{x} = \vec{b}$ s regulární maticí $\mathbb{A}$. Cílem je nalézt metody, které mohou být pro určité typy úloh efektivnější než Gaussova eliminační metoda.

4.1 Základní pojmy

Základní myšlenkou iteračních metod je generování posloupnosti vektorů $\{\vec{x}^{(k)}\}_{k=1}^{\infty}$, která konverguje k přesnému řešení soustavy $\vec{x}^*$, tedy

$$\lim_{k\to\infty} \vec{x}^{(k)} = \vec{x}^*. \tag{4.1}$$

V praxi tedy obecně nikdy nezískáme přesné řešení, ale můžeme se k němu přiblížit s téměř libovolnou přesností.

Definice 4.1 (Obecná iterační metoda)**.** Nechť je dána počáteční aproximace $\vec{x}^{(0)}$. Obecná iterační metoda generuje další členy posloupnosti pomocí předpisu

$$\vec{x}^{(k+1)} = \mathbb{B}^{(k)}\vec{x}^{(k)} + \vec{c}^{(k)}, \tag{4.2}$$

kde matice $\mathbb{B}^{(k)}$ a vektor $\vec{c}^{(k)}$ závisí na použité metodě. Dále požadujeme, aby pro přesné řešení $\vec{x}^*$ platilo

$$\vec{x}^* = \mathbb{B}^{(k)}\vec{x}^* + \vec{c}^{(k)}. \tag{4.3}$$

Tato podmínka se nazývá podmínka konzistence.

Věta 4.2. Iterační metoda tvaru $\vec{x}^{(k+1)} = \mathbb{B}^{(k)}\vec{x}^{(k)} + \vec{c}^{(k)}$ splňující podmínku konzistence konverguje k $\vec{x}^*$ pro libovolnou počáteční volbu $\vec{x}^{(0)}$ právě tehdy, když platí

$$\lim_{k\to\infty} \mathbb{B}^{(k)}\mathbb{B}^{(k-1)}\dots\mathbb{B}^{(0)} = \mathbb{O}. \tag{4.4}$$

Důkaz. Mějme iterační metodu tvaru $\vec{x}^{(k+1)} = \mathbb{B}^{(k)}\vec{x}^{(k)} + \vec{c}^{(k)}$ splňující podmínku konzistence $\vec{x}^* = \mathbb{B}^{(k)}\vec{x}^* + \vec{c}^{(k)}$. Potom

$$\begin{aligned}\vec{x}^{(k)} - \vec{x}^* = \mathbb{B}^{(k-1)}\vec{x}^{(k-1)} + \vec{c}^{(k-1)} - \mathbb{B}^{(k-1)}\vec{x}^* - \vec{c}^{(k-1)} &= \mathbb{B}^{(k-1)}(\vec{x}^{(k-1)} - \vec{x}^*)\\ &= \mathbb{B}^{(k-1)}\mathbb{B}^{(k-2)}(\vec{x}^{(k-2)} - \vec{x}^*)\\ &= \mathbb{B}^{(k-1)}\mathbb{B}^{(k-2)}\dots\mathbb{B}^{(0)}(\vec{x}^{(0)} - \vec{x}^*).\end{aligned} \tag{4.5}$$

Rozdíl $\vec{x}^{(0)} - \vec{x}^*$ je pevně daný, a tudíž pro $k \to \infty$

$$\vec{x}^{(k)} - \vec{x}^* \to \vec{0} \iff \mathbb{B}^{(k)}\mathbb{B}^{(k-1)} \dots \mathbb{B}^{(0)} \to \mathbb{O}. \tag{4.6}$$

□

Poznámka 4.3 (Samoopravující vlastnost)**.** Iterační metody mají takzvanou samoopravující vlastnost. Pokud během výpočtu dojde k chybě (například vlivem zaokrouhlení), lze na výsledek pohlížet jako na novou počáteční aproximaci $\vec{x}^{(0)}$. Díky konvergenci pro libovolný počáteční vektor se tyto chyby v dalších iteracích eliminují, nikoliv kumulují. Z tohoto důvodu není nutné provádět detailní analýzu citlivosti na výpočetní chyby.

Definice 4.4 (Stacionární iterační metoda)**.** Iterační metody dělíme na:

- **stacionární**, pro které jsou matice $\mathbb{B}^{(k)}$ a vektor $\vec{c}^{(k)}$ konstantní, tj. $\mathbb{B}^{(k)} = \mathbb{B}$ a $\vec{c}^{(k)} = \vec{c}$ pro všechna k. Předpis má tedy tvar
$$\vec{x}^{(k+1)} = \mathbb{B}\vec{x}^{(k)} + \vec{c}. \tag{4.7}$$
- **nestacionární**, kde $\mathbb{B}^{(k)}$ a $\vec{c}^{(k)}$ jsou různé pro různá k.

Poznámka 4.5. Stacionární metody jsou jednodušší na implementaci i na teoretickou analýzu. V dalším textu se budeme zabývat výhradně stacionárními metodami.

Definice 4.6 (Pevný bod)**.** Protože ve stacionární iterační metodě jsou matice $\mathbb{B}$ a vektor $\vec{c}$ konstantní, můžeme působení matice $\mathbb{B}$ a vektoru $\vec{c}$ v k-té iteraci zapsat pomocí vektorového zobrazení $\vec{\varphi}$, tzn. $\vec{x}^{(k+1)} = \vec{\varphi}(\vec{x}^{(k)})$. Pokud existuje $\vec{x}^*$ takové, že $\vec{\varphi}(\vec{x}^*) = \vec{x}^*$, říkáme, že $\vec{x}^*$ je pevný bod zobrazení $\vec{\varphi}$.

Věta 4.7. Stacionární iterační metoda $\vec{x}^{(k+1)} = \mathbb{B}\vec{x}^{(k)} + \vec{c}$ splňující podmínku konzistence konverguje k $\vec{x}^*$ pro libovolné $\vec{x}^{(0)}$ právě tehdy, když platí

$$\lim_{k\to\infty} \mathbb{B}^k = \mathbb{O}. \tag{4.8}$$

Důkaz. Tvrzení plyne z věty (4.2), neboť pro stacionární metodu platí $\mathbb{B}^{(k)}\mathbb{B}^{(k-1)} \dots \mathbb{B}^{(0)} = \mathbb{B}^k$ (pro každé k je matice $\mathbb{B}^{(k)}$ stejná) a tedy

$$\mathbb{O} = \lim_{k\to\infty} \mathbb{B}^{(k)}\mathbb{B}^{(k-1)} \dots \mathbb{B}^{(0)} = \lim_{k\to\infty} \mathbb{B}^k \tag{4.9}$$

□

Věta 4.8 (Konvergence stacionární metody)**.** Stacionární iterační metoda $\vec{x}^{(k+1)} = \mathbb{B}\vec{x}^{(k)} + \vec{c}$ konverguje k $\vec{x}^*$ pro libovolné $\vec{x}^{(0)}$ právě tehdy, když pro spektrální poloměr iterační matice $\mathbb{B}$ platí

$$\rho(\mathbb{B}) < 1. \tag{4.10}$$

Důkaz. Tvrzení plyne z podmínek pro konvergenci maticové posloupnosti $\mathbb{B}^k$, viz věta (1.93). □

Věta 4.9 (Postačující podmínka konvergence)**.** Existuje-li pro stacionární iterační metodu $\vec{x}^{(k+1)} = \mathbb{B}\vec{x}^{(k)} + \vec{c}$ maticová norma $||\cdot||$ taková, že

$$||\mathbb{B}|| < 1, \tag{4.11}$$

potom stacionární iterační metoda konverguje k $\vec{x}^*$ pro libovolné $\vec{x}^{(0)}$.

Důkaz. Tvrzení plyne z věty (1.95). □

4.1.1 Složitost iteračních metod

Poznámka 4.10. Máme-li stacionární iterační metodu $\vec{x}^{(k+1)} = \mathbb{B}\vec{x}^{(k)} + \vec{c}$, kde $\mathbb{B} \in \mathbb{C}^{n,n}$ a $\vec{c} \in \mathbb{C}^n$, potom složitost maticového násobení je $O(n^2)$ a přičítání vektoru $O(n)$. Složitost jedné iterace je tedy $O(n^2 + n) = O(n^2)$.

Pro počet iterací $k << n$ je metoda efektivnější než GEM. Pokud bychom se s počtem iterací k přiblížili k n, pak se složitost bude blížti k $O(n^3)$, což už je stejně jako GEM.

4.1.2 Odhady chyb a ukončovací kritéria

Při praktickém použití iteračních metod je klíčové vědět, kdy výpočet ukončit. K tomu slouží odhady chyb.

Definice 4.11 (Reziduum)**.** Pro k-tou aproximaci řešení $\vec{x}^{(k)}$ definujeme reziduální vektor (reziduum) jako

$$\vec{r}^{(k)} = \mathbb{A}\vec{x}^{(k)} - \vec{b}. \tag{4.12}$$

Reziduum tedy udává, jak dobře aproximace $\vec{x}^{(k)}$ splňuje původní rovnici.

Věta 4.12 (Aposteriorní odhady chyby)**.** Nechť stacionární iterační metoda konverguje. Pak pro chybu k-té aproximace platí následující odhady:

1. $||\vec{x}^{(k)} - \vec{x}^*|| \leq ||\mathbb{A}^{-1}|| \cdot ||\mathbb{A}\vec{x}^{(k)} - \vec{b}|| = ||\mathbb{A}^{-1}|| \cdot ||\vec{r}^{(k)}||$
2. $||\vec{x}^{(k-1)} - \vec{x}^*|| \leq ||(\mathbb{I} - \mathbb{B})^{-1}|| \cdot ||\vec{x}^{(k)} - \vec{x}^{(k-1)}||$

při použití souhlasné maticové a vektorové normy.

Důkaz 1. V celé této kapitole předpokládáme, že řešíme soustavy rovnic $\mathbb{A}\vec{x} = \vec{b}$ s regulární maticí $\mathbb{A}$. Proto existuje inverzní matice $\mathbb{A}^{-1}$, a tedy můžeme psát:

$$\vec{x}^{(k)} - \vec{x}^* = \mathbb{A}^{-1}(\mathbb{A}\vec{x}^{(k)} - \mathbb{A}\vec{x}^*) = \mathbb{A}^{-1}(\mathbb{A}\vec{x}^{(k)} - \vec{b}) = \mathbb{A}^{-1}\vec{r}^{(k)}. \tag{4.13}$$

Odtud už jednoduše

$$||\vec{x}^{(k)} - \vec{x}^*|| \leq ||\mathbb{A}^{-1}|| \cdot ||\vec{r}^{(k)}||. \tag{4.14}$$

□

Důkaz 2. Vyjdeme z toho, že $\vec{x}^* = \mathbb{B}\vec{x}^* + \vec{c} \iff (\mathbb{I} - \mathbb{B})\vec{x}^* = \vec{c}$. My bychom ale chtěli také říct, že $\vec{x}^* = (\mathbb{I} - \mathbb{B})^{-1}\vec{c}$. Předtím musíme ovšem ukázat, že matice $\mathbb{I} - \mathbb{B}$ je regulární, tedy že existuje její inverzní matice $(\mathbb{I} - \mathbb{B})^{-1}$. Z Schurovy věty (1.59) víme, že $\mathbb{B} = \mathbb{U}^*\mathbb{R}\mathbb{U}$ a tedy také $\mathbb{I} - \mathbb{B} = \mathbb{U}^*(\mathbb{I} - \mathbb{R})\mathbb{U}$. Dále také z (4.8) víme, že $\vec{x}^{(k)}$ konverguje k $\vec{x}^*$ právě tehdy, když $\rho(\mathbb{B}) < 1$. Tato poslední nerovnost ale implikuje, že $|r_{ii}| < 1$ (TODO: dohledat proč), a tedy $\mathbb{I} - \mathbb{R}$ je regulární matice, protože její diagonální prvky jsou nenulové. Tím pádem je regulární i $\mathbb{I} - \mathbb{B}$, a tedy existuje její inverzní matice $(\mathbb{I} - \mathbb{B})^{-1}$. Nyní se vraťme k tomu, co jsme chtěli dokázat, a využijeme dokázaný vztah pro $\vec{x}^*$.

$$\begin{aligned} ||\vec{x}^{(k)} - \vec{x}^*|| &= ||\vec{x}^{(k)} - (\mathbb{I} - \mathbb{B})^{-1}\vec{c}|| = ||(\mathbb{I} - \mathbb{B})^{-1}\left[(\mathbb{I} - \mathbb{B})\vec{x}^{(k)} - \vec{c}\right]|| \\ &= ||(\mathbb{I} - \mathbb{B})^{-1}\left[\vec{x}^{(k)} - \mathbb{B}\vec{x}^{(k)} - \vec{c}\right]|| = ||(\mathbb{I} - \mathbb{B})^{-1}\left[\vec{x}^{(k)} - \vec{x}^{(k+1)}\right]|| \\ &\leq ||(\mathbb{I} - \mathbb{B})^{-1}|| \cdot ||\vec{x}^{(k)} - \vec{x}^{(k+1)}||. \end{aligned} \tag{4.15}$$

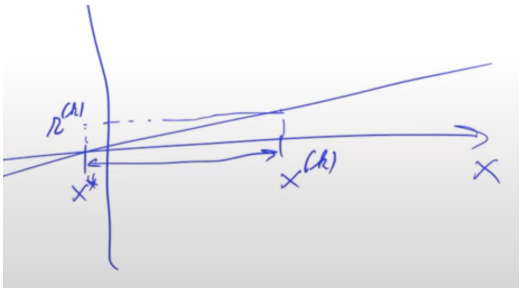
Zbývá ještě okomentovat to, že nám výsledek vyšel pro $\vec{x}^{(k+1)}$, což neodpovídá původnímu tvrzení. To ale nevadí, protože stačí na začátku důkazu posunout $k \leftarrow k - 1$. □

Poznámka 4.13 (Ukončovací kritéria)**.** Aposteriorní odhady chyb jsou základem pro praktická ukončovací kritéria. Výpočet typicky zastavíme, pokud je norma rezidua $||\vec{r}^{(k)}||$ (odhad 1) nebo norma rozdílu dvou po sobě jdoucích iterací $||\vec{x}^{(k)} - \vec{x}^{(k-1)}||$ (odhady 2 a 3) dostatečně malá. Často se používá relativní kritérium vztažené k „řádům úlohy“, např. normě matice $\mathbb{A}$ nebo pravé strany $\vec{b}$:

$$\frac{||\vec{r}^{(k)}||}{||\vec{b}||} < \epsilon \tag{4.16}$$

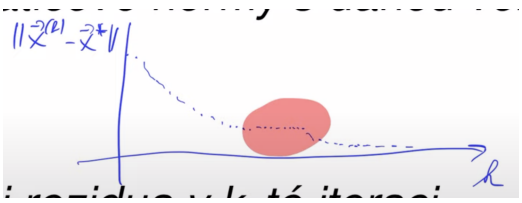
pro zadanou toleranci $\epsilon > 0$.

Poznámka 4.14. Malé reziduum však nemusí nutně znamenat, že jsme našli dobré řešení. Představme si, že jsme v realných čísel, a mějme soustavu $f(x) = ax + b$. Potom pro malé a je tvar f: tzn. tvar f je téměř



Obrázek 4.1: graf f s malým a

vodorovný, a proto i přes malé $r^{(k)}$ je rozdíl mezi $x^{(k)}$ a x^* velmi velký. Odhad nicméně stále funguje, protože se v něm nachází matice $\mathbb{A}^{-1}$, která v tomto případě odpovídá převrácené hodnotě malého a, tedy velkému číslu. TODO: Vyjasnit, co to znamená, že soustava je špatně podmíněná Stejně tak i malý rozdíl mezi dvěma po sobě jdoucími iteracemi nemusí znamenat, že jsme blízko k řešení. Může totiž nastat situace, kdy se konvergence na chvíli zpomalí, ale stále jsme daleko od řešení, protože se opět rozjede později. To se nám u stacionárních metod



Obrázek 4.2: znázornění vzdálenosti dvou po sobě jdoucích iterací v závislosti na k v případě, že se konvergence na chvíli zpomalí

spíše nestane, nicméně je třeba i tento případ mít na paměti.

Poznámka 4.15. Aposteriorní odhady chyby jsou výpočty, které můžeme udělat až po tom, co provedeme nějaké iterace. Naopak apriorní odhady, které si ukážeme dále, provádíme před samotným výpočtem.

Věta 4.16 (Apriorní odhad chyby)**.** Nechť pro iterační matici $\mathbb{B}$ platí $||\mathbb{B}|| < 1$ v nějaké maticové normě souhlasné s danou vektorovou normou. Pak pro chybu k-té iterace platí

$$||\vec{x}^{(k)} - \vec{x}^*|| \leq ||\mathbb{B}||^k \left(||\vec{x}^{(0)}|| + \frac{||\vec{c}||}{1 - ||\mathbb{B}||} \right). \tag{4.17}$$

Důkaz. Využijeme toho, že jsme si již v důkazu (4.12) ukázali, že lze napsat $\vec{x}^* = (\mathbb{I} - \mathbb{B})^{-1}\vec{c}$. Dále napíšeme

$$\vec{x}^{(k)} = \mathbb{B}\vec{x}^{(k-1)} + \vec{c} = \mathbb{B}(\mathbb{B}\vec{x}^{(k-2)} + \vec{c}) + \vec{c} = \cdots = \mathbb{B}^k\vec{x}^{(0)} + \sum_{i=0}^{k-1} \mathbb{B}^i\vec{c}. \tag{4.18}$$

Připomeňme ještě, že pokud $\rho(\mathbb{B}) < 1$ (což máme splněno, protože $||\mathbb{B}|| < 1$), potom

$$\sum_{i=0}^{\infty} \mathbb{B}^i = (\mathbb{I} - \mathbb{B})^{-1}. \tag{4.19}$$

Nyní vše jen dáme dohromoady:

$$\begin{aligned} ||\vec{x}^{(k)} - \vec{x}^*|| &= ||\mathbb{B}^k\vec{x}^{(0)} + \sum_{i=0}^{k-1}\mathbb{B}^i\vec{c} - (\mathbb{I}-\mathbb{B})^{-1}\vec{c}|| = ||\mathbb{B}^k\vec{x}^{(0)} + \sum_{i=0}^{k-1}\mathbb{B}^i\vec{c} - \sum_{i=0}^{\infty}\mathbb{B}^i\vec{c}|| = ||\mathbb{B}^k\vec{x}^{(0)} + \sum_{i=k}^{\infty}\mathbb{B}^i\vec{c}|| \\ &= ||\mathbb{B}^k\vec{x}^{(0)} + \mathbb{B}^k\sum_{i=0}^{\infty}\mathbb{B}^i\vec{c}|| \leq ||\mathbb{B}||^k \cdot ||\vec{x}^{(0)} - \sum_{i=0}^{\infty}\mathbb{B}^i\vec{c}|| \leq ||\mathbb{B}||^k \cdot \left[||\vec{x}^0|| + \sum_{i=0}^{+\infty}||\mathbb{B}||^i||\vec{c}||\right] \\ &\leq ||\mathbb{B}||^k \cdot \left[||\vec{x}^{(0)}|| + \frac{||\vec{c}||}{1-||\mathbb{B}||}\right]. \end{aligned} \tag{4.20}$$

□

Poznámka 4.17. Apriorní odhad chyby není moc užitečný pro praktické počítání. Jeho význam spočívá v tom, že nám umožňuje určit, jak rychle se bude posloupnost $\{\vec{x}^{(k)}\}$ blížit k řešení $\vec{x}^*$. Pokud je $||\mathbb{B}||$ blízké 1, pak se posloupnost $\{\vec{x}^{(k)}\}$ bude blížit k řešení velmi pomalu, a tedy i počet iterací bude velký. Naopak pokud je $||\mathbb{B}||$ blízké 0, pak se posloupnost $\{\vec{x}^{(k)}\}$ bude blížit k řešení velmi rychle, a tedy i počet iterací bude malý. Říká nám tedy, že naším cílem by mělo být volit iterační matici $\mathbb{B}$ tak, aby její norma byla co nejmenší.

Nyní již víme, za jakých podmínek stacionární metody konvergují a jak odhadovat jejich chybu. Další otázkou je, jak konkrétně volit iterační matici $\mathbb{B}$ a vektor $\vec{c}$ tak, aby byla splněna podmínka konzistence pro řešení soustavy $\mathbb{A}\vec{x} = \vec{b}$.

4.1.3 Metoda postupných aproximací

Nejjednodušší způsob, jak navrhnout iterační metodu, je pokusit se v každém kroku opravit stávající aproximaci $\vec{x}^{(k)}$ o chybu, kterou generuje. Jediný snadno dostupný ukazatel této chyby je reziduum $\vec{r}^{(k)}$. Zkusme tedy definovat metodu jednoduchou korekcí

$$\vec{x}^{(k+1)} = \vec{x}^{(k)} - \vec{r}^{(k)} = \vec{x}^{(k)} - (\mathbb{A}\vec{x}^{(k)} - \vec{b}) = (\mathbb{I}-\mathbb{A})\vec{x}^{(k)} + \vec{b}. \tag{4.21}$$

Definice 4.18 (Metoda postupných aproximací)**.** Metodu s iterační maticí $\mathbb{B} = \mathbb{I} - \mathbb{A}$ a vektorem $\vec{c} = \vec{b}$ nazýváme metodou postupných aproximací. Její iterační předpis je

$$\vec{x}^{(k+1)} = (\mathbb{I}-\mathbb{A})\vec{x}^{(k)} + \vec{b}. \tag{4.22}$$

Poznámka 4.19. Podmínka konzistence je pro tuto metodu zřejmě splněna:

$$\vec{x}^* = (\mathbb{I}-\mathbb{A})\vec{x}^* + \vec{b} \iff \vec{x}^* = \vec{x}^* - \mathbb{A}\vec{x}^* + \vec{b} \iff \mathbb{A}\vec{x}^* = \vec{b}. \tag{4.23}$$

Poznámka 4.20. TODO popsat proč je metoda výhodná pro řídké matice, že můžu iterovat jen přes nenulové prvky

Věta 4.21 (Konvergence metody postupných aproximací)**.** Metoda postupných aproximací, daná vztahem $\vec{x}^{(k+1)} = (\mathbb{I}-\mathbb{A})\vec{x}^{(k)} + \vec{b}$, konverguje pro libovolnou počáteční aproximaci $\vec{x}^{(0)}$ k řešení soustavy $\mathbb{A}\vec{x} = \vec{b}$ právě tehdy, když

$$\rho(\mathbb{I}-\mathbb{A}) < 1. \tag{4.24}$$

Postačující podmínkou pro konvergenci je, aby pro nějakou maticovou normu $||\cdot||$ platilo

$$||\mathbb{I}-\mathbb{A}|| < 1. \tag{4.25}$$

Důkaz. Tvrzení plyne přímo z obecných podmínek pro konvergenci stacionárních metod, kde za iterační matici

$\mathbb{B}$ dosadíme $\mathbb{I} - \mathbb{A}$, viz (4.8) a (4.9). □

Lemma 4.22 (Spektrální zobrazení pro polynomy)**.** Nechť $\mathbb{A} \in \mathbb{C}^{n,n}$ a $\lambda \in \sigma(\mathbb{A})$. Nechť $p(x)$ je polynom proměnné x. Potom platí, že $p(\lambda) \in \sigma(p(\mathbb{A}))$.

Důkaz. Nechť $\lambda \in \sigma(\mathbb{A})$ a $\vec{x}$ příslušný vlastní vektor. Potom

$$\mathbb{A}^k\vec{x} = \mathbb{A}^{k-1}(\lambda\vec{x}) = \lambda^k\vec{x}. \tag{4.26}$$

Nyní zapíšeme polynom $p(x)$ jako $p(x) = \sum_{k=0}^{n} a_k x^k$. Potom

$$p(\mathbb{A})\vec{x} = \sum_{k=0}^{n} a_k\mathbb{A}^k\vec{x} = \sum_{k=0}^{n} a_k\lambda^k\vec{x} = p(\lambda)\vec{x}. \tag{4.27}$$

Proto $p(\lambda) \in \sigma(p(\mathbb{A}))$. □

Věta 4.23. Nechť je matice $\mathbb{A}$ hermitovská. Potom metoda postupných aproximací konverguje právě tehdy, když platí

$$2\mathbb{I} > \mathbb{A} > \mathbb{O}. \tag{4.28}$$

Důkaz. Nechť $\mathbb{A}$ je hermitovská matice. Potom její spektrum je reálné ($\sigma(\mathbb{A}) \subset \mathbb{R}$). Víme, že metoda postupných aproximací konverguje právě tehdy, když platí $\rho(\mathbb{I} - \mathbb{A}) < 1$. Protože spektrum $\mathbb{A}$ je realné, nutně $\sigma(\mathbb{I} - \mathbb{A}) \subset (-1, 1)$ (TODO: přepsat tenhle důkaz lépe pomocí předchozího lemma, tento krok je trochu nadbytečný). Definujeme polynom $p(x) = 1 - x$ (protože pracujeme s maticí $p(\mathbb{A}) = \mathbb{I} - \mathbb{A}$). Předchozí lemma díky $\lambda \in \sigma(\mathbb{A})$ implikuje, že $p(\lambda) = 1 - \lambda \in \sigma(\mathbb{I} - \mathbb{A})$ a tedy $p(\lambda) \in (-1, 1)$. Potom ale $\lambda \in (0, 2)$, resp. $2 > \lambda > 0$. Z tohoto vidíme, že všechna vlastní čísla matice $\mathbb{A}$ jsou kladná, tzn. $\mathbb{A} > \mathbb{O}$ (je PD). Dále $2 > \lambda \iff 2 - \lambda > 0$. Pokud si zadefinujeme polynom $q(x) = 2 - x$, tak opět z předchozího lemmatu $0 < 2 - \lambda = q(\lambda) \in \sigma(q(\mathbb{A}))$. To ale znamená, že matice $2\mathbb{I} - \mathbb{A} > 0$, tedy $2\mathbb{I} > \mathbb{A}$. □

Poznámka 4.24. Podmínka $2\mathbb{I} > \mathbb{A} > \mathbb{O}$ je poměrně silná a v praxi ji splňuje jen málo matic, protože vyžaduje, aby $\sigma(\mathbb{A}) \subset (0, 2)$. Z tohoto důvodu metoda postupných aproximací ve své základní podobě často nekonverguje nebo konverguje velmi pomalu.

4.1.4 Předpodmínění

Abychom zlepšili konvergenční vlastnosti metody postupných aproximací (a iteračních metod obecně), používá se technika zvaná **předpodmínění**. Myšlenka spočívá v nahrazení původní soustavy $\mathbb{A}\vec{x} = \vec{b}$ ekvivalentní soustavou, která bude mít „lepší“ vlastnosti. Tuto novou soustavu získáme vynásobením původní rovnice zleva vhodnou regulární maticí $\mathbb{H}$, kterou nazýváme **předpodmiňovač** nebo **předpodmiňovací matice**.

$$\mathbb{H}\mathbb{A}\vec{x} = \mathbb{H}\vec{b}. \tag{4.29}$$

Protože je matice $\mathbb{H}$ regulární, řešení této nové soustavy je shodné s řešením té původní. Např. pro metodu postupných aproximací víme, že konverguje na intervalu (0,2), pokud tedy zvolíme $\mathbb{H}$, která „smrskne“ vlastní čísla matice $\mathbb{A}$ do tohoto intervalu, získáme hledané „lepší“ vlastnosti. Obecně tedy budeme chápat přepodmínění jako „umravnění“ spektra. (TODO rozepsat, že se může stát, že HA bude singulární když ho špatně zvolíme - zcukne nějaké vlastní číslo na nulu) Nyní aplikujeme myšlenku metody postupných aproximací na tuto novou, předpodmíněnou soustavu. Reziduum pro tuto soustavu je $\vec{r}^{(k)} = \mathbb{H}\mathbb{A}\vec{x}^{(k)} - \mathbb{H}\vec{b} = \mathbb{H}(\mathbb{A}\vec{x}^{(k)} - \vec{b})$a iterační krok je tedy

$$\vec{x}^{(k+1)} = \vec{x}^{(k)} - \vec{r}^{(k)} = \vec{x}^{(k)} - \mathbb{H}(\mathbb{A}\vec{x}^{(k)} - \vec{b}) = (\mathbb{I} - \mathbb{H}\mathbb{A})\vec{x}^{(k)} + \mathbb{H}\vec{b}. \tag{4.30}$$

Věta 4.25 (Konvergence předpodmíněné metody)**.** Předpodmíněná metoda postupných aproximací konverguje pro libovolné $\vec{x}^{(0)}$ k řešení soustavy $\mathbb{A}\vec{x} = \vec{b}$ právě tehdy, když

$$\rho(\mathbb{I} - \mathbb{H}\mathbb{A}) < 1. \tag{4.31}$$

Postačující podmínkou konvergence je, aby pro nějakou maticovou normu platilo $||\mathbb{I} - \mathbb{H}\mathbb{A}|| < 1$.

Důkaz. Tvrzení plyne z obecných podmínek pro konvergenci stacionárních metod, viz (4.21). □

Věta 4.26. Nechť je matice $\mathbb{A}$ hermitovská. Potom předpodmíněná metoda postupných aproximací konverguje, právě tehdy když

$$\mathbb{W} + \mathbb{W}^* > \mathbb{A} > \mathbb{O}, \tag{4.32}$$

kde $\mathbb{W} = \mathbb{H}^{-1}$. Konvergence je navíc monotónní vzhledem k energetické vektorové normě $||\cdot||_{\mathbb{A}}$.

Důkaz. Pro $\mathbb{A}$ hermitovskou pozitivně definitní, jsme definovali energetickou normu, která na matici $\mathbb{B}$ působí vztahem

$$||\mathbb{B}||_{\mathbb{A}} = ||\mathbb{A}^{\frac{1}{2}}\mathbb{B}\mathbb{A}^{-\frac{1}{2}}||_2. \tag{4.33}$$

Dokážeme, že je splněna podmínka pro konvergenci stacionární metody, tedy že $||\mathbb{B}||_{\mathbb{A}} < 1$, přičemž v našem případě je iterační matice $\mathbb{B} = \mathbb{I} - \mathbb{H}\mathbb{A}$.

$$||\mathbb{B}||_{\mathbb{A}} = ||\mathbb{A}^{\frac{1}{2}}(\mathbb{I} - \mathbb{H}\mathbb{A})\mathbb{A}^{-\frac{1}{2}}||_2 = ||\mathbb{I} - \mathbb{A}^{\frac{1}{2}}\mathbb{H}\mathbb{A}^{\frac{1}{2}}||_2 = ||\hat{\mathbb{B}}||_2, \tag{4.34}$$

kde jsme označili $\hat{\mathbb{B}} \coloneqq \mathbb{I} - \mathbb{A}^{\frac{1}{2}}\mathbb{H}\mathbb{A}^{\frac{1}{2}}$. Z (TODO KDE) víme, že

$$||\hat{\mathbb{B}}||_2 = \rho(\hat{\mathbb{B}}^*\hat{\mathbb{B}})^{\frac{1}{2}}. \tag{4.35}$$

Protože součin $\hat{\mathbb{B}}^*\hat{\mathbb{B}}$ je hermitovská pozitivně definitní matice (TODO doplnit odkaz na odvození v důkazu přepisu dvojkové normy), má kladná všechna vlastní čísla. Nyní tedy dokážeme, že jsou všechna vlastní čísla matice $\hat{\mathbb{B}}$ menší než 1, tedy že $\rho(\hat{\mathbb{B}}) < 1$, což už dává původní tvrzení.

$$\begin{aligned}\hat{\mathbb{B}}^*\hat{\mathbb{B}} = (\mathbb{I} - \mathbb{A}^{\frac{1}{2}}\mathbb{H}\mathbb{A}^{\frac{1}{2}})^*(\mathbb{I} - \mathbb{A}^{\frac{1}{2}}\mathbb{H}\mathbb{A}^{\frac{1}{2}}) &= (\mathbb{I} - \mathbb{A}^{\frac{1}{2}}\mathbb{H}^*\mathbb{A}^{\frac{1}{2}})(\mathbb{I} - \mathbb{A}^{\frac{1}{2}}\mathbb{H}\mathbb{A}^{\frac{1}{2}}) \\ &= \mathbb{I} - \mathbb{A}^{\frac{1}{2}}(\mathbb{H} + \mathbb{H}^*)\mathbb{A}^{\frac{1}{2}} + \mathbb{A}^{\frac{1}{2}}\mathbb{H}^*\mathbb{A}\mathbb{H}\mathbb{A}^{\frac{1}{2}}.\end{aligned} \tag{4.36}$$

kde jsme využili mimo jiné toho, že $\mathbb{I}^* = \mathbb{I}$ a $\mathbb{A}^* = \mathbb{A}$. Kdybychom dokázali, že

$$\mathbb{A}^{\frac{1}{2}}(\mathbb{H} + \mathbb{H}^*)\mathbb{A}^{\frac{1}{2}} + \mathbb{A}^{\frac{1}{2}}\mathbb{H}^*\mathbb{A}\mathbb{H}\mathbb{A}^{\frac{1}{2}} \tag{4.37}$$

je pozitivně definitní, pak by to znamenalo, že vlastní čísla $\hat{\mathbb{B}}^*\hat{\mathbb{B}}$ jsou menší než 1, protože ve výrazu bychom měli identitu minus něco kladného (TODO doplnit jak aplikujeme lemma o polynomech).

$$\begin{aligned}\hat{\mathbb{B}}^*\hat{\mathbb{B}} &= \mathbb{I} - \mathbb{A}^{\frac{1}{2}}(\mathbb{H} + \mathbb{H}^*)\mathbb{A}^{\frac{1}{2}} + \mathbb{A}^{\frac{1}{2}}\mathbb{H}^*\mathbb{A}\mathbb{H}\mathbb{A}^{\frac{1}{2}} \\ &= \mathbb{I} - \mathbb{A}^{\frac{1}{2}}\mathbb{H}^*((\mathbb{H}^*)^{-1} + \mathbb{H}^{-1})\mathbb{H}\mathbb{A}^{\frac{1}{2}} + \mathbb{A}^{\frac{1}{2}}\mathbb{H}^*\mathbb{A}\mathbb{H}\mathbb{A}^{\frac{1}{2}} \\ &= \mathbb{I} - \mathbb{A}^{\frac{1}{2}}\mathbb{H}^*((\mathbb{H}^*)^{-1} + \mathbb{H}^{-1} - \mathbb{A})\mathbb{H}\mathbb{A}^{\frac{1}{2}} \\ &= \mathbb{I} - \mathbb{A}^{\frac{1}{2}}\mathbb{H}^*(\mathbb{W}^* + \mathbb{W} - \mathbb{A})\mathbb{H}\mathbb{A}^{\frac{1}{2}}.\end{aligned} \tag{4.38}$$

Z předpokladů je $\mathbb{W}^* + \mathbb{W} - \mathbb{A} > 0$, označme

$$\begin{aligned}\mathbb{M} &\coloneqq \mathbb{W}^* + \mathbb{W} - \mathbb{A}, \\ \hat{\mathbb{M}} &\coloneqq \mathbb{A}^{\frac{1}{2}}\mathbb{H}^*\mathbb{M}\mathbb{H}\mathbb{A}^{\frac{1}{2}}.\end{aligned} \tag{4.39}$$

Potom

$$(\mathbb{A}^{\frac{1}{2}}\mathbb{H}^*\mathbb{M}\mathbb{H}\mathbb{A}^{\frac{1}{2}}\vec{x},\vec{x}) = (\mathbb{M}\mathbb{H}\mathbb{A}^{\frac{1}{2}}\vec{x},\mathbb{H}\mathbb{A}^{\frac{1}{2}}\vec{x}) = (\mathbb{M}\vec{x}',\vec{x}') > 0, \tag{4.40}$$

kde jsme zavedli $\vec{x}' = \mathbb{H}\mathbb{A}^{\frac{1}{2}}\vec{x}$ a poslední nerovnost plyne z pozitivní definitnostni $\mathbb{M}$. Skutečně jsme tedy dokázali napsat $\hat{\mathbb{B}}^*\hat{\mathbb{B}}$ jako identitu minus pozitivně definitní matici $\hat{\mathbb{M}}$, a tedy

$$\lambda \in \sigma(\hat{\mathbb{B}}^*\hat{\mathbb{B}}) \implies \lambda = 1 - \lambda' \text{ pro } \lambda' \in \sigma(\hat{\mathbb{M}}) > 0. \tag{4.41}$$

Monotónní konvergenci nebudeme dále rozebírat. □

Poznámka 4.27. V případě metody bez předpodmínění je $\mathbb{H} = \mathbb{I}$, a tedy i $\mathbb{W} = \mathbb{H}^{-1} = \mathbb{I}$. Podmínka konvergence z předchozí věty tak přechází na tvar $2\mathbb{I} > \mathbb{A} > \mathbb{O}$, což odpovídá již dříve uvedenému kritériu.

Poznámka 4.28 (Volba předpodmiňovače)**.** Klíčovou otázkou je, jak správně zvolit matici $\mathbb{H}$. Ideální volbou by byla $\mathbb{H} = \mathbb{A}^{-1}$. V takovém případě by iterační matice byla $\mathbb{B} = \mathbb{I} - \mathbb{A}^{-1}\mathbb{A} = \mathbb{I} - \mathbb{I} = \mathbb{O}$. Spektrální poloměr nulové matice je 0, takže by metoda konvergovala v jediné iteraci.

$$\vec{x}^{(k+1)} = \vec{x}^{(k)} - \mathbb{H}\mathbb{A}\vec{x}^{(k)} + \mathbb{H}\vec{b} = \vec{x}^{(k)} - \mathbb{A}^{-1}\mathbb{A}\vec{x}^{(k)} + \mathbb{A}^{-1}\vec{b} = \vec{x}^{(k)} - \vec{x}^{(k)} + \vec{x}^* = \vec{x}^*. \tag{4.42}$$

Problém je, že výpočet inverzní matice $\mathbb{A}^{-1}$ je stejně náročný jako řešení původní soustavy, čemuž jsme se chtěli vyhnout. Umění předpodmínění spočívá v nalezení takové matice $\mathbb{H}$, která co nejlépe aproximuje $\mathbb{A}^{-1}$, ale zároveň je její aplikace (tj. násobení vektoru maticí $\mathbb{H}$) a konstrukce výrazně levnější než řešení soustavy s maticí $\mathbb{A}$.

4.1.5 Otázky

- Jaké jsou nutné a postačující podmínky konvergence stacionárních metod?
- Jaké máme apriorní a aposteriorní odhady chyby?
- Co je to předpodmínění, k čemu slouží a jak změní iterační předpis?

4.2 Richardsonovy iterace

Richardsonova metoda je jednou ze základních stacionárních metod. Využívá předpodmiňovač ve velmi jednoduché formě $\mathbb{H} = \theta\mathbb{I}$, kde $\theta \in \mathbb{R}$ je reálný parametr.

Definice 4.29 (Richardsonova metoda)**.** Pro daný relaxační parametr $\theta \in \mathbb{R}$ je Richardsonova metoda dána předpisem

$$\vec{x}^{(k+1)} = \vec{x}^{(k)} - \theta(\mathbb{A}\vec{x}^{(k)} - \vec{b}). \tag{4.43}$$

Jde o předpodmíněnou metodu postupných aproximací s předpodmiňovačem $\mathbb{H} = \theta\mathbb{I}$. Iterační matice a vektor jsou tedy

$$\mathbb{B} = \mathbb{I} - \theta\mathbb{A}, \quad \vec{c} = \theta\vec{b}. \tag{4.44}$$

Věta 4.30. Je-li matice $\mathbb{A}$ hermitovská, pak metoda Richardsonových iterací konverguje právě tehdy, když

$$\frac{2}{\theta}\mathbb{I} > \mathbb{A} > \mathbb{O}. \tag{4.45}$$

Konvergence je navíc monotónní vzhledem k vektorové normě $||\cdot||_{\mathbb{A}}$.

Důkaz. Využijeme věty (4.26), která říká, že je-li $\mathbb{A}$ hermitovská, pak předpodmíněná metoda postupných aproximací konverguje, pokud platí $\mathbb{W} + \mathbb{W}^* > \mathbb{A} > 0$, kde $\mathbb{W} = \mathbb{H}^{-1}$. V našem případě je $\mathbb{H} = \theta\mathbb{I}$, a tedy

$\mathbb{W} = \frac{1}{\theta}\mathbb{I}$. Protože θ je realné číslo, platí $\mathbb{W} + \mathbb{W}^* = \frac{2}{\theta}\mathbb{I}$. Podmínka konvergence tedy přechází na tvar

$$\mathbb{W} + \mathbb{W}^* = \frac{2}{\theta}\mathbb{I} > \mathbb{A} > 0. \tag{4.46}$$

□

Důsledek 4.31. Pro hermitovskou a pozitivně definitní matici $\mathbb{A}$ metoda konverguje, pokud je relaxační parametr θ zvolen v intervalu $0 < \theta < \frac{2}{\rho(\mathbb{A})}$.

Důkaz. Z lemma (4.22) plyne, že pokud si zadefinujeme polynom

$$p(x) = \frac{2}{\theta} - x, \tag{4.47}$$

pak pro každé $\lambda \in \sigma(\mathbb{A})$ platí $p(\lambda) = \frac{2}{\theta} - \lambda \in \sigma(p(\mathbb{A}))$. Dále $p(\mathbb{A}) = \frac{2}{\theta} - \mathbb{A} > 0$. Vlastní čísla matice $p(\mathbb{A})$ jsou tedy kladná, a proto $\frac{2}{\theta} - \lambda > 0$. To už jen přeuspořádáme a dostaneme $\theta < \frac{2}{\lambda}$ a tedy $\theta < \frac{2}{\rho(\mathbb{A})}$. □

4.2.1 Implementace

TODO

4.3 Jacobiho metoda

Jacobiho metodu navrhl v roce 1845 Carl Gustav Jakob Jacobi. Myšlenka této metody spočívá v tom, že v každé iteraci napočítáme i-tou složku nového vektoru $\vec{x}^{(k+1)}$ tak, aby byla přesně splněna i-tá rovnice $\sum_{j=1}^{n} a_{ij}x_j = b_i$ soustavy, přičemž pro všechny ostatní složky použijeme hodnoty z předchozí iterace $\vec{x}^{(k)}$.

Definice 4.32 (Jacobiho metoda)**.** Pro $i \in \hat{n}$ je Jacobiho metoda dána po složkách předpisem

$$x_i^{(k+1)} = \frac{1}{a_{ii}}\left(b_i - \sum_{j=1, j\neq i}^{n} a_{ij}x_j^{(k)}\right). \tag{4.48}$$

Samozřejmě zatím nevíme, zda opravdu takto vzniklé $\vec{x}^{(k+1)}$ je lepším řešením dané soustavy, to musíme analyzovat. Pro jednodušší analýzu si nejprve zavedeme maticový tvar Jacobiho metody.

Definice 4.33. Libovolnou čtvercovou matici $\mathbb{A}$ s nenulovými prvky na diagonále můžeme zapsat ve tvaru

$$\mathbb{A} = \mathbb{D} - \mathbb{L} - \mathbb{R}, \tag{4.49}$$

kde

- $\mathbb{D}$ je diagonální matice tvořená diagonálou matice $\mathbb{A}$.
- $\mathbb{L}$ je ostře[1] dolní trojúhelníková matice, jejíž prvky jsou záporně vzaté prvky matice $\mathbb{A}$ pod diagonálou.
- $\mathbb{R}$ je ostře horní trojúhelníková matice, jejíž prvky jsou záporně vzaté prvky matice $\mathbb{A}$ nad diagonálou.

Poznámka 4.34. Pomocí tohoto rozkladu můžeme iterační předpis Jacobiho metody přepsat do maticové podoby. Z definice plyne

$$\vec{x}^{(k+1)} = \mathbb{D}^{-1}\left[\vec{b} + (\mathbb{D} - \mathbb{A})\vec{x}^{(k)}\right] = (\mathbb{I} - \mathbb{D}^{-1}\mathbb{A})\vec{x}^{(k)} + \mathbb{D}^{-1}\vec{b}, \tag{4.50}$$

jelikož $\frac{1}{a_{ii}}$ můžeme reprezentovat jako $\mathbb{D}^{-1}$ a v sumě procházíme jednotlivé rovnice s vynecháním diagonálního prvku, tedy $(\mathbb{D} - \mathbb{A})x^{(k)}$. Z maticového tvaru vidíme, že Jacobiho metoda je stacionární iterační metoda s

[1] diagonála je nulová

předpodmiňovačem $\mathbb{H}_J = \mathbb{D}^{-1}$. Iterační matice a vektor jsou

$$\mathbb{B}_J = \mathbb{I} - \mathbb{D}^{-1}\mathbb{A}, \quad \vec{c}_j = \mathbb{D}^{-1}\vec{b}. \tag{4.51}$$

Poznámka 4.35. Obecně pokud v libovolné stacionární metodě použijeme předpodmiňovač $\mathbb{H} = \mathbb{D}^{-1}$, pak mluvíme o tzv. Jacobiho podmínění.

Poznámka 4.36. Bude se nám také hodit tvar $\mathbb{B}_J = \mathbb{D}^{-1}(\mathbb{L} + \mathbb{R})$, který plyne čistě z dosazení rozkladu $\mathbb{A} = \mathbb{D} - \mathbb{L} - \mathbb{R}$ do vztahu $\vec{x}^{(k+1)} = \mathbb{D}^{-1}\left[\vec{b} + (\mathbb{D} - \mathbb{A})\vec{x}^{(k)}\right]$.

Věta 4.37 (Konvergence Jacobiho metody)**.** Nutnou a postačující podmínkou pro to, aby Jacobiho metoda konvergovala pro libovolnou počáteční aproximaci $\vec{x}^{(0)}$, je

$$\rho(\mathbb{D}^{-1}(\mathbb{L} + \mathbb{R})) < 1. \tag{4.52}$$

Postačující podmínkou je $||\mathbb{D}^{-1}(\mathbb{L} + \mathbb{R})|| < 1$ v libovolné maticové normě.

Důkaz. Důkaz opět plyne z předchozích obecných podmínek pro konvergenci stacionárních metod, viz (4.25). □

Definice 4.38 (Matice s ostře převažující diagonálou)**.** Řekneme, že matice $\mathbb{A}$ má ostře (řádkově) převažující diagonálu, pokud pro všechny řádky platí

$$\sum_{j=1, j\neq i}^{n} |a_{ij}| < |a_{ii}|, \quad \forall i = 1, \ldots, n. \tag{4.53}$$

Slovy musí platit, že součet absolutních hodnot všech mimodiagonálních prvků v každém řádku je menší než absolutní hodnota diagonálního prvku v tomtéž řádku.

Věta 4.39. Má-li matice $\mathbb{A}$ ostře převažující diagonálu, pak Jacobiho metoda konverguje pro libovolnou počáteční aproximaci $\vec{x}^{(0)}$.

Důkaz. Nechť $\mathbb{A}$ má ostře převažující diagonálu. Ukážeme, že v takovém případě $||\mathbb{B}||_{+\infty} < 1$. Jak jsme dříve odvodili, pro Jacobiho metodu platí, že $\mathbb{B}_J = \mathbb{D}^{-1}(\mathbb{L} + \mathbb{R})$. Potom

$$b_{ij} = \begin{cases} 0 \text{ pro } i = j, \\ \frac{-a_{ij}}{a_{ii}} \text{ pro } i \neq j. \end{cases} \tag{4.54}$$

Připomeňme, že maximovou normu lze ekvivalentně přepsat jako řádkovou normu, viz (1.91), tedy

$$||\mathbb{B}||_{+\infty} = \max_{i\in\hat{n}} \sum_{j=1}^{n} |b_{ij}| = \max_{i\in\hat{n}} \sum_{j=1, j\neq i}^{n} \frac{|a_{ij}|}{|a_{ii}|} < 1, \tag{4.55}$$

kde jsme využili definice ostře převládající diagonály, tedy že $\forall\, i \in \hat{n}$ platí $\sum_{j=1, j\neq i}^{n} |a_{ij}| < |a_{ii}|$. □

Věta 4.40. Je-li matice $\mathbb{A}$ hermitovská, pak Jacobiho metoda konverguje právě tehdy, když

$$2\mathbb{D} > \mathbb{A} > 0 \tag{4.56}$$

Konvergence je navíc monotónní vzhledem k vektorové normě $||\cdot||_{\mathbb{A}}$.

Důkaz. Opět využijeme věty (4.26), která říká, že je-li $\mathbb{A}$ hermitovská, pak předpodmíněná metoda postupných aproximací konverguje, pokud platí $\mathbb{W} + \mathbb{W}^* > \mathbb{A} > 0$, kde $\mathbb{W} = \mathbb{H}^{-1}$. V našem případě je $\mathbb{H} = \mathbb{D}^{-1}$, a tedy $\mathbb{W} = \mathbb{D}$. Protože $\mathbb{A}$ je hermitovská, má na diagonále reálné prvky, a tedy $\mathbb{W} = \mathbb{D} \in \mathbb{R}^{n,n}$. Odtud plyne, že

$\mathbb{W} + \mathbb{W}^* = 2\mathbb{D}$. Aby byly splněny podmínky konvergence z věty (4.26), musíme mít

$$2\mathbb{D} > \mathbb{A} > 0, \tag{4.57}$$

což ale dává dokazované tvrzení. □

4.3.1 Implementace

TODO

4.4 Gaussova-Seidelova metoda

Tuto metodu popsal Carl Friedrich Gauss v roce 1823 a nezávisle na něm Philipp Ludwig von Seidel v roce 1874. Na rozdíl od Jacobiho metody využívá Gaussova-Seidelova metoda při výpočtu složky $x_i^{(k+1)}$ již dříve napočítané složky $\vec{x}_1^{(k+1)}, \ldots, \vec{x}_{i-1}^{(k+1)}$ z aktuální iterace. Tím se často dosáhne rychlejší konvergence.

Definice 4.41 (Gaussova-Seidelova metoda)**.** Pro $i \in \hat{n}$ je Gaussova-Seidelova metoda dána po složkách předpisem

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k)} \right). \tag{4.58}$$

Poznámka 4.42. Její maticový tvar odvodíme pomocí rozkladu $\mathbb{A} = \mathbb{D} - \mathbb{L} - \mathbb{R}$. Z definice plyne

$$\begin{aligned} \vec{x}^{(k+1)} &= \mathbb{D}^{-1} \left[\vec{b} + \mathbb{L}\vec{x}^{(k+1)} + \mathbb{R}\vec{x}^{(k)} \right], \\ \mathbb{D}\vec{x}^{(k+1)} - \mathbb{L}\vec{x}^{(k+1)} &= \vec{b} + \mathbb{R}\vec{x}^{(k)} \\ \vec{x}^{(k+1)} &= (\mathbb{D} - \mathbb{L})^{-1}(\vec{b} + \mathbb{R}\vec{x}^{(k)}) \\ \vec{x}^{(k+1)} &= (\mathbb{D} - \mathbb{L})^{-1}\mathbb{R}\vec{x}^{(k)} + (\mathbb{D} - \mathbb{L})^{-1}\vec{b} \end{aligned} \tag{4.59}$$

Protože

$$(\mathbb{D} - \mathbb{L})^{-1}\mathbb{R} = (\mathbb{D} - \mathbb{L})^{-1}(\mathbb{D} - \mathbb{L} - \mathbb{A}) = \mathbb{I} - (\mathbb{D} - \mathbb{L})^{-1}\mathbb{A}, \tag{4.60}$$

je

$$\vec{x}^{(k+1)} = (\mathbb{I} - (\mathbb{D} - \mathbb{L})^{-1}\mathbb{A})\vec{x}^{(k)} + (\mathbb{D} - \mathbb{L})^{-1}\vec{b}. \tag{4.61}$$

Z maticového tvaru vidíme, že Gaussova-Seidelova metoda je stacionární iterační metoda s předpodmiňovačem $\mathbb{H}_{GS} = (\mathbb{D} - \mathbb{L})^{-1}$. Iterační matice a vektor jsou

$$\mathbb{B}_{GS} = (\mathbb{D} - \mathbb{L})^{-1}\mathbb{R}, \quad \vec{c}_{gs} = (\mathbb{D} - \mathbb{L})^{-1}\vec{b}. \tag{4.62}$$

Věta 4.43 (Konvergence Gaussovy-Seidelovy metody)**.** Nutnou a postačující podmínkou pro to, aby Gaussova-Seidelova metoda konvergovala pro libovolnou $\vec{x}^{(0)}$, je

$$\rho((\mathbb{D} - \mathbb{L})^{-1}\mathbb{R}) < 1. \tag{4.63}$$

Postačující podmínkou je $||(\mathbb{D} - \mathbb{L})^{-1}\mathbb{R}|| < 1$ v některé maticové normě.

Důkaz. Důkaz opět plyne z předchozích obecných podmínek pro konvergenci stacionárních metod, viz (4.25). □

Věta 4.44. Má-li matice $\mathbb{A}$ ostře převažující diagonálu, pak Gaussova-Seidelova metoda konverguje pro libovolnou počáteční aproximaci $\vec{x}^{(0)}$.

Důkaz. Nechť $\mathbb{A}$ má ostře převažující diagonálu. Stejně jako u Jacobiho metody ukážeme, že v takovém případě $||\mathbb{B}||_{+\infty} < 1$. Jak jsme dříve odvodili, pro G.S. metodu platí, že $\mathbb{B}_{GS} = (\mathbb{D} - \mathbb{L})^{-1}\mathbb{R}$. Připomeňme, že maximová norma je definovaná vztahem

$$||\mathbb{B}_{GS}||_{+\infty} = \max_{||\vec{x}||_\infty = 1} ||\mathbb{B}_{GS}\vec{x}||_{+\infty}. \tag{4.64}$$

Označme $\vec{u} = \arg\max_{||\vec{x}||_\infty=1} ||\mathbb{B}_{GS}\vec{x}||$, tzn. $\vec{u}$ představuje vektor, ve kterém se nabývá maxima. Dále označme $\vec{v} = \mathbb{B}_{GS}\vec{u}$, abychom mohli zjednodušeně psát

$$||\mathbb{B}_{GS}||_{+\infty} = ||\mathbb{B}_{GS}\vec{u}||_{+\infty} = ||\vec{v}||_{+\infty}. \tag{4.65}$$

Maximová norma vektoru $\vec{v}$ je dána vztahem

$$||\vec{v}||_{+\infty} = \max_{i\in\hat{n}} |v_i|, \tag{4.66}$$

jde tedy o maximální hodnotu absolutní hodnoty složek vektoru $\vec{v}$. Označíme si index složky, ve které je maxima nabýváno: $s = \arg\max_{i\in\hat{n}} |v_i|$. Odtud pak

$$||\mathbb{B}||_{+\infty} = ||\vec{v}||_{+\infty} = |v_s| \tag{4.67}$$

Dosaďme nyní zpět do vztahu pro $\vec{v}$:

$$\begin{aligned} \vec{v} = \mathbb{B}_{GS}\vec{u} &= (\mathbb{D} - \mathbb{L})^{-1}\mathbb{R}\vec{u}, \\ (\mathbb{D} - \mathbb{L})\vec{v} &= \mathbb{R}\vec{u}, \\ (\mathbb{D} - \mathbb{L})\vec{v} - \mathbb{R}\vec{u} &= \mathbb{O}, \end{aligned} \tag{4.68}$$

Získali jsme soustavu lineárních rovnic, a my se podíváme na její s-tou rovnici,

$$\sum_{j=1}^{s-1} a_{sj}v_j + a_{ss}v_s + \sum_{j=s+1}^{n} a_{sj}u_j = 0 \tag{4.69}$$

kde

$$\begin{aligned} \sum_{j=1}^{s-1} a_{sj}v_j &= -\mathbb{L}\vec{v} \\ a_{ss}v_s &= \mathbb{D}\vec{v} \\ \sum_{j=s+1}^{n} a_{sj}u_j &= -\mathbb{R}\vec{u} \end{aligned} \tag{4.70}$$

z definice rozkladu $\mathbb{A} = \mathbb{D} - \mathbb{L} - \mathbb{R}$. Z rovnice si vyjádříme prvek v_s:

$$v_s = \frac{1}{a_{ss}} \left(-\sum_{j=1}^{s-1} a_{sj}v_j - \sum_{j=s+1}^{n} a_{sj}u_j \right). \tag{4.71}$$

Potom s využitím trojíhelníkové nerovnosti dostaneme

$$|v_s| \leq \sum_{j=1}^{s-1} \frac{|a_{sj}|}{|a_{ss}|}|v_j| + \sum_{j=s+1}^{n} \frac{|a_{sj}|}{|a_{ss}|}|u_j|. \tag{4.72}$$

Vzpomeneme si, že jsme si zadefinovali $\vec{u}$ jako vektor, ve kterém maximová norma $||\mathbb{B}_{GS}||_\infty$ nabývá maxima. Proto $||\vec{u}||_\infty = 1$ a tedy $\forall\, j \in \hat{n}$ je $|u_j| \leq 1$. Dále jsme volili $||\vec{v}||_\infty = |v_s|$, okud $\forall\, j \in \hat{n}$ je $|v_j| \leq |v_s|$. Když tyto

dvě nerovnosti použijeme k odhadu předešlého vztahu, dostaneme

$$|v_s| \leq \sum_{j=1}^{s-1} \frac{|a_{sj}|}{|a_{ss}|}|v_s| + \sum_{j=s+1}^{n} \frac{|a_{sj}|}{|a_{ss}|} = |v_s| \left(\sum_{j=1}^{s-1} \frac{|a_{sj}|}{|a_{ss}|} \right) + \sum_{j=s+1}^{n} \frac{|a_{sj}|}{|a_{ss}|} = |v_s| \cdot \alpha + \beta, \tag{4.73}$$

kde jsme jednotlivé součty označili jako α a β. Z definice převládající diagonály plyne, že $\alpha + \beta < 1$, jelikož součet $\alpha + \beta$ představuje součet absolutních hodnot všech mimodiagonálních prvků v řádku s-té matice $\mathbb{A}$, a tedy $|a_{ss}|(\alpha + \beta) = |a_{ss}| \cdot \sum_{j=1, j\neq s}^{n} |a_{sj}| < |a_{ss}|$. Dohromady máme

$$\begin{aligned} |v_s| &\leq |v_s| \cdot \alpha + \beta \\ |v_s|(1 - \alpha) &\leq \beta \\ |v_s| &\leq \frac{\beta}{1 - \alpha} < 1, \end{aligned} \tag{4.74}$$

kde poslední nerovnost plyne z toho, že $\alpha + \beta < 1$, tedy $\beta < 1 - \alpha$. Nakonec už se zbývá jen vrátit k původním informacím o maximové normě:

$$||\mathbb{B}_{GS}||_{+\infty} = |v_s| < 1. \tag{4.75}$$

□

Věta 4.45. Je-li matice $\mathbb{A}$ hermitovská a pozitivně definitní, pak Gaussova-Seidelova metoda konverguje pro libovolnou počáteční aproximaci $\vec{x}^{(0)}$. Konvergence je navíc monotónní vzhledem k vektorové normě $||\cdot||_{\mathbb{A}}$.

Důkaz. Opět využijeme věty (4.26), která říká, že je-li $\mathbb{A}$ hermitovská, pak předpodmíněná metoda postupných aproximací konverguje, pokud platí $\mathbb{W} + \mathbb{W}^* > \mathbb{A} > 0$, kde $\mathbb{W} = \mathbb{H}^{-1}$. V našem případě je $\mathbb{H} = (\mathbb{D} - \mathbb{L})^{-1}$, a tedy $\mathbb{W} = \mathbb{D} - \mathbb{L}$. Připomeňme, že vzhledem k hermitovskosti matice $\mathbb{A}$ platí:

$$\mathbb{D} - \mathbb{L} - \mathbb{R} = \mathbb{A} = \mathbb{A}^* = \mathbb{D}^* - \mathbb{L}^* - \mathbb{R}^*, \tag{4.76}$$

a tedy $\mathbb{D} = \mathbb{D}^*$ (diagonála hermitovské matice je reálná). Když si také představíme co dělá „ohvězdíčkování“, např. s dolní trojúhelníkovou maticí $\mathbb{L}$, dostaneme, že vzniká horní trojúhelníková matice, a tedy podle definice rozkladu je $\mathbb{L}^* = \mathbb{R}$. Podobně $\mathbb{R}^* = \mathbb{L}$. Nyní

$$\mathbb{W}^* + \mathbb{W} = \mathbb{D}^* - \mathbb{L}^* + \mathbb{D} - \mathbb{L} = 2\mathbb{D} - \mathbb{L} - \mathbb{R} = \mathbb{D} + \mathbb{A}, \tag{4.77}$$

a abychom splnili předpoklady věty (4.26), musíme mít

$$\mathbb{W}^* + \mathbb{W} = \mathbb{D} + \mathbb{A} > \mathbb{A} > 0. \tag{4.78}$$

Tato podmínka je ale ekvivalentní tomu, že $\mathbb{D} + \mathbb{A} - \mathbb{A} > 0$, resp. $\mathbb{D} > 0$, což skutečně platí, protože $\mathbb{A}$ má z pozitivní definitnosti na diagonále kladné prvky. Tyto prvky tvoří matici $\mathbb{D}$ a tedy i ona je pozitivně definitní. □

Poznámka 4.46 (Srovnání Jacobiho a Gaussovy-Seidelovy metody)**.** Všiměme si, že Gaussova-Seidelova metoda poskytne (ve většině případů) rychlejší konvergenci než Jacobiho metoda, protože matice $\mathbb{H}_{GS} = (\mathbb{D} - \mathbb{L})^{-1}$ je lepší aproximací $\mathbb{A}^{-1}$ než $\mathbb{H}_J = \mathbb{D}^{-1}$. Navíc existují případy, kdy Gaussova-Seidelova metoda konverguje, zatímco Jacobiho metoda nikoli (například pro pozitivně definitní matici $\mathbb{A}$, pro kterou $2\mathbb{D} - \mathbb{A}$ není pozitivně definitní, viz věty o konvergenci Jacobiho metody). Existují však i případy, kdy je tomu naopak.

4.5 Superrelaxační metoda (SOR)

Metodu SOR (Successive Over-Relaxation) představil David M. Young, Jr. v roce 1950. Jedná se o modifikaci a zobecnění Gaussovy-Seidelovy metody, která zavedením nového parametru umožňuje výrazně urychlit konver-

genci. Metoda je obzvláště efektivní pro řešení soustav lineárních rovnic, které vznikají při numerickém řešení parciálních diferenciálních rovnic, například metodou konečných diferencí. Základní myšlenka je následující: krok Gaussovy-Seidelovy metody můžeme zapsat jako úpravu stávající aproximace o jistou korekci:

$$x_i^{(k+1)} = x_i^{(k)} + \Delta x_i^{(k)}, \tag{4.79}$$

kde $\Delta x_i^{(k)}$ je přesně taková změna, aby byla splněna i-tá rovnice soustavy s již nově napočítanými složkami. Superrelaxační metoda tuto korekci škáluje pomocí tzv. relaxačního parametru ω, tedy

$$x_i^{(k+1)} = x_i^{(k)} + \omega \Delta x_i^{(k)}, \tag{4.80}$$

a je zřejmé, že pro $\omega = 1$ se jedná o Gaussovu-Seidelovu metodu. Máme-li tedy v Gaussově-Seidelově metodě pro i-tou složku následující vztah

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k)} \right), \tag{4.81}$$

a chceme-li tento vztah přepsat do tvaru (4.79), pak pro zachování rovnost přičteme a zase odečteme $x_i^{(k)}$:

$$x_i^{(k+1)} = x_i^{(k)} + \frac{1}{a_{ii}} \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - a_{ii} x_i^{(k)} - \sum_{j=i}^{n} a_{ij} x_j^{(k)} \right). \tag{4.82}$$

Nyní už jen zbývá zavést relaxační parametr ω.

Definice 4.47 (Superrelaxační metoda (SOR))**.** Pro daný relaxační parametr $\omega \in \mathbb{R}$ je SOR metoda definována po složkách jako

$$x_i^{(k+1)} = x_i^{(k)} + \omega \left(\frac{1}{a_{ii}} \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i}^{n} a_{ij} x_j^{(k)} \right) \right). \tag{4.83}$$

Poznámka 4.48. Maticový tvar metody odvodíme opět z její definice:

$$\begin{aligned} \vec{x}^{(k+1)} &= \vec{x}^{(k)} + \omega \mathbb{D}^{-1} \left(\vec{b} + \mathbb{L}\vec{x}^{(k+1)} - \mathbb{D}\vec{x}^{(k)} + \mathbb{R}\vec{x}^{(k)} \right), \\ \mathbb{D}\vec{x}^{(k+1)} - \omega \mathbb{L}\vec{x}^{(k+1)} &= \mathbb{D}\vec{x}^{(k)} - \omega \mathbb{D}\vec{x}^{(k)} + \omega \vec{b} + \omega \mathbb{R}\vec{x}^{(k)} \\ (\mathbb{D} - \omega \mathbb{L})\vec{x}^{(k+1)} &= (\mathbb{D} - \omega \mathbb{D} + \omega \mathbb{R})\vec{x}^{(k)} + \omega \vec{b} \\ \vec{x}^{(k+1)} &= (\mathbb{D} - \omega \mathbb{L})^{-1}(\mathbb{D} - \omega \mathbb{D} + \omega \mathbb{R})\vec{x}^{(k)} + \omega (\mathbb{D} - \omega \mathbb{L})^{-1} \vec{b}. \end{aligned} \tag{4.84}$$

Protože

$$(\mathbb{D} - \omega \mathbb{L})^{-1}(\mathbb{D} - \omega \mathbb{D} + \omega \mathbb{R}) = (\mathbb{D} - \omega \mathbb{L})^{-1}(\mathbb{D} - \omega \mathbb{L} + \omega \mathbb{L} - \omega \mathbb{D} + \omega \mathbb{R}) = \mathbb{I} - \omega (\mathbb{D} - \omega \mathbb{L})^{-1} \mathbb{A}, \tag{4.85}$$

je

$$\vec{x}^{(k+1)} = (\mathbb{D} - \omega \mathbb{L})^{-1}(\mathbb{I} - \omega (\mathbb{D} - \omega \mathbb{L})^{-1} \mathbb{A})\vec{x}^{(k)} + \omega (\mathbb{D} - \omega \mathbb{L})^{-1} \vec{b}. \tag{4.86}$$

Z maticového tvaru vidíme, že SOR je stacionární metoda s předpodmiňovačem $\mathbb{H}_\omega = \omega(\mathbb{D} - \omega \mathbb{L})^{-1}$. Iterační matice a vektor jsou

$$\mathbb{B}_\omega = (\mathbb{D} - \omega \mathbb{L})^{-1} \big((1 - \omega)\mathbb{D} + \omega \mathbb{R} \big) = \mathbb{I} - \omega (\mathbb{D} - \omega \mathbb{L})^{-1} \mathbb{A} \tag{4.87}$$

Věta 4.49 (Konvergence SOR)**.** Nutnou a postačující podmínkou pro konvergenci SOR metody je $\rho(\mathbb{B}_\omega) < 1$. Postačující podmínkou je $||\mathbb{B}_\omega|| < 1$ v některé maticové normě.

Důkaz. Důkaz opět plyne z předchozích obecných podmínek pro konvergenci stacionárních metod, viz (4.25). □

Věta 4.50. Pro $\omega \in \mathbb{R}$ platí,

$$\rho(\mathbb{B}_\omega) \geq |\omega - 1|. \tag{4.88}$$

Proto SOR metoda diverguje $\forall \omega \notin [0, 2]$.

Důkaz. Vzpomeňme si, že dle Jordanovy věty (1.47) je každá matice podobná Jordanově kanonickém tvaru, tedy $\mathbb{A} = \mathbb{X}^{-1}\mathbb{J}\mathbb{X}$. Odtud také

$$\det \mathbb{A} = \frac{1}{\det \mathbb{X}} \det \mathbb{J} \det \mathbb{X} = \det \mathbb{J} = \prod_{i=1}^{n} \lambda_i, \tag{4.89}$$

kde poslední rovnost platí, protože Jordanova matice $\mathbb{J}$ je horní trojúhelníková s vlastními čísly na diagonále. Nyní využijme toho, že iterační matice pro SOR metodu lze vyjádřit jako $\mathbb{B}_\omega = (\mathbb{D} - \omega\mathbb{L})^{-1}\big((1-\omega)\mathbb{D} + \omega\mathbb{R}\big)$ a aplikujme výše zmíněný poznatek. Pro $\lambda_i \in \sigma(\mathbb{B}_\omega)$ tedy

$$\prod_{i=1}^{n} \lambda_i = \det \mathbb{B}_\omega = \frac{1}{\det(\mathbb{D} - \omega\mathbb{L})} \det\left((1-\omega)\mathbb{D} + \omega\mathbb{R}\right) = \prod_{i=1}^{n} \frac{1}{d_{ii}} \cdot \prod_{i=1}^{n} (1-\omega) d_{ii} = (1-\omega)^n, \tag{4.90}$$

kde jsme využili toho, že jak $\mathbb{D} - \omega\mathbb{L}$, tak $(1-\omega)\mathbb{D} + \omega\mathbb{R}$ jsou trojúhelníkové matice, a tedy determinanty odpovídají součinu diagonálních prvků. Potom

$$\prod_{i=1}^{n} |\lambda_i| = |(1-\omega)|^n, \tag{4.91}$$

a tedy buď $\forall i \in \hat{n}$ je $|\lambda_i| = |1 - \omega|$, což by implikovalo, že $\rho(\mathbb{B}_\omega) = |1 - \omega|$, nebo alespoň pro jedno i je $|\lambda_i| < |1 - \omega|$, a pak aby platila rovnost musí existovat alespoň jedno j takové, že $|\lambda_j| > |1 - \omega|$, a pak $\rho(\mathbb{B}_\omega) > |1 - \omega|$. To už je tvrzení věty. □

Věta 4.51. Má-li matice $\mathbb{A}$ ostře převažující diagonálu, pak metoda SOR konverguje pro $\omega \in (0, 1]$.

Důkaz. Nepřednášel se. □

Věta 4.52 (Ostrowski)**.** Je-li matice $\mathbb{A}$ hermitovská a pozitivně definitní, pak metoda SOR konverguje právě tehdy, když $\omega \in (0, 2)$. Konvergence je navíc monotónní vzhledem k vektorové normě $||\cdot||_{\mathbb{A}}$.

Důkaz. Opět využijeme věty (4.26), která říká, že je-li $\mathbb{A}$ hermitovská, pak předpodmíněná metoda postupných aproximací konverguje, pokud platí $\mathbb{W} + \mathbb{W}^* > \mathbb{A} > 0$, kde $\mathbb{W} = \mathbb{H}^{-1}$. V našem případě je $\mathbb{H} = \omega(\mathbb{D} - \omega\mathbb{L})^{-1}$, a tedy $\mathbb{W} = \frac{1}{\omega}(\mathbb{D} - \omega\mathbb{L})$. Vzpomeňme si, že jsme v analogickém důkazu pro Gaussovu-Seidelovu metodu ukázali, že z

$$\mathbb{D}^* - \mathbb{L}^* - \mathbb{R}^* = \mathbb{A}^* = \mathbb{A} = \mathbb{D} - \mathbb{L} - \mathbb{R}, \tag{4.92}$$

plynou rovnosti $\mathbb{D} = \mathbb{D}^*$, $\mathbb{L} = \mathbb{R}^*$ a $\mathbb{R} = \mathbb{L}^*$. Nyní

$$\mathbb{W}^* + \mathbb{W} = \frac{1}{\omega}(\mathbb{D} - \omega\mathbb{R} + \mathbb{D} - \omega\mathbb{L}) = \frac{2}{\omega}\mathbb{D} - \mathbb{L} - \mathbb{R} = \left(\frac{2}{\omega} - 1\right)\mathbb{D} + \mathbb{D} - \mathbb{L} - \mathbb{R} = \left(\frac{2}{\omega} - 1\right)\mathbb{D} + \mathbb{A}. \tag{4.93}$$

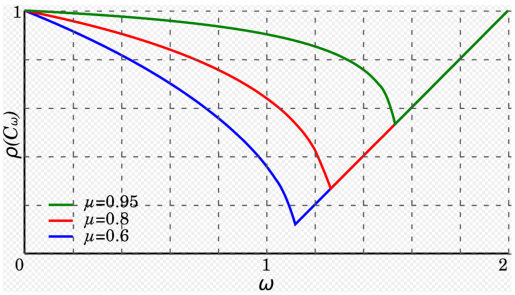
Aby byly splněny podmínky věty (4.26), musíme mít

$$\mathbb{W}^* + \mathbb{W} = \left(\frac{2}{\omega} - 1\right)\mathbb{D} + \mathbb{A} > \mathbb{A} > 0, \tag{4.94}$$

a tedy musí být $\left(\frac{2}{\omega} - 1\right) > 0$. Stejným argumentem jako v důkazu pro Gaussovu-Seidelovu je $\mathbb{D} > 0$ a tedy zbývá $\frac{2}{\omega} - 1 > 0$, což je splněno pro $\omega \in (0, 2)$. □

4.5.1 Optimální volba relaxačního parametru

Kromě samotné konvergence nás zajímá i její rychlost. Cílem je nalézt takový optimální parametr ω_{opt}, který minimalizuje spektrální poloměr iterační matice $\rho(\mathbb{B}_\omega)$, a tím maximalizuje rychlost konvergence. (TODO: přidat odkaz proč tohle zrychluje konvergenci) Analýza optimální volby ω budeme provádět pro speciální třídu matic, tzv. dvoucyklických a shodně uspořádaných. Tyto matice typicky vznikají při diskretizaci eliptických a parabolických parciálních diferenciálních rovnic metodou konečných diferencí. TODO: Nezkouší se



Obrázek 4.3: závislost spektrálního poloměru $\rho(\mathbb{B}_\omega)$ na relaxačním parametru ω pro dvoucyklickou a shodně uspořádanou matici

4.6 Korektnost a porovnání konvergence probraných iterativních metod

Než se dostaneme k samotnému shrnutí, pojďme se ještě podívat na korektnost. Problém by mohl nastat při dělení $\frac{1}{a_{ii}}$, je tedy potřeba garantovat, že $\mathbb{D} = \mathbb{O}$. Používali jsme dva různé předpoklady:

- Má-li matice $\mathbb{A}$ ostře převažující diagonálu, pak z ostré nerovnosti nutně plyne, že $a_{ii} \neq 0$.
- Je-li matice $\mathbb{A}$ hermitovská a pozitivně definitní, pak má kladné diagonální prvky, tedy $a_{ii} > 0$.

Tyto předpoklady take zajišťují existenci matic $\mathbb{D}^{-1}$ a $(\mathbb{D} - \mathbb{L})^{-1}$. Následující tabulka shrnuje podmínky konvergence pro jednotlivé iterační metody v závislosti na používaných předpokladech.

	s převládající diagonálou	hermitovská
Jacobi	ano	$2\mathbb{D} > \mathbb{A} > 0$
Gauss-Seidel	ano	$\mathbb{A} > 0$
SOR	ano	$\mathbb{A} > 0$

4.7 Porovnání přímých a iteračních metod

Volba mezi přímou metodou (jako je Gaussova eliminace) a iterační metodou závisí na vlastnostech řešené soustavy.

4.7.1 Výhody iteračních metod

- **Výpočetní složitost:** Při vhodném použití jsou často výrazně rychlejší. Složitost Gaussovy eliminace je $O(n^3)$, zatímco složitost jedné iterace je typicky $O(n^2)$ (pro násobení matice a vektoru). Pro dobrou konvergenci stačí často jen malý počet iterací, nezávislý na velikosti matice.
- **Využití sparsity:** Iterační metody dokáží lépe využít vlastností řídkých matic (matic s velkým počtem nulových prvků). Při efektivním uložení řídké matice může být složitost jedné iterace mnohem nižší než $O(n^2)$. Naopak Gaussova eliminace může řídkou matici zaplnit novými nenulovými prvky (tzv. fill-in), což zvyšuje paměťové nároky.

- **Implementace a paměť:** Iterační metody nemění původní matici $\mathbb{A}$, což zjednodušuje implementaci. V některých případech ani není nutné matici $\mathbb{A}$ explicitně ukládat do paměti, pokud umíme efektivně počítat součin $\mathbb{A}\vec{x}$.
- **Využití apriorní informace:** Pokud je k dispozici dobrá počáteční aproximace řešení, iterační metody mohou konvergovat velmi rychle. Přímé metody tuto informaci využít nemohou.

4.7.2 Nevýhody iteračních metod

- **Přesnost řešení:** Většina metod neposkytne teoreticky přesné řešení v konečném počtu kroků.
- **Ukončovací kritérium:** Je nutné definovat kritérium, kdy je aproximace "dostatečně"přesná a výpočet lze ukončit.
- **Závislost na matici:** Doba výpočtu (počet iterací) silně závisí na vlastnostech matice $\mathbb{A}$. Konvergence není zaručena pro každou regulární matici, na rozdíl od modifikované Gaussovy eliminace.
- **Složitost analýzy:** Analýza konvergence a volba optimálních parametrů je často velmi složitá.

4.8 Otázky

- Jacobiho, Gaussova-Seidelova a SOR metoda: odvození, maticový tvar a konvergence
- Porovnání přímých a iteračních metod pro řešení soustav lineárních rovnic

Kapitola 5

Metody výpočtu vlastních čísel matic

V předchozích kapitolách jsme se naučili řešit soustavy lineárních rovnic. Nyní se zaměříme na druhý fundamentální problém lineární algebry: výpočet vlastních čísel a vlastních vektorů matic.

5.1 Aplikace a motivace

Problém nalezení vlastních čísel a vlastních vektorů má široké uplatnění v mnoha oblastech vědy a techniky. Vlastní čísla často reprezentují fundamentální charakteristiky systémů. Mezi klíčové aplikace patří:

- **Analýza vibrací a rezonancí:** V mechanice a stavebnictví odpovídají vlastní čísla vlastním frekvencím kmitání konstrukcí, jako jsou nosníky, mosty nebo křídla letadel. Znalost těchto frekvencí je klíčová pro návrh bezpečných a stabilních struktur, jak ukazuje například slavný kolaps mostu Tacoma Narrows.
- **Kvantová mechanika:** Vlastní čísla a vlastní vektory hrají ústřední roli při popisu atomárních a molekulárních systémů. Vlastní čísla Hamiltonova operátoru odpovídají energetickým hladinám atomu, což se projevuje například ve formě atomových spekter.
- **Zpracování obrazu a dat:** Metody jako analýza hlavních komponent (PCA) využívají vlastní vektory (tzv. "eigenfaces") pro rozpoznávání obličejů a kompresi dat.
- **Analýza sítí a webu:** Algoritmus PageRank, který používá Google pro hodnocení důležitosti webových stránek, je založen na výpočtu dominantního vlastního vektoru obrovské matice sousednosti webového grafu.

5.2 Základní pojmy

Na rozdíl od řešení soustav lineárních rovnic, kde existují přímé metody (jako GEM), je obecný problém nalezení všech vlastních čísel matice fundamentálně odlišný, jak ukáže analýza důsledků následující věty.

Věta 5.1. Nalezení kořenů libovolného polynomu $p_n(x) = \sum_{k=0}^{n} a_k x^k$ stupně n je ekvivalentní výpočtu spektra tzv. Frobeniovy doprovodné matice $\mathbb{C} \in \mathbb{R}^{n,n}$ definované jako

$$\mathbb{C} = \begin{pmatrix} -a_{n-1}/a_n & -a_{n-2}/a_n & \dots & -a_1/a_n & -a_0/a_n \\ 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & 0 \end{pmatrix}. \tag{5.1}$$

Důkaz. Nepřednáší se. □

Poznámka 5.2. Z lineární algebry víme, že pro polynomy stupně $n \geq 5$ neexistuje obecný vzorec pro nalezení kořenů pomocí konečného počtu aritmetických operací a odmocnin. Z předchozí věty tak přímo plyne, že nemůže existovat přímá metoda pro výpočet kompletního spektra libovolné matice řádu $n \geq 5$.

Poznámka 5.3. Z nemožnosti existence přímých metod plyne, že všechny obecné metody pro výpočet vlastních čísel musí být ze své podstaty **iterační**. Stejně jako u iteračních metod pro řešení soustav je tedy nutné mít k dispozici spolehlivé aposteriorní odhady chyb, které nám umožní definovat vhodná ukončovací kritéria pro výpočet. Apriorní odhady chyb v tomto kurzu probírat nebudeme.

5.2.1 Lokalizace a odhady chyb vlastních čísel

Prvním krokem při analýze je často lokalizace, tedy určení oblasti v komplexní rovině, kde se vlastní čísla nacházejí. Následně při samotném výpočtu potřebujeme odhadovat chybu našich aproximací.

Věta 5.4 (Aposteriorní odhad chyby pro hermitovské matice)**.** Nechť $\mathbb{A} \in \mathbb{C}^{n,n}$ je hermitovská matice. Nechť $\hat{\lambda}$ je aproximace některého jejího vlastního čísla a $\hat{\vec{x}} \neq \vec{0}$ je aproximace příslušného vlastního vektoru. Pro reziduální vektor

$$\vec{r} = \mathbb{A}\hat{\vec{x}} - \hat{\lambda}\hat{\vec{x}} \tag{5.2}$$

pak platí následující odhad chyby:

$$\min_{\lambda_i \in \sigma(\mathbb{A})} |\hat{\lambda} - \lambda_i| \leq \frac{||\vec{r}||_2}{||\hat{\vec{x}}||_2}. \tag{5.3}$$

Tento odhad říká, že vzdálenost naší aproximace $\hat{\lambda}$ od nejbližšího skutečného vlastního čísla λ_i je omezena normou rezidua.

Důkaz. Nechť $\mathbb{A} \in \mathbb{C}^{n,n}$ je hermitovská. Potom ze Schurovy věty (1.59), resp. jejího důsledku, který mluví o hermitovských maticích, umíme napsat

$$\mathbb{A} = \mathbb{U}^*\mathbb{D}\mathbb{U} \iff \mathbb{A}\mathbb{U}^* = \mathbb{U}^*\mathbb{D}, \tag{5.4}$$

kde $\mathbb{D}$ je diagonální matice s vlastními čísly na diagonále a $\mathbb{U}$ je unitární matice. Pokud sloupce $\mathbb{U}^*$ označíme $\vec{u}^{(1)}, ..., \vec{u}^{(n)}$, máme

$$\mathbb{A}\vec{u}^{(i)} = \mathbb{A}\mathbb{U}^*\vec{e}_i = \mathbb{U}^*\mathbb{D}\vec{e}_i = \lambda_i\vec{u}^{(i)}. \tag{5.5}$$

Vektory $\vec{u}^{(1)}, ..., \vec{u}^{(n)}$ jsou tedy vlastní matice $\mathbb{A}$ a protože jde o sloupce unitární matice, jsou ortonormální (TODO viz VĚTA 1.10). Dále umíme napsat

$$\hat{\vec{x}} = \sum_{i=1}^{n} \alpha_i \vec{u}^{(i)}, \tag{5.6}$$

jelikož soubor n ortonormálních vektorů v prostoru dimenze n tvoří její bázi. Reziduum jsme definovali jako $\vec{r} = \mathbb{A}\hat{\vec{x}} - \hat{\lambda}\hat{\vec{x}}$, a pokud jej převedeme do báze $(\vec{u}^{(1)}, ..., \vec{u}^{(n)})$, dostaneme

$$\vec{r} = \sum_{i=1}^{n} \alpha_i \left(\mathbb{A}\vec{u}^{(i)} - \hat{\lambda}\vec{u}^{(i)}\right) = \sum_{i=1}^{n} \alpha_i \left(\lambda_i - \hat{\lambda}\right)\vec{u}^{(i)}. \tag{5.7}$$

Podívejme se na podíl norem z tvrzení:

$$\frac{||\vec{r}||_2^2}{||\hat{\vec{x}}||_2^2} = \frac{||\sum_{i=1}^{n} \alpha_i \left(\lambda_i - \hat{\lambda}^2\right)\vec{u}^{(i)}||^2}{||\sum_{i=1}^{n} \alpha_i\vec{u}^{(i)}||^2}. \tag{5.8}$$

Protože $\vec{u}^{(i)}$ jsou ortonormální, tedy $||\vec{u}^{(i)}||_2 = 1$. Poté využijeme toho, že norma ortonormálního vektoru[1] je

[1] Zde se nám velmi hodí hermitovskost $\mathbb{A}$, která nám umožnila pracovat s ortonormálními bázemi. Pokud bychom neměli ortonormalitu, museli bychom výraz složitě odhadovat.

rovna druhé odmocnině[2] z součtu čtverců jeho složek:

$$\frac{||\vec{r}||_2^2}{||\hat{\vec{x}}||_2^2} = \frac{\sum_{i=1}^n |\alpha_i|^2 |\lambda_i - \hat{\lambda}|^2}{\sum_{i=1}^n |\alpha_i|^2} = \sum_{i=1}^n \beta_i |\lambda_i - \hat{\lambda}|^2 \tag{5.9}$$

kde $\beta_i = \frac{|\alpha_i|^2}{\sum_{j=1}^n |\alpha_j|^2}$ a tedy $\sum_{i=1}^n \beta_i = 1$. Výraz můžeme odhadnout ze spodu:

$$\frac{||\vec{r}||_2^2}{||\hat{\vec{x}}||_2^2} = \sum_{i=1}^n \beta_i |\lambda_i - \hat{\lambda}|^2 \geq \min_{\lambda_i \in \sigma(\mathbb{A})} |\lambda_i - \hat{\lambda}|^2 \sum_{i=1}^n \beta_i = \min_{\lambda_i \in \sigma(\mathbb{A})} |\lambda_i - \hat{\lambda}|^2. \tag{5.10}$$

Po odmocnění dostáváme tvrzení věty. □

5.2.2 Otázky

- Motivace hledání vlastních čísel (obecně)
- Hlavní rozdíl metod pro řešení soustav lineárních rovnic a metody pro výpočet vlastních čísel

5.3 Částečný problém vlastních čísel

Výpočet kompletního spektra matice je, jak jsme si ukázali, výpočetně velmi náročná úloha. V mnoha praktických aplikacích nám však postačí nalézt pouze jedno nebo několik málo specifických vlastních čísel – například to, které je v absolutní hodnotě největší, nebo naopak nejmenší. Tento úkol nazýváme **částečným problémem vlastních čísel**. Základní metodou pro řešení tohoto problému je mocninná metoda.

Poznámka 5.5 (Terminologie)**.** V kontextu mocninné metody budeme pro jednoduchost používat následující zjednodušenou terminologii:

- **Největší vlastní číslo** bude označovat to vlastní číslo, které je v absolutní hodnotě největší. Budeme předpokládat, že je jediné.
- **Největší vlastní vektor** bude označovat libovolný vlastní vektor příslušející k v absolutní hodnotě největšímu vlastnímu číslu.

5.3.1 Mocninná metoda

5.3.1.1 Odvození mocninné metody

Základní myšlenku mocninné metody si můžeme ilustrovat na jednoduchém příkladu.

Příklad 5.6. Mějme diagonální matici $\mathbb{A}$ a zvolme libovolný počáteční vektor $\vec{x}^{(0)}$:

$$\mathbb{A} = \begin{pmatrix} 3 & 0 \\ 0 & 2 \end{pmatrix}, \quad \vec{x}^{(0)} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}. \tag{5.11}$$

Vlastní čísla této matice jsou zřejmě $\lambda_1 = 3$ a $\lambda_2 = 2$. Pokud budeme opakovaně aplikovat matici $\mathbb{A}$ na vektor $\vec{x}^{(0)}$, dostaneme posloupnost $\vec{x}^{(k)} = \mathbb{A}^k \vec{x}^{(0)}$. Pro náš konkrétní případ to je:

$$\vec{x}^{(k)} = \begin{pmatrix} 3^k & 0 \\ 0 & 2^k \end{pmatrix} \begin{pmatrix} 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 3^k \\ 2^k \end{pmatrix}. \tag{5.12}$$

Problémem této posloupnosti je, že s rostoucím k její normy rostou do nekonečna. Abychom získali konvergentní posloupnost, můžeme ji v každém kroku normovat, například vydělením největším vlastním číslem v k-té

[2] Odmocninu neuvádíme, protože počítáme rovnou normu na druhou.

mocnině:

$$\lim_{k\to\infty}\frac{1}{3^k}\vec{x}^{(k)} = \lim_{k\to\infty}\frac{1}{3^k}\begin{pmatrix}3^k\\2^k\end{pmatrix} = \lim_{k\to\infty}\begin{pmatrix}1\\(2/3)^k\end{pmatrix} = \begin{pmatrix}1\\0\end{pmatrix}. \tag{5.13}$$

Limitní vektor $(1,0)^T$ je právě vlastní vektor matice $\mathbb{A}$ příslušející k v absolutní hodnotě největšímu vlastnímu číslu $\lambda_1 = 3$. Výpočet k-té mocniny matice $\mathbb{A}^k$ je pro obecné (nediagonální) matice velmi náročný. Místo toho můžeme posloupnost $\vec{x}^{(k)}$ generovat iteračně:

$$\vec{x}^{(k+1)} = \mathbb{A}\vec{x}^{(k)}. \tag{5.14}$$

Tento postup je ekvivalentní, neboť $\vec{x}^{(k)} = \mathbb{A}^k\vec{x}^{(0)}$, ale výpočetně mnohem schůdnější. Zobecněme nyní příklad pro libovolný počáteční vektor $\vec{x}^{(0)} = (x_1, x_2)^T$ za předpokladu, že $x_1 \neq 0$. Pak dostáváme:

$$\lim_{k\to\infty}\frac{1}{3^k}\vec{x}^{(k)} = \lim_{k\to\infty}\frac{1}{3^k}\begin{pmatrix}3^k x_1\\2^k x_2\end{pmatrix} = \begin{pmatrix}x_1\\0\end{pmatrix} = x_1\begin{pmatrix}1\\0\end{pmatrix}. \tag{5.15}$$

Opět jsme dostali násobek největšího vlastního vektoru. Vidíme, že je klíčové, aby počáteční vektor $\vec{x}^{(0)}$ měl nenulový průmět do směru největšího vlastního vektoru (v našem případě aby složka x_1 byla nenulová). Pokud by byla nulová, metoda by neměla šanci tento směr "objevit"a konvergovala by k vlastnímu vektoru příslušejícímu k druhému největšímu vlastnímu číslu. Dělení pomocí $\lambda_1^k = 3^k$ jsme prováděli, aby nám vznikající posloupnost konvergovala. Normování pomocí λ_1^k není v praxi ale možné, protože λ_1 neznáme – je to právě to, co hledáme. Z toho důvodu se v každém kroku iterace provádí normalizace vektoru, například vydělením jeho největší složkou v absolutní hodnotě nebo jeho eukleidovskou normou. Tento postup zajišťuje numerickou stabilitu (velikost vektoru neroste do nekonečna) a vede nás k formální definici mocninné metody.

Definice 5.7 (Mocninná metoda)**.** Mocninná metoda konstruuje posloupnosti vektorů a skalárů. Pro daný počáteční vektor $\vec{x}^{(0)}$ s normou $||\vec{x}^{(0)}|| = 1$ pro $k = 0, 1, 2, \ldots$ počítáme:

1. $\vec{y}^{(k+1)} = \mathbb{A}\vec{x}^{(k)}$
2. $l_{k+1} = \text{argmax}_{i=1,\ldots,n}|y_i^{(k+1)}|$ (index největší složky vektoru $\vec{y}^{(k+1)}$)
3. $\rho_{k+1} = y_{l_{k+1}}^{(k+1)}$, (největší složka vektoru $\vec{y}^{(k+1)}$)
4. $\vec{x}^{(k+1)} = \frac{1}{\rho_{k+1}}\vec{y}^{(k+1)}$ (normovaný vektor)

Za vhodných předpokladů pak platí, že posloupnost ρ_k konverguje k největšímu vlastnímu číslu a posloupnost $\vec{x}^{(k)}$ konverguje k příslušnému vlastnímu vektoru.

Poznámka 5.8. Měli bychom se také zamyslet nad tím, zda nemůže dojít k dělení nulou. Pokud ale $\rho_{k+1} = 0$, pak je největší složka vektoru $\vec{y}^{(k+1)}$ nulová, což znamená, že celý vektor je nulový. To by znamenalo, že buď počáteční vektor $\vec{x}^{(0)}$ byl nulový, nebo že matice $\mathbb{A}$ je singulární. V následujícím textu budeme předpokládat, že počáteční vektor $\vec{x}^{(0)}$ nulový není, situaci se singulární maticí $\mathbb{A}$ probereme později.

5.3.1.2 Ukázkové příklady průběhu mocninné metody

Příklad 5.9. Uvažujme opět matici $\mathbb{A} = \begin{pmatrix}3 & 0\\0 & 2\end{pmatrix}$ s počátečním vektorem $\vec{x}^{(0)} = (1,1)^T$ a ukažme si, jak probíhá formalizovaná metoda. Průběh několika prvních iterací je shrnut v tabulce:

k	$\vec{y}^{(k+1)}$	l_{k+1}	ρ_{k+1}	$\vec{x}^{(k+1)}$
0	$(3,2)^T$	1	3	$(1,2/3)^T$
1	$(3,4/3)^T$	1	3	$(1,4/9)^T$
2	$(3,8/9)^T$	1	3	$(1,8/27)^T$
⋮	⋮	⋮	⋮	⋮
i	$(3,2\cdot(2/3)^i)^T$	1	3	$(1,(2/3)^{i+1})^T$

Vidíme, že posloupnost ρ_k okamžitě nalezla hodnotu 3, což je největší vlastní číslo. Posloupnost vektorů $\vec{x}^{(k)}$

zjevně konverguje k vektoru $(1,0)^T$, což je příslušný vlastní vektor.

Příklad 5.10. Změňme nyní pouze počáteční vektor na $\vec{x}^{(0)} = (0,1)^T$. Iterace pak vypadají následovně:

k	$\vec{y}^{(k+1)}$	l_{k+1}	ρ_{k+1}	$\vec{x}^{(k+1)}$
0	$(0,2)^T$	2	2	$(0,1)^T$
1	$(0,2)^T$	2	2	$(0,1)^T$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
i	$(0,2)^T$	2	2	$(0,1)^T$

V tomto případě metoda generuje konstantní posloupnost, která konverguje k vlastnímu číslu $\lambda_2 = 2$ a příslušnému vlastnímu vektoru $(0,1)^T$. Důvodem je, že jsme za počáteční vektor zvolili přímo jiný vlastní vektor matice. Klíčové je, že tento vektor má nulový průmět do směru "největšího"vlastního vektoru $(1,0)^T$. Metoda tak nemá šanci tento dominantní směr objevit a zesílit, a proto konverguje k dalšímu dominantnímu směru, který byl v počátečním vektoru obsažen.

Příklad 5.11. Uvažujme nyní počáteční vektor $\vec{x}^{(0)} = (\epsilon,1)^T$, kde ϵ je velmi malé kladné číslo. Tento případ simuluje situaci, kdy je průmět do směru největšího vlastního vektoru nepatrný. Průběh iterací je zachycen v tabulce:

k	$\vec{y}^{(k+1)}$	l_{k+1}	ρ_{k+1}	$\vec{x}^{(k+1)}$
0	$(3\epsilon,2)^T$	2	2	$(\frac{3}{2}\epsilon,1)^T$
1	$(\frac{9}{2}\epsilon,2)^T$	2	2	$(\frac{9}{4}\epsilon,1)^T$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$i-1$	$(\frac{3^i}{2^{i-1}}\epsilon,2)^T$	2	2	$(\frac{3^i}{2^i}\epsilon,1)^T$
i	$(\frac{3^{i+1}}{2^i}\epsilon,2)^T$	1	$\frac{3^{i+1}}{2^i}\epsilon$	$(1,\frac{2^{i+1}}{3^{i+1}\epsilon})^T$
$i+1$	$(3,\frac{2^{i+2}}{3^{i+1}\epsilon})^T$	1	3	$(1,\frac{2^{i+2}}{3^{i+2}\epsilon})^T$

Zde index i označuje iteraci, ve které poprvé platí, že první složka vektoru $\vec{y}$ je v absolutní hodnotě větší než druhá, tj. $\frac{3^{i+1}}{2^i}\epsilon > 2$. Zpočátku se zdá, že metoda konverguje k vlastnímu číslu 2, protože druhá složka vektoru je dominantní. Avšak malá, ale nenulová složka ϵ je v každém kroku násobena faktorem 3, zatímco druhá složka faktorem 2. Dříve či později tak první složka převáží, dojde ke "zlomu"a metoda začne správně konvergovat k největšímu vlastnímu číslu 3 a příslušnému vlastnímu vektoru.

Poznámka 5.12. Ukázali jsme, že důležitým předpokladem pro funkčnost mocninné metody je, aby počáteční vektor $\vec{x}^{(0)}$ měl nenulový průmět do směru největšího vlastního vektoru. Z předchozího příkladu je ale zřejmé, že metoda je poměrně robustní vůči volbě počátečního vektoru.

- Vidíme, že stačí, aby průmět $\vec{x}^{(0)}$ do směru největšího vlastního vektoru byl i velice malý.
- Toto v praxi u reálných úloh zajistí zaokrouhlovací chyby, které téměř vždy vnesou do výpočtu nepatrnou složku ve směru dominantního vlastního vektoru.
- Navíc, pokud budeme $\vec{x}^{(0)}$ volit náhodně, pak je nulová pravděpodobnost[3] trefit vektor s nulovým průmětem do směru největšího vlastního vektoru.

5.3.1.3 Konvergence mocninné metody pro hermitovské matice

Nyní se pokusme tyto úvahy o konvergenci formalizovat. Začněme situací, kdy je matice $\mathbb{A}$ je hermitovská. Takovou matici lze zapsat pomocí spektrálního rozkladu $\mathbb{A} = \mathbb{U}\mathbb{D}\mathbb{U}^*$, který plyne z důsledků Schurovy věty (1.59), kde $\mathbb{U}$ je unitární matice, jejíž sloupce $\vec{u}^{(1)},\dots,\vec{u}^{(n)}$ tvoří ortonormální bázi vlastních vektorů, a $\mathbb{D}$ je diagonální matice s odpovídajícími reálnými vlastními čísly $\lambda_1,\dots,\lambda_n$ na diagonále. (Detailnější rozbor pravdivosti těchto tvrzení je popsán v důkazu (5.4).) Předpokládejme, že vlastní čísla jsou uspořádána tak, že $|\lambda_1| > |\lambda_2| \geq \cdots \geq |\lambda_n|$, tzn. největší vlastní číslo je λ_1 a odpovídající vlastní vektor je $\vec{u}^{(1)}$. Počáteční vektor

[3] To si můžeme představit např. v $\mathbb{R}^2$, kde se pohybovaly i naše příklady. Cílem bylo se vyvarovat volbě $\vec{x}_1^{(0)} = 0$, což znamená vyvarovat se první souřadné ose. Víme ale, že osa představuje v rámci plochy množinu míry nula, a tedy při náhodné volbě zanedbatelnou plochu.

$\vec{x}^{(0)}$ můžeme vyjádřit jako lineární kombinaci vlastních vektorů: $\vec{x}^{(0)} = \sum_{i=1}^{n} \alpha_i \vec{u}^{(i)}$. V bázi vlastních vektorů má tedy souřadnice $\vec{v} = \mathbb{U}^* \vec{x}^{(0)} = (\alpha_1, \ldots, \alpha_n)^T$. Aplikace k-té mocniny matice $\mathbb{A}$ pak odpovídá:

$$\mathbb{A}^k \vec{x}^{(0)} = \mathbb{U}\mathbb{D}^k \mathbb{U}^* \vec{x}^{(0)} = \mathbb{U}\mathbb{D}^k \vec{v}. \tag{5.16}$$

Tento výraz rozepsaný po složkách v bázi $\{\vec{u}^{(i)}\}$ ukazuje, jak se jednotlivé složky zesilují:

$$\mathbb{A}^k \vec{x}^{(0)} = \sum_{i=1}^{n} \alpha_i \lambda_i^k \vec{u}^{(i)} = \lambda_1^k \left(\alpha_1 \vec{u}^{(1)} + \sum_{i=2}^{n} \alpha_i \left(\frac{\lambda_i}{\lambda_1} \right)^k \vec{u}^{(i)} \right). \tag{5.17}$$

Pokud je průmět do směru největšího vlastního vektoru nenulový (tj. $\alpha_1 \neq 0$), pak pro $k \to \infty$ členy $(\lambda_i/\lambda_1)^k$ pro $i > 1$ zanikají a směr vektoru $\mathbb{A}^k \vec{x}^{(0)}$ konverguje ke směru $\vec{u}^{(1)}$. Po normalizaci tedy dostáváme:

$$\lim_{k \to \infty} \frac{\mathbb{A}^k \vec{x}^{(0)}}{||\mathbb{A}^k \vec{x}^{(0)}||} = \pm \vec{u}^{(1)}. \tag{5.18}$$

5.3.1.4 Obecné konvergenční věty

Lemma 5.13. Vektor $\vec{x}^{(k)}$ generovaný mocninnou metodou lze vyjádřit ve tvaru

$$\vec{x}^{(k)} = \frac{1}{\rho_1 \rho_2 \cdots \rho_k} \mathbb{A}^k \vec{x}^{(0)}. \tag{5.19}$$

Důkaz. Tento přepis plyne přímo z definice mocninné metody.

$$\vec{x}^{(k+1)} = \frac{1}{\rho_{k+1}} \vec{y}^{(k+1)} = \frac{1}{\rho_{k+1}} \mathbb{A} \vec{x}^{(k)} \tag{5.20}$$

Nyní tento předpis jen aplikujeme rekruzivně:

$$\vec{x}^{(k)} = \frac{1}{\rho_k} \vec{y}^{(k)} = \frac{1}{\rho_k} \mathbb{A} \vec{x}^{(k-1)} = \frac{1}{\rho_k} \mathbb{A} \left(\frac{1}{\rho_{k-1}} \vec{y}^{(k-1)} \right) = \frac{1}{\rho_k \rho_{k-1}} \mathbb{A}^2 \vec{x}^{(k-2)} = \ldots \frac{1}{\rho_k \rho_{k-1} \cdots \rho_1} \mathbb{A}^k \vec{x}^{(0)}. \tag{5.21}$$

□

Poznámka 5.14. Pokud posloupnost odhadů ρ_k konverguje k největšímu vlastnímu číslu λ_1, pak pro velká k platí, že součin $\rho_1 \rho_2 \ldots \rho_k \sim \lambda_1^k$ (TODO vysvětlit).

Věta 5.15. Nechť matice $\mathbb{A} \in \mathbb{C}^{n,n}$ má jedno v absolutní hodnotě největší vlastní číslo λ_1. Nechť je toto vlastní číslo buď jednonásobné, nebo vícenásobné se shodnou algebraickou a geometrickou násobností r. Nechť regulární matice $\mathbb{X}$ převádí $\mathbb{A}$ do Jordanova kanonického tvaru $\mathbb{A} = \mathbb{X}\mathbb{J}_{\mathbb{A}}\mathbb{X}^{-1}$ tak, že bloky příslušející λ_1 jsou na začátku. Pak pro libovolný počáteční vektor $\vec{x}^{(0)}$, který má nenulový průmět do podprostoru generovaného vlastními vektory k λ_1, platí, že posloupnost ρ_k z mocninné metody konverguje k λ_1 a posloupnost vektorů $\vec{x}^{(k)}$ konverguje k příslušnému vlastnímu vektoru.

Důkaz. Už ve tvrzení předpokládáme, že největší vlastní číslo se nachází na prvním místě. Budeme také předpokládat, že $\forall k \in \hat{n}$ je $l_k = 1$. Tvrzení by šlo zobecnit, nicméně důkaz by se stížil jen o technické detaily, proto se omezíme na tento případ. V definici metody jsme označili ρ_k největší složku vektoru $\vec{y}^{(k)}$. Protože jsme si výše řekli, že uvažujeme případ, kdy $l_k = 1$, platí $\rho_k = (\vec{e}^{(1)})^T y^{(k)}$. Potom je zřejmé, že

$$x_1^{(k)} = (\vec{e}^{(1)})^T \vec{x}^{(k)} = \frac{(\vec{e}^{(1)})^T y^{(k)}}{\rho_k} = 1. \tag{5.22}$$

Potom můžeme zapsat:

$$\rho_k = \frac{(\vec{e}^{(1)})^T y^{(k)}}{(\vec{e}^{(1)})^T \vec{x}^{(k)}}, \tag{5.23}$$

protože dělení jedničkou nezmění hodnotu ρ_k. Dále dosadíme za $y^{(k)}$ z definice metody:

$$\rho_k = \frac{(\vec{e}^{(1)})^T \mathbb{A}\vec{x}^{(k)}}{(\vec{e}^{(1)})^T \vec{x}^{(k)}}, \tag{5.24}$$

a aplikujeme lemma (5.13):

$$\rho_k = \frac{(\vec{e}^{(1)})^T \mathbb{A}\left(\frac{1}{\rho_k \dots \rho_1}\mathbb{A}^k \vec{x}^{(0)}\right)}{(\vec{e}^{(1)})^T \left(\frac{1}{\rho_k \dots \rho_1}\mathbb{A}^k \vec{x}^{(0)}\right)} = \frac{(\vec{e}^{(1)})^T \mathbb{A}^{k+1} \vec{x}^{(0)}}{(\vec{e}^{(1)})^T \mathbb{A}^k \vec{x}^{(0)}}. \tag{5.25}$$

Předpoklady nám říkají, že umíme zapsat $\mathbb{A}$ v Jordanově kanonickém tvaru $\mathbb{A} = \mathbb{X}\mathbb{J}_{\mathbb{A}}\mathbb{X}^{-1}$, kde $\mathbb{J}_{\mathbb{A}}$ je Jordanova matice. Potom

$$\rho_k = \frac{(\vec{e}^{(1)})^T \mathbb{X}\mathbb{J}_{\mathbb{A}}^{k+1}\mathbb{X}^{-1}\vec{x}^{(0)}}{(\vec{e}^{(1)})^T \mathbb{X}\mathbb{J}_{\mathbb{A}}^{k}\mathbb{X}^{-1}\vec{x}^{(0)}}. \tag{5.26}$$

Vzpomeňme si, že protože vlastní číslo λ_1 je buď jednonásobné, nebo vícenásobné s shodnou algebraickou a geometrickou násobností r, prvních r řádků[4] matice $\mathbb{J}_{\mathbb{A}}$ je čistě diagonální[5] s λ_1 na diagonále.

$$\rho_k = \frac{(\vec{e}^{(1)})^T \mathbb{X} \begin{pmatrix} \lambda_1^{k+1} & & & & & \\ & \ddots & & & & \\ & & \lambda_1^{k+1} & & & \\ & & & \mathbb{J}_2^{k+1} & & \\ & & & & \ddots & \\ & & & & & \mathbb{J}_s^{k+1} \end{pmatrix} \mathbb{X}^{-1}\vec{x}^{(0)}}{(\vec{e}^{(1)})^T \mathbb{X} \begin{pmatrix} \lambda_1^{k} & & & & & \\ & \ddots & & & & \\ & & \lambda_1^{k} & & & \\ & & & \mathbb{J}_2^{k} & & \\ & & & & \ddots & \\ & & & & & \mathbb{J}_s^{k} \end{pmatrix} \mathbb{X}^{-1}\vec{x}^{(0)}} \tag{5.27}$$

$$\rho_k == \frac{\lambda_1^{k+1}(\vec{e}^{(1)})^T \mathbb{X} \begin{pmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & & & \\ & & & \left(\frac{1}{\lambda_1}\mathbb{J}_2\right)^{k+1} & & \\ & & & & \ddots & \\ & & & & & \left(\frac{1}{\lambda_1}\mathbb{J}_s\right)^{k+1} \end{pmatrix} \mathbb{X}^{-1}\vec{x}^{(0)}}{\lambda_1^{k}(\vec{e}^{(1)})^T \mathbb{X} \begin{pmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & & & \\ & & & \left(\frac{1}{\lambda_1}\mathbb{J}_2\right)^{k} & & \\ & & & & \ddots & \\ & & & & & \left(\frac{1}{\lambda_1}\mathbb{J}_s\right)^{k} \end{pmatrix} \mathbb{X}^{-1}\vec{x}^{(0)}} \tag{5.28}$$

Nyní si uvědomme, že v blocích $\frac{1}{\lambda_1}\mathbb{J}_i$ kde $i \neq 1$ jsou na diagonále čísla ostře menší než 1, protože λ_1 je největší vlastní číslo a na diagonále jsou podíly $\frac{\lambda_i}{\lambda_1}$. Proto spektrální poloměr $\rho\left(\frac{1}{\lambda_1}\mathbb{J}_i\right) = \frac{\lambda_i}{\lambda_1} < 1$ a tedy $\left(\frac{1}{\lambda_1}\mathbb{J}_i\right)^k \to \mathbb{O}$

[4] může být i jen jeden řádek, pokud λ_1 je jednonásobné

[5] nad diagonálou nejsou žádné jedničky, jako bychom je tam měli v případě odlišné geometrické a algebraické násobnosti

pro $k \to +\infty$.

$$\rho_k \to \frac{\lambda_1 \left(\vec{e}^{(1)}\right)^T \mathbb{X}^{-1} \begin{pmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & & & \\ & & & 0 & & \\ & & & & \ddots & \\ & & & & & 0 \end{pmatrix} \mathbb{X}\vec{x}^{(0)}}{\left(\vec{e}^{(1)}\right)^T \mathbb{X}^{-1} \begin{pmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & & & \\ & & & 0 & & \\ & & & & \ddots & \\ & & & & & 0 \end{pmatrix} \mathbb{X}\vec{x}^{(0)}} \tag{5.29}$$

My jsme předpokládali, že počáteční vektor $\vec{x}^{(0)}$ má nenulový průmět do podprostoru generovaného vlastními vektory k λ_1. To znamená, že pokud označíme

$$\vec{u} = \begin{pmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & & & \\ & & & 0 & & \\ & & & & \ddots & \\ & & & & & 0 \end{pmatrix} \mathbb{X}\vec{x}^{(0)}, \tag{5.30}$$

tak $\vec{u} \neq 0$ (TODO doplnit proč). Dále je zřejmé, že součin $\left(\vec{e}^{(1)}\right)^T \neq 0$, protože $\mathbb{X}$ je regulární. Proto můžeme v podílu zkrátit $\vec{u}$ a máme $\rho_k \to \lambda_1$. Nyní zbývá dokázat konvergenci vektorů $\vec{x}^{(k)}$ k příslušnému vlastnímu vektoru. Použijeme opět vyjádření ρ_k přes první složku vektoru $\vec{y}^{(k)}$ a dále lemma (5.13):

$$\vec{x}^{(k)} = \frac{1}{\left(\vec{e}^{(1)}\right)^T \mathbb{A}\vec{x}^{(k)}} \mathbb{A}\vec{x}^{(k)} = \frac{\frac{1}{\rho_k \dots \rho_1} \mathbb{A}^{k+1}\vec{x}^{(0)}}{\frac{1}{\rho_k \dots \rho_1} \left(\vec{e}^{(1)}\right)^T \mathbb{A}^k \vec{x}^{(0)}}. \tag{5.31}$$

I dále budeme postupovat obdobně rozepsáním matice $\mathbb{A}$ v Jordanově kanonickém tvaru:

$$\vec{x}^{(k)} = \frac{\mathbb{A}^{k+1}\vec{x}^{(0)}}{\left(\vec{e}^{(1)}\right)^T \mathbb{A}^k \vec{x}^{(0)}} = \frac{\mathbb{X} \begin{pmatrix} \lambda_1^{k+1} & & & & & \\ & \ddots & & & & \\ & & \lambda_1^{k+1} & & & \\ & & & \mathbb{J}_2^{k+1} & & \\ & & & & \ddots & \\ & & & & & \mathbb{J}_s^{k+1} \end{pmatrix} \mathbb{X}^{-1}\vec{x}^{(0)}}{\left(\vec{e}^{(1)}\right) \mathbb{X} \begin{pmatrix} \lambda_1^{k+1} & & & & & \\ & \ddots & & & & \\ & & \lambda_1^{k+1} & & & \\ & & & \mathbb{J}_2^{k+1} & & \\ & & & & \ddots & \\ & & & & & \mathbb{J}_s^{k+1} \end{pmatrix} \mathbb{X}^{-1}\vec{x}^{(0)}} \tag{5.32}$$

a dále stejně jako výše dojdeme do stavu po vytknutí λ_1^{k+1}:

$$\vec{x}^{(k)} \to \frac{\mathbb{X}\begin{pmatrix}1 & & & & & \\ & \ddots & & & & \\ & & 1 & & & \\ & & & 0 & & \\ & & & & \ddots & \\ & & & & & 0\end{pmatrix}\mathbb{X}^{-1}\vec{x}^{(0)}}{\left(\vec{e}^{(1)}\right)\mathbb{X}\begin{pmatrix}1 & & & & & \\ & \ddots & & & & \\ & & 1 & & & \\ & & & 0 & & \\ & & & & \ddots & \\ & & & & & 0\end{pmatrix}\mathbb{X}^{-1}\vec{x}^{(0)}} \tag{5.33}$$

Opět stejným argumentem vektor v čitateli je jistě nenulový. Ve jmenovateli potřebujeme, aby byla nenulová první složka, nicméně její nulovost neumíme ovlivnit. Vektor jako celek ale nenulový být musí, a tak v případě, že by první složka byla nulová, zopakujeme celý postup pro jiný vektor standardní báze. Zbývá ještě dokázat, že limitní vektor $\vec{x}^{(k)} \to \vec{x}$ je skutečně vlastní vektor příslušný vlastnímu číslu λ_1. Z předpisu metody ale víme, že

$$\vec{x}^{(k+1)} = \frac{1}{\rho_{k+1}}\mathbb{A}\vec{x}^{(k)}, \tag{5.34}$$

a když přejdeme limitně $k \to +\infty$, potom

$$\vec{x} = \frac{1}{\lambda_1}\mathbb{A}\vec{x} \iff \mathbb{A}\vec{x} = \lambda_1\vec{x}. \tag{5.35}$$

□

Věta 5.16. Nechť matice $\mathbb{A} \in \mathbb{C}^{n,n}$ má dvě v absolutní hodnotě největší vlastní čísla λ_1 a $-\lambda_1$. Nechť $\vec{x}^{(0)}$ je takový, že jeho průmět do podprostoru generovaného příslušnými vlastními vektory je nenulový. Pak v mocninné metodě konverguje posloupnost $\sqrt{\rho_{2k}\rho_{2k+1}}$ k absolutní hodnotě největších vlastních čísel $|\lambda_1|$.

Důkaz. Nepřednáší se. □

5.3.1.5 Praktické aspekty a varianty

Poznámka 5.17 (Odhad vlastního čísla)**.** V příkladu 5.11 jsme ukázali, že pokud se index l_k (pozice největšího prvku v absolutní hodnotě) po několika iteracích ustálí, pak posloupnost ρ_k dobře aproximuje největší vlastní číslo λ_1. Můžeme si to ukázat na příkladu, kdy od nějakého k je $l_k = 1$ (to bude častý výsledek různých metod, kdy budeme předpokládat, že největší vlastní číslo je to první). Potom z definice metody je

$$\rho_{k+1}\vec{x}^{(k+1)} = \vec{y}^{(k+1)} = \mathbb{A}\vec{x}^{(k)}, \tag{5.36}$$

a tedy ρ_{k+1} aproximuje největší vlastní číslo λ_1. Pokud by se l_k měnil, lze odhad získat z definice vlastního vektoru. Pro aproximaci $\vec{x}^{(k)}$ platí

$$\mathbb{A}\vec{x}^{(k)} = \vec{y}^{(k+1)} \approx \lambda_1\vec{x}^{(k)}. \tag{5.37}$$

Odtud pro každou složku i dostáváme odhad $\lambda_1 \approx y_i^{(k+1)}/x_i^{(k)}$. Jako celkový odhad lze vzít průměr těchto hodnot nebo hodnotu z jedné vybrané složky.

Poznámka 5.18 (Krylovova posloupnost a posun spektra)**.** Samotný směr největšího vlastního vektoru je dán tzv. Krylovovou posloupností $\vec{x}^{(0)}, \mathbb{A}\vec{x}^{(0)}, \mathbb{A}^2\vec{x}^{(0)}, \ldots$. Dělení číslem ρ_k v každém kroku slouží primárně

k zajištění numerické stability. Rychlost konvergence metody závisí na podílu $|\lambda_2/\lambda_1|$. Konvergenci můžeme urychlit vhodným posunem spektra o číslo λ^*, tj. aplikací metody na matici $\mathbb{A} - \lambda^*\mathbb{I}$. Tím lze zmenšit podíl $|\frac{\lambda_2-\lambda^*}{\lambda_1-\lambda^*}|$. K výslednému vlastnímu číslu je pak nutné přičíst λ^* zpět.

Poznámka 5.19 (Inverzní mocninná metoda)**.** Pokud chceme nalézt v absolutní hodnotě nejmenší vlastní číslo matice $\mathbb{A}$, můžeme použít mocninnou metodu na matici $\mathbb{A}^{-1}$. Největší vlastní číslo matice $\mathbb{A}^{-1}$ je rovno $1/\lambda_{\min}$, kde $\lambda_{\min}$ je nejmenší vlastní číslo $\mathbb{A}$. Výsledné vlastní číslo matice $\mathbb{A}$ je tedy $1/\rho$. Pro nalezení vlastního čísla nejblíže zvolenému číslu λ' aplikujeme mocninnou metodu na matici $(\mathbb{A} - \lambda'\mathbb{I})^{-1}$. Hledané vlastní číslo pak získáme jako $\frac{1}{\rho} + \lambda'$.

Poznámka 5.20 (Implementace inverzních variant)**.** Při aplikaci inverzní mocninné metody není nutné explicitně počítat inverzi matice. Krok $\vec{y}^{(k+1)} = \mathbb{A}^{-1}\vec{x}^{(k)}$ nahradíme řešením soustavy lineárních rovnic $\mathbb{A}\vec{y}^{(k+1)} = \vec{x}^{(k)}$. Pro řešení této soustavy lze s výhodou použít iterační metody, neboť pro velká k se vektory $\vec{y}^{(k)}$ a $\vec{y}^{(k+1)}$ liší jen málo, a $\vec{y}^{(k)}$ tak slouží jako výborná počáteční aproximace pro výpočet $\vec{y}^{(k+1)}$. Tento přístup je v numerické matematice velmi častý.

5.3.2 Redukční metoda (Deflace)

Mocninná metoda nám umožňuje nalézt jedno, v absolutní hodnotě největší vlastní číslo. Pokud bychom chtěli nalézt i další vlastní čísla (např. druhé největší), můžeme použít tzv. redukční metodu, známou také jako metoda deflace. Cílem je zkonstruovat novou matici $\mathbb{B}$ o řád menší, která má stejné spektrum jako původní matice $\mathbb{A}$, pouze s odstraněným již nalezeným vlastním číslem λ_1. Tento postup se ovšem nehodí pro výpočet kompletního spektra matice, protože s každým krokem deflace dochází ke ztrátě přesnosti.

Poznámka 5.21 (Princip Wielandtovy deflace)**.** Nechť λ_1 je nalezené vlastní číslo matice $\mathbb{A} \in \mathbb{C}^{n,n}$ a $\vec{x}$ je k němu příslušný vlastní vektor. Myšlenka metody spočívá v nalezení takové podobnostní transformace, která matici $\mathbb{A}$ převede na blokový tvar, z něhož lze matici $\mathbb{B}$ snadno identifikovat. Provedeme přechod do nové báze, tvořené vektory $\{\vec{x}, \vec{e}_2, \ldots, \vec{e}_n\}$ (za předpokladu, že první složka x_1 vektoru $\vec{x}$ je nenulová). Matice přechodu $\mathbb{P}$ má tyto vektory ve sloupcích. V této nové bázi má matice $\mathbb{A}$ tvar:

$$\mathbb{P}^{-1}\mathbb{A}\mathbb{P} = \begin{pmatrix} \lambda_1 & \vec{q}^T \\ \vec{0} & \mathbb{B} \end{pmatrix}. \tag{5.38}$$

Matice přechodu $\mathbb{P}$ a její inverze $\mathbb{P}^{-1}$ mají následující tvar:

$$\mathbb{P} = \begin{pmatrix} x_1 & 0 & \ldots & 0 \\ x_2 & 1 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ x_n & 0 & \ldots & 1 \end{pmatrix}, \quad \mathbb{P}^{-1} = \begin{pmatrix} 1/x_1 & 0 & \ldots & 0 \\ -x_2/x_1 & 1 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ -x_n/x_1 & 0 & \ldots & 1 \end{pmatrix}. \tag{5.39}$$

Inverzní matici $\mathbb{P}^{-1}$ není třeba složitě počítat; jedná se o matici, která provádí Gaussovu eliminaci v prvním sloupci. Roznásobením $\mathbb{P}^{-1}\mathbb{A}\mathbb{P}$ získáme explicitní tvar matice $\mathbb{B}$ a vektoru $\vec{q}^T$. Vlastní čísla horní blokově trojúhelníkové matice $\mathbb{P}^{-1}\mathbb{A}\mathbb{P}$ jsou λ_1 a vlastní čísla matice $\mathbb{B}$. Pokud nyní nalezneme vlastní číslo λ_2 a příslušný vlastní vektor $\vec{z}$ matice $\mathbb{B}$ (pomocí mocninné metody), můžeme zrekonstruovat odpovídající vlastní vektor $\vec{y}$ původní matice $\mathbb{A}$. Hledáme ho jako lineární kombinaci bázových vektorů, tj. ve tvaru $\vec{y} = \mathbb{P}\begin{pmatrix} z_1 \\ \vec{z} \end{pmatrix}$, kde z_1 musíme dopočítat. Pokud by bylo $\lambda_1 = \lambda_2$, pak platí $\mathbb{B}\vec{z} = \lambda_2\vec{z} = \lambda_1\vec{z}$ a

$$\begin{pmatrix} \lambda_1 & \vec{q}^T \\ \vec{0} & \mathbb{B} \end{pmatrix}\begin{pmatrix} z_1 \\ \vec{z} \end{pmatrix} = \begin{pmatrix} \lambda_1 z_1 + \vec{q}^T\vec{z} \\ \mathbb{B}\vec{z} \end{pmatrix} = \begin{pmatrix} \lambda_1 z_1 \\ \lambda_1\vec{z} \end{pmatrix}. \tag{5.40}$$

Zřejmě tedy $\lambda_1 z_1 + \vec{q}^T\vec{z} = \lambda_1 z_1$ a proto $\vec{q}^T\vec{z} = 0$. z_1 tedy volím libovolně. (TODO rozepsat více, že to souvisí s

geom násobností) Pokud naopak $\lambda_1 \neq \lambda_2$,

$$\begin{pmatrix} \lambda_1 & \vec{q}^T \\ \vec{0} & \mathbb{B} \end{pmatrix} \begin{pmatrix} z_1 \\ \vec{z} \end{pmatrix} = \begin{pmatrix} \lambda_1 z_1 + \vec{q}^T \vec{z} \\ \mathbb{B}\vec{z} \end{pmatrix} = \begin{pmatrix} \lambda_1 z_1 + \vec{q}^T \vec{z} \\ \lambda_2 \vec{z} \end{pmatrix} \overset{!}{=} \begin{pmatrix} \lambda_2 z_1 \\ \lambda_2 \vec{z} \end{pmatrix}. \tag{5.41}$$

kde požadavek označený ! plyne z $\mathbb{A}\vec{y} \overset{!}{=} \lambda_2 \begin{pmatrix} z_1 \\ \vec{z} \end{pmatrix}$. Potom můžeme vyjádřit neznámou složku z_1 jako:

$$z_1 = \frac{\vec{q}^T \vec{z}}{\lambda_2 - \lambda_1}. \tag{5.42}$$

Vlastní vektor původní matice $\mathbb{A}$ je pak dán vztahem $\vec{y} = z_1\vec{x} + (0, z_2, \ldots, z_n)^T$. V případě, že $\lambda_1 = \lambda_2$, musí platit $\vec{q}^T\vec{z} = 0$ a složku z_1 lze volit libovolně. (TODO napsat tohle nahoru k variantě 1=2)

5.3.3 Otázky

- Co je to částečný problém vlastních čísel a jaká metoda se pro jeho řešení primárně používá?
- Jaký je princip mocninné metody a jaké jsou podmínky její konvergence? Jak se tyto podmínky zajistí v praxi?
- Jak lze pomocí mocninné metody nalézt v absolutní hodnotě nejmenší vlastní číslo matice?
- K čemu slouží redukční metoda (deflace) a proč se nehodí pro výpočet kompletního spektra?

5.4 Analýza kompletního spektra matice

5.4.1 Trojúhelníková metoda

První z metod pro výpočet kompletního spektra je trojúhelníková metoda. Jejím cílem je nalézt podobnostní transformaci, která převede původní matici na horní trojúhelníkový tvar, z jehož diagonály pak můžeme snadno odečíst všechna vlastní čísla.

Definice 5.22 (Trojúhelníková metoda)**.** Metoda konstruuje dvě posloupnosti matic: $\{\mathbb{L}^{(k)}\}_{k=1}^{\infty}$ a $\{\mathbb{R}^{(k)}\}_{k=1}^{\infty}$. Postup je následující:

1. Zvolíme libovolnou počáteční matici $\mathbb{L}^{(0)}$ (například $\mathbb{L}^{(0)} = \mathbb{I}$).
2. Pro $k = 0, 1, 2, \ldots$ rekurzivně počítáme:
   - Provedeme LR rozklad[6] matice $\mathbb{A}\mathbb{L}^{(k)}$.
   - Tím získáme další členy posloupností $\mathbb{L}^{(k+1)}$ a $\mathbb{R}^{(k+1)}$ tak, že platí:

$$\mathbb{L}^{(k+1)}\mathbb{R}^{(k+1)} = \mathbb{A}\mathbb{L}^{(k)}, \tag{5.43}$$

     kde $\mathbb{L}^{(k+1)}$ je dolní trojúhelníková matice s jedničkami na diagonále a $\mathbb{R}^{(k+1)}$ je horní trojúhelníková matice.

Poznámka 5.23 (Konvergence a podobnostní transformace)**.** Pokud posloupnosti matic konvergují, tj. $\mathbb{L}^{(k)} \to \mathbb{L}$ a $\mathbb{R}^{(k)} \to \mathbb{R}$, pak v limitě dostáváme rovnici:

$$\mathbb{A}\mathbb{L} = \mathbb{L}\mathbb{R}. \tag{5.44}$$

Protože matice $\mathbb{L}$ je regulární (jakožto limita regulárních matic – dolních trojúhelníkových s jedničkami na

[6] jde o jiný název pro LU rozklad, který se zároveň od naší původní konvence liší tím, že jedničky jsou na diagonále dolní trojúhelníkové matice

diagonále), můžeme rovnici přepsat na tvar:

$$\mathbb{A} = \mathbb{L}\mathbb{R}\mathbb{L}^{-1}. \tag{5.45}$$

Tento vztah ukazuje, že matice $\mathbb{A}$ je podobná horní trojúhelníkové matici $\mathbb{R}$. Vlastní čísla matice $\mathbb{A}$ jsou tedy shodná s vlastními čísly matice $\mathbb{R}$, která můžeme přečíst z její diagonály.

Poznámka 5.24 (Výpočet vlastních vektorů)**.** Jakmile metoda zkonverguje a my známe limitní matice $\mathbb{L}$ a $\mathbb{R}$, můžeme dopočítat vlastní vektory.

1. Nejprve nalezneme vlastní vektory $\vec{y}^i$ matice $\mathbb{R}$ řešením soustavy $(\mathbb{R} - \lambda_i\mathbb{I})\vec{y}^i = \vec{0}$ pro každé vlastní číslo $\lambda_i = r_{ii}$. Jelikož je matice $\mathbb{R} - \lambda_i\mathbb{I}$ horní trojúhelníková s nulou na diagonále, lze její vlastní vektor $\vec{y}^i$ snadno dopočítat zpětnou substitucí.
2. Matice $\mathbb{R}$ je vyjádřením matice $\mathbb{A}$ v bázi tvořené sloupci matice $\mathbb{L}$.

$$\mathbb{A}\mathbb{L}\vec{y}^i = \mathbb{L}\mathbb{R}\vec{y}^i = \mathbb{L}\lambda_i\vec{y}^i = \lambda_i\mathbb{L}\vec{y}^i \implies \mathbb{A}\vec{x}^i = \lambda_i\vec{x}^i. \tag{5.46}$$

   Vlastní vektory $\vec{x}^i$ původní matice $\mathbb{A}$ proto získáme zpětnou transformací:

$$\vec{x}^i = \mathbb{L}\vec{y}^i \tag{5.47}$$

Poznámka 5.25 (Robustnost metody)**.** Počáteční matici $\mathbb{L}^{(0)}$ lze volit libovolně, nemusí být ani dolní trojúhelníková. Díky tomu má metoda samoopravující schopnost. Pokud bychom v některé iteraci napočítali matici $\mathbb{L}^{(k)}$ s chybou, můžeme ji jednoduše považovat za novou startovací matici a pokračovat ve výpočtu.

5.4.1.1 Konvergence a existence rozkladu

Celá metoda je postavena na opakovaném provádění LU rozkladu. Jak víme z kapitoly 2, LU rozklad (resp. LDR rozklad) existuje pouze pro silně regulární matice. Pro první iteraci, kde rozkládáme matici $\mathbb{A}\mathbb{L}^{(0)}$, tedy musíme předpokládat, že je tato matice silně regulární.

Poznámka 5.26 (Spojitá závislost LU rozkladu)**.** Připomeňme si vzorce pro výpočet prvků z kompaktního schématu LU rozkladu:

$$l_{ij} = a_{ij} - \sum_{k=1}^{j-1} l_{ik}u_{kj} \quad \text{pro } j \leq i,$$

$$u_{ij} = \frac{1}{l_{ii}}\left(a_{ij} - \sum_{k=1}^{i-1} l_{ik}u_{kj}\right) \quad \text{pro } i < j.$$

Z těchto vztahů je vidět, že prvky matic $\mathbb{L}$ a $\mathbb{R}$ závisí spojitě na prvcích matice $\mathbb{A}$. To má důležitý důsledek: pokud pro matici $\mathbb{A}$ existuje LU rozklad, bude existovat i pro matici $\mathbb{A} + \mathbb{E}$, pokud bude norma matice $\mathbb{E}$ dostatečně malá.

Tuto myšlenku formálně zachycují následující věty.

Věta 5.27. Nechť matice $\mathbb{A}$ je silně regulární (a tedy existuje její LU rozklad). Pak existuje takové $\epsilon > 0$, že pro libovolnou matici $\mathbb{E}$ splňující $||\mathbb{E}|| < \epsilon$ existuje i LU rozklad matice $\mathbb{A} + \mathbb{E}$.

Věta 5.28. Nechť $\mathbb{A} = \mathbb{I} + \mathbb{E}$, kde $||\mathbb{E}||$ je dostatečně malé. Potom existuje rozklad $\mathbb{A} = \mathbb{L}\mathbb{R}$ a platí, že pokud $||\mathbb{E}|| \to 0$, pak také $\mathbb{L} \to \mathbb{I}$ a $\mathbb{R} \to \mathbb{I}$.

Pro samotný důkaz konvergence trojúhelníkové metody je klíčové následující tvrzení, které dává do souvislosti k-tou iteraci metody a LU rozklad k-té mocniny matice $\mathbb{A}$.

Lemma 5.29. Pokud pro $\mathbb{A}^k\mathbb{L}^{(0)}$ existuje trojúhelníkový rozklad $\mathbb{A}^k\mathbb{L}^{(0)} = \mathcal{L}^{(k)}\mathcal{R}^{(k)}$, pak pro matice z trojúhelníkové metody platí:

$$\begin{aligned} \mathbb{L}^{(k)} &= \mathcal{L}^{(k)}, \\ \mathbb{R}^{(k)}\mathbb{R}^{(k-1)}\ldots\mathbb{R}^{(1)} &= \mathcal{R}^{(k)}. \end{aligned} \tag{5.48}$$

Důkaz. Z předpisu trojúhelníkové metody platí

$$\mathbb{L}^{(k)}\mathbb{R}^{(k)} = \mathbb{A}\mathbb{L}^{(k-1)}. \tag{5.49}$$

Tvrzení věty požaduje, aby

$$\mathcal{L}^{(k)}\mathcal{R}^{(k)} = \mathbb{L}^{(k)}\mathbb{R}^{(k)}\mathbb{R}^{(k-1)}\dots\mathbb{R}^{(1)}. \tag{5.50}$$

V tomto tvaru použijeme iterativně předpis trojúhelníkové metody:

$$\mathcal{L}^{(k)}\mathcal{R}^{(k)} = \mathbb{A}\mathbb{L}^{(k-1)}\mathbb{R}^{(k-2)}\dots\mathbb{R}^{(1)} = \mathbb{A}\mathbb{A}\mathbb{L}^{(k-2)}\mathbb{R}^{(k-3)}\dots\mathbb{R}^{(1)} = \dots = \mathbb{A}^{k-1}\mathbb{L}^{(1)}\mathbb{R}^{(1)}. \tag{5.51}$$

čímž jsme dokázali jeho platnost. □

Poznámka 5.30 (Strategie důkazu konvergence)**.** Z předchozí věty plyne, že ke zkoumání konvergence posloupnosti $\{\mathbb{L}^{(k)}\}$ stačí zkoumat konvergenci LU rozkladu matice $\mathbb{A}^k\mathbb{L}^{(0)}$. Dokážeme-li, že posloupnost $\{\mathcal{L}^{(k)}\}$ konverguje k nějaké matici $\mathbb{L}$, automaticky tím dokážeme i konvergenci $\{\mathbb{L}^{(k)}\} \to \mathbb{L}$. Konvergence posloupnosti $\{\mathbb{R}^{(k)}\}$ pak plyne z rozpisu:

$$\begin{gathered}\mathbb{L}^{(k+1)}\mathbb{R}^{(k+1)} = \mathbb{A}\mathbb{L}^{(k)} \iff \mathbb{R}^{(k+1)} = \left(\mathbb{L}^{(k+1)}\right)^{-1}\mathbb{A}\mathbb{L}^{(k)}, \\ \mathbb{L}\mathbb{R} = \mathbb{A}\mathbb{L} \iff \mathbb{R} = \mathbb{L}^{-1}\mathbb{A}\mathbb{L}.\end{gathered} \tag{5.52}$$

a tedy pro $\mathbb{L}^{(k+1)} \to \mathbb{L}$ máme $\mathbb{R}^{(k+1)} \to \mathbb{L}^{-1}\mathbb{A}\mathbb{L} = \mathbb{R}$.

Věta 5.31 (Konvergence trojúhelníkové metody)**.** Nechť matice $\mathbb{A} \in \mathbb{C}^{n,n}$ je regulární a má všechna vlastní čísla jednonásobná a v absolutní hodnotě navzájem různá, tzn. je diagonalizovatelná, $\mathbb{A} = \mathbb{X}\mathbb{D}\mathbb{X}^{-1}$, kde $|\lambda_1| > |\lambda_2| > \dots > |\lambda_n|$. Dále předpokládejme, že:

- pro dostatečně velké k existují LU rozklady matic $\mathbb{A}\mathbb{L}^{(k)}$,
- existují LU rozklady matic $\mathbb{X}$ a $\mathbb{X}^{-1}\mathbb{L}^{(0)}$.

Pak posloupnosti matic $\{\mathbb{L}^{(k)}\}$ a $\{\mathbb{R}^{(k)}\}$ generované trojúhelníkovou metodou konvergují k maticím $\mathbb{L}$ a $\mathbb{R}$ a na diagonále matice $\mathbb{R}$ je spektrum matice $\mathbb{A}$, seřazené sestupně podle velikosti v absolutní hodnotě.

Důkaz. Podle poznámky nad větou stačí dokázat, existenci LU rozkladu $\mathbb{A}^k\mathbb{L}^{(0)}\mathcal{L}^k\mathcal{R}^k$ že $\mathcal{L}^{(k)} \to \mathbb{L}$. Z předpokladů věty umíme napsat $\mathbb{A} = \mathbb{X}\mathbb{D}\mathbb{X}^{-1}$, kde na diagonále matice $\mathbb{D}$ jsou vlastní čísla matice $\mathbb{A}$. Odtud $\mathbb{A}^k = \mathbb{X}\mathbb{D}^k\mathbb{X}^{-1}$ a proto $\mathbb{A}^k\mathbb{L}^{(0)} = \mathbb{X}\mathbb{D}^k\mathbb{X}^{-1}\mathbb{L}^{(0)}$. Matice $\mathbb{X}$ je regulární (TODO: to nestačí, potřebuji silně reg), a tedy existuje LU rozklad $\mathbb{X} = \mathbb{L}_X\mathbb{R}_X$ a $\mathbb{X}^{-1}\mathbb{L}^{(0)} = \mathbb{L}_Y\mathbb{R}_Y$. Potom

$$\mathbb{A}^k\mathbb{L}^{(0)} = \mathbb{L}_X\mathbb{R}_X\mathbb{D}^k\mathbb{L}^Y\mathbb{R}^Y = \mathbb{L}_X\mathbb{R}_X\mathbb{D}^k\mathbb{L}_Y\mathbb{D}^{-k}\mathbb{D}^k\mathbb{R}_Y. \tag{5.53}$$

Zkoumejme matici $\mathbb{D}^k\mathbb{L}_Y\mathbb{D}^{-k}$. Víme, že jde o matici dolní trojúhelníkovou, tudíž pro $j > i$ je $[\mathbb{D}^k\mathbb{L}_Y\mathbb{D}^{-k}]_{ij} = 0$. Pro $i = j$ je $[\mathbb{D}^k\mathbb{L}_Y\mathbb{D}^{-k}]_{ii} = \lambda_i^k 1 \lambda_i^{-k} = 1$ z toho, že $\mathbb{L}_Y$ má na diagonále jedničky a matice $\mathbb{D}$ vlastní čísla $\mathbb{A}$. Pod diagonálou, tzn. $j < i$, máme $\lambda_i^k l_{ij} \lambda_j^{-k}$. Protože máme vlastní čísla seřazená sestupně, tak pro $j < i$ je $|\lambda_j| > |\lambda_i|$. Proto $\frac{\lambda_i}{\lambda_j} < 1$ a tedy $\left(\frac{\lambda_i}{\lambda_j}\right)^k l_{ij} \to 0$ pro $k \to +\infty$. Celkově proto

$$\mathbb{D}^k\mathbb{L}_Y\mathbb{D}^{-k} = \mathbb{I} + \mathbb{E}^{(k)}, \tag{5.54}$$

kde $\mathbb{E}^{(k)} \to \mathbb{O}$ a $\mathbb{E}^{(k)}$ je dolní trojúhelníková matice s nuly na diagonále. Vraťme se zpět k původní matici $\mathbb{A}^k\mathbb{L}^{(0)}$:

$$\mathbb{A}^k\mathbb{L}^{(0)} = \mathbb{L}_X\mathbb{R}_X(\mathbb{I} + \mathbb{E}^{(k)})\mathbb{D}^k\mathbb{R}_Y = \mathbb{L}_X\mathbb{R}_X(\mathbb{I} + \mathbb{E}^{(k)})\mathbb{R}_X^{-1}\mathbb{R}_X\mathbb{D}^k\mathbb{R}_Y = \mathbb{L}_X(\mathbb{I} + \mathbb{R}_X\mathbb{E}^{(k)}\mathbb{R}_X^{-1})\mathbb{R}_X\mathbb{D}^k\mathbb{R}_Y. \tag{5.55}$$

My víme, že $(\mathbb{I} + \mathbb{R}_X\mathbb{E}^{(k)}\mathbb{R}_X^{-1}) \to \mathbb{I}$ pro $k \to +\infty$. Z věty o existenci LU rozkladu 5.28 tedy plyne, že pro

dostatečně velké k existuje LU rozklad matice $(\mathbb{I} + \mathbb{R}_X\mathbb{E}^{(k)}\mathbb{R}_X^{-1}) = \mathbb{L}_E^{(k)}\mathbb{R}_E^{(k)}$ a $\mathbb{L}_E^{(k)} \to \mathbb{I}$ a $\mathbb{R}_E^{(k)} \to \mathbb{I}$. Můžeme tedy psát, že

$$\mathbb{A}^k\mathbb{L}^{(0)} = \mathbb{L}_X\mathbb{L}_E^{(k)}\mathbb{R}_E^{(k)}\mathbb{R}_X\mathbb{D}^k\mathbb{R}_Y, \tag{5.56}$$

kde $\mathbb{L}_X\mathbb{L}_E^{(k)}$ je dolní trojíhelníková a $\mathbb{L}_X\mathbb{L}_E^{(k)} \to \mathbb{L}_X$. Dále $\mathbb{R}_E^{(k)}\mathbb{R}_X\mathbb{D}^k\mathbb{R}_Y$ je horní trojúhelníková. Tím máme dokázanou konvergenci. Zbývá dokázat, že na diagonále matice $\mathbb{R}$ jsou vlastní čísla matice $\mathbb{A}$ seřazená podle velikosti. Vyjdeme opět z předpisu trojúhelníkové metody: $\mathbb{A}\mathbb{L}^{(k)} = \mathbb{L}^{(k+1)}\mathbb{R}^{(k+1)}$. Také nyní víme, že pro $k \to +\infty$ tento vztah konverguje k $\mathbb{A}\mathbb{L} = \mathbb{L}\mathbb{R}$. Potom $\mathbb{R} = \mathbb{L}^{-1}\mathbb{A}\mathbb{L}$. V důkazu předešlé části, jsme používali označení $\mathbb{L} = \mathbb{L}_X$, když si na něj tedy vzpomeneme, můžeme se také vrátit k rozepsání $\mathbb{X} = \mathbb{L}_X\mathbb{R}_X$. Potom díky $\mathbb{A} = \mathbb{X}\mathbb{D}\mathbb{X}^{-1}$ máme:

$$\mathbb{R} = \mathbb{L}_X^{-1}\mathbb{A}\mathbb{L}_X = \mathbb{L}_X^{-1}\mathbb{X}\mathbb{D}\mathbb{X}^{-1}\mathbb{L}_X = \mathbb{L}_X^{-1}\mathbb{L}_X\mathbb{R}_X\mathbb{D}\mathbb{R}_X^{-1}\mathbb{L}_X^{-1}\mathbb{L}_X = \mathbb{R}_X\mathbb{D}\mathbb{R}_X^{-1}, \tag{5.57}$$

což je horní trojúhelníkové, diagonální a horní trojúhelníkové matice. Na diagonále tohoto součinu je tedy součin diagonálních prvků, přičemž diagonální prvky matic $\mathbb{R}_X$ a $\mathbb{R}_X^{-1}$ se pokrátí, a zbývá diagonála $\mathbb{D}$, což jsou správně seřazená vlastní čísla původní matice $\mathbb{A}$. □

Poznámka 5.32. Konvergencí metody za jiných, slabších předpokladů se zde nebudeme zabývat.

5.4.2 LR algoritmus

LR algoritmus je další metoda pro výpočet kompletního spektra, úzce související s trojúhelníkovou metodou.

Definice 5.33 (LR algoritmus)**.** Metoda konstruuje posloupnost matic $\{\mathbb{A}^{(k)}\}_{k=1}^{\infty}$. Pro $k = 1, 2, \dots$ má iterace následující tvar:

1. Provedeme LU rozklad matice $\mathbb{A}^{(k)}$:
$$\mathbb{A}^{(k)} = \hat{\mathbb{L}}^{(k)}\hat{\mathbb{R}}^{(k)}, \tag{5.58}$$
kde $\hat{\mathbb{L}}^{(k)}$ je dolní trojúhelníková matice s jedničkami na diagonále a $\hat{\mathbb{R}}^{(k)}$ je horní trojúhelníková matice. (Stříšky používáme pro odlišení od matic z trojúhelníkové metody).
2. Následující člen posloupnosti $\mathbb{A}^{(k+1)}$ získáme vynásobením faktorů v opačném pořadí:
$$\mathbb{A}^{(k+1)} = \hat{\mathbb{R}}^{(k)}\hat{\mathbb{L}}^{(k)}. \tag{5.59}$$

Jako počáteční matici volíme $\mathbb{A}^{(0)} = \mathbb{A}$.

Poznámka 5.34 (Podobnostní transformace)**.** Každý krok LR algoritmu představuje podobnostní transformaci, jelikož z definičních vztahů můžeme vyjádřit $\hat{\mathbb{R}}^{(k)} = (\hat{\mathbb{L}}^{(k)})^{-1}\mathbb{A}^{(k)}$ a dosadit do druhého kroku:

$$\mathbb{A}^{(k+1)} = \hat{\mathbb{R}}^{(k)}\hat{\mathbb{L}}^{(k)} = (\hat{\mathbb{L}}^{(k)})^{-1}\mathbb{A}^{(k)}\hat{\mathbb{L}}^{(k)}. \tag{5.60}$$

Z toho plyne, že všechny matice v posloupnosti $\{\mathbb{A}^{(k)}\}$ jsou si navzájem podobné a mají tedy stejná vlastní čísla jako původní matice $\mathbb{A}$. Pokud tato posloupnost konverguje k horní trojúhelníkové matici, na její diagonále budou ležet vlastní čísla matice $\mathbb{A}$. Pro výpočet vlastních vektorů je třeba znát celkovou transformační matici. Dá se ukázat, že platí $\mathbb{A}^{(k)} = (\hat{\mathbb{L}}^{(1)} \dots \hat{\mathbb{L}}^{(k-1)})^{-1}\mathbb{A}(\hat{\mathbb{L}}^{(1)} \dots \hat{\mathbb{L}}^{(k-1)})$. Hledanou maticí, jejíž sloupce tvoří vlastní vektory v nové bázi, je tedy součin $\mathbb{L} = \hat{\mathbb{L}}^{(1)}\hat{\mathbb{L}}^{(2)} \dots$.

Poznámka 5.35 (Srovnání s trojúhelníkovou metodou)**.** .

- LR algoritmus má menší nároky na paměť. V trojúhelníkové metodě potřebujeme uložit matici $\mathbb{A}$, dále $\mathbb{L}^{(0)}$ a nakonec $\mathbb{A}\mathbb{L}^{(0)}$. V každém kroku pak in-place přepočítáme $\mathbb{A}\mathbb{L}^{(k)}$ na $\mathbb{L}^{(k+1)}\mathbb{R}^{(k+1)}$. Poté na pozici $\mathbb{A}\mathbb{L}^{(k)}$ spočítáme $\mathbb{A}\mathbb{L}^{k+1}$ a takto pokračujeme dále. Celkem tedy musíme ukládat tři matice. LR algoritmu stačí jen 2, jelikož nepotřebuje uchovávat původní matici $\mathbb{A}$. Po spuštění algoritmu se původní $\mathbb{A} = \mathbb{A}^{(0)}$ přepíše in-place na $\mathbb{L}^{(1)}\mathbb{R}^{(1)}$. Poté si musíme jinde spočítat $\mathbb{A}^{(1)} = \mathbb{R}^{(1)}\mathbb{L}^{(1)}$. Toto můžeme následně opět

inplace přepsat rozkladem $\mathbb{L}^{(2)}\mathbb{R}^{(2)}$ atd.

- LR algoritmus nemá tak dobrou samoopravující schopnost. V trojúhelníkové matici jsme v každém kroku vraceli informaci o původní matici $\mathbb{A}$, nicméně v LR algoritmu si ji nepamatujeme a v případě chyby v některém kroku nemáme jak se k původní matici vrátit.

5.4.2.1 Konvergence LR algoritmu

Konvergenci LR algoritmu lze elegantně dokázat pomocí jeho vztahu k trojúhelníkové metodě.

Věta 5.36. Existuje-li trojúhelníkový rozklad matice $\mathbb{A}^k = \mathcal{L}^{(k)}\mathcal{R}^{(k)}$, pak pro faktory z LR algoritmu platí:

$$\begin{aligned}\mathcal{L}^{(k)} &= \hat{\mathbb{L}}^{(1)}\hat{\mathbb{L}}^{(2)}\dots\hat{\mathbb{L}}^{(k)},\\ \mathcal{R}^{(k)} &= \hat{\mathbb{R}}^{(k)}\hat{\mathbb{R}}^{(k-1)}\dots\hat{\mathbb{R}}^{(1)}.\end{aligned} \tag{5.61}$$

Důkaz. Vyjdeme z předpisu pro LR metodu, tedy $\mathbb{L}^{\hat{(k)}}\mathbb{R}^{\hat{(k)}} = \mathbb{A}^{(k)}$ a $\mathbb{A}^{(k+1)} = \mathbb{R}^{\hat{(k)}}\mathbb{L}^{\hat{(k)}}$. Stejně jako v obdobné větě pro trojúhelníkovou metodu vyjdeme z toho, co chceme dokázat:

$$\mathcal{L}^{(k)}\mathcal{R}^{(k)} = \hat{\mathbb{L}}^{(1)}\hat{\mathbb{L}}^{(2)}\dots\hat{\mathbb{L}}^{(k-1)}\hat{\mathbb{L}}^{(k)}\hat{\mathbb{R}}^{(k)}\hat{\mathbb{R}}^{(k-1)}\dots\hat{\mathbb{R}}^{(1)}. \tag{5.62}$$

Do něj opět rekurzivně dosadíme předpis LR metody:

$$\begin{aligned}\mathcal{L}^{(k)}\mathcal{R}^{(k)} &= \hat{\mathbb{L}}^{(1)}\hat{\mathbb{L}}^{(2)}\dots\hat{\mathbb{L}}^{(k-1)}\mathbb{A}^{(k)}\hat{\mathbb{R}}^{(k-1)}\hat{\mathbb{R}}^{(k-2)}\dots\hat{\mathbb{R}}^{(1)}\\ &= \hat{\mathbb{L}}^{(1)}\hat{\mathbb{L}}^{(2)}\dots\hat{\mathbb{L}}^{(k-1)}\hat{\mathbb{R}}^{(k-1)}\hat{\mathbb{R}}^{(k-1)}\hat{\mathbb{L}}^{(k-1)}\hat{\mathbb{R}}^{(k-2)}\dots\hat{\mathbb{R}}^{(1)}.\end{aligned} \tag{5.63}$$

$$\begin{aligned}\mathcal{L}^{(k)}\mathcal{R}^{(k)} &= \hat{\mathbb{L}}^{(1)}\dots\hat{\mathbb{L}}^{(k-2)}\mathbb{A}^{(k-1)}\mathbb{A}^{(k-1)}\hat{\mathbb{R}}^{(k-2)}\dots\hat{\mathbb{R}}^{(1)}\\ &= \hat{\mathbb{L}}^{(1)}\dots\hat{\mathbb{L}}^{(k-2)}\hat{\mathbb{R}}^{(k-2)}\hat{\mathbb{L}}^{(k-2)}\hat{\mathbb{R}}^{(k-2)}\hat{\mathbb{L}}^{(k-2)}\hat{\mathbb{R}}^{(k-2)}\dots\hat{\mathbb{R}}^{(1)}.\end{aligned} \tag{5.64}$$

Postupně dojdeme až k

$$\mathcal{L}^{(k)}\mathcal{R}^{(k)} = \mathbb{A}^k, \tag{5.65}$$

což potvrzuje platnost řešení. □

Poznámka 5.37 (Vztah mezi metodami)**.** Pokud v trojúhelníkové metodě zvolíme speciální případ $\mathbb{L}^{(0)} = \mathbb{I}$, pak se analýzy obou metod propojí. Z předchozích vět víme, že $\mathbb{L}^{(k)} = \mathcal{L}^{(k)}$ a $\mathbb{R}^{(k)}\mathbb{R}^{(k-1)}\dots\mathbb{R}^{(1)} = \mathcal{R}^{(k)}$. Zkombinováním těchto poznatků dostáváme přímý vztah mezi maticemi z obou metod:

$$\begin{aligned}\mathbb{L}^{(k)} &= \hat{\mathbb{L}}^{(1)}\hat{\mathbb{L}}^{(2)}\dots\hat{\mathbb{L}}^{(k)},\\ \mathbb{R}^{(k)} &= \hat{\mathbb{R}}^{(k)}.\end{aligned}$$

Poznámka 5.38. Díky tomuto vztahu můžeme ukázat konvergenci posloupnosti $\{\mathbb{A}^{(k)}\}$. Z předchozí poznámky plyne:

$$\mathbb{L}^{(k)} = \mathbb{L}^{(k-1)}\hat{\mathbb{L}}^{(k)} \iff \hat{\mathbb{L}}^{(k)} = (\mathbb{L}^{(k-1)})^{-1}\mathbb{L}^{(k)} \qquad \wedge \qquad \mathbb{R}^{(k)} = \hat{\mathbb{R}}^{(k)}. \tag{5.66}$$

To dosadíme do definice LR algoritmu:

$$\mathbb{A}^{(k)} = \hat{\mathbb{L}}^{(k)}\hat{\mathbb{R}}^{(k)} = \left((\mathbb{L}^{(k-1)})^{-1}\mathbb{L}^{(k)}\right)\mathbb{R}^{(k)}. \tag{5.67}$$

Pokud trojúhelníková metoda konverguje (tj. $\mathbb{L}^{(k)} \to \mathbb{L}$ a $\mathbb{R}^{(k)} \to \mathbb{R}$), pak limita tohoto výrazu je:

$$\lim_{k\to\infty}\mathbb{A}^{(k)} = (\mathbb{L})^{-1}\mathbb{L}\mathbb{R} = \mathbb{R}. \tag{5.68}$$

Posloupnost matic $\{\mathbb{A}^{(k)}\}$ z LR algoritmu tedy skutečně konverguje k horní trojúhelníkové matici $\mathbb{R}$.

Věta 5.39 (Konvergence LR algoritmu)**.** Nechť je matice $\mathbb{A} \in \mathbb{C}^{n,n}$ regulární a diagonalizovatelná. Předpokládejme, že trojúhelníková metoda s volbou $\mathbb{L}^{(0)} = \mathbb{I}$ pro tuto matici konverguje. Pak konverguje i LR algoritmus a platí, že posloupnost $\{\mathbb{A}^{(k)}\}$ konverguje k horní trojúhelníkové matici, která má na diagonále vlastní čísla matice $\mathbb{A}$ seřazená sestupně podle velikosti v absolutní hodnotě.

5.5 QR algoritmus

Velkou nevýhodou LR algoritmu je jeho numerická nestabilita, která se projevuje zejména při aplikaci na matice větších rozměrů. Z tohoto důvodu navrhl v roce 1961 J. G. F. Francis takzvaný QR algoritmus, který je dnes jednou z nejpoužívanějších metod pro výpočet kompletního spektra. Princip QR algoritmu je shodný s LR algoritmem, ale místo numericky méně stabilního LU rozkladu využívá QR rozklad, tedy rozklad na ortogonální (v reálném případě) nebo unitární (v komplexním případě) a horní trojúhelníkovou matici. V dalším textu se pro jednoduchost omezíme pouze na reálné matice.

5.5.1 QR rozklad

Než představíme samotný algoritmus, zadefinujme si QR rozklad.

Věta 5.40. Nechť $\mathbb{A} \in \mathbb{R}^{n,n}$ je regulární matice. Pak existuje její rozklad do tvaru

$$\mathbb{A} = \mathbb{Q}\mathbb{R}, \tag{5.69}$$

kde $\mathbb{Q}$ je ortogonální matice (tj. $\mathbb{Q}^T\mathbb{Q} = \mathbb{I}$) a $\mathbb{R}$ je horní trojúhelníková matice. Pokud navíc požadujeme, aby diagonální prvky matice $\mathbb{R}$ byly kladné, je tento rozklad jednoznačný.

Důkaz. Prozatím dokážeme jen jednoznačnost QR rozkladu. Jeho existence vyplyne z jednotlivých metod výpočtu. Předpokládejme, že existují dva různé QR rozklady $\mathbb{A} = \mathbb{Q}_1\mathbb{R}_1 = \mathbb{Q}_2\mathbb{R}_2$. Potom

$$\mathbb{Q}_2^T\mathbb{Q}_1 = \mathbb{R}_2\mathbb{R}_1^{-1} \qquad \wedge \qquad \mathbb{Q}_1^T\mathbb{Q}_2 = \mathbb{R}_1\mathbb{R}_2^{-1}. \tag{5.70}$$

Na pravých stranách máme matice horní trojúhelníkové, proto i např. $\mathbb{Q}_2^T\mathbb{Q}_1$ je horní trojúhelníková matice. Její transpozice $(\mathbb{Q}_2^T\mathbb{Q}_1)^T = \mathbb{Q}_1^T\mathbb{Q}_2$ je tedy dolní trojúhelníková, zároveň je ale podle druhé rovnosti výše horní trojúhelníková. Proto $\mathbb{Q}_2^T\mathbb{Q}_1 = \mathbb{D}$ musí být diagonální. Analogicky také $\mathbb{Q}_1^T\mathbb{Q}_2 = \mathbb{D}$ musí být diagonální. Nyní se podívejme na následující rozklad:

$$\mathbb{Q}_2 = \mathbb{Q}_2\mathbb{R}_2\mathbb{R}_2^{-1} = \mathbb{Q}_1\mathbb{R}_1\mathbb{R}_2^{-1} \overset{*}{=} \mathbb{Q}_1\mathbb{Q}_1^T\mathbb{Q}_2 = \mathbb{Q}_1\mathbb{D}, \tag{5.71}$$

kde v rovnosti označené $*$ jsme využili vztah (5.70). Podobně se podívejme na matici $\mathbb{R}_2$:

$$\mathbb{R}_2 = \mathbb{Q}_2^T\mathbb{Q}_2\mathbb{R}_2 \overset{*}{=} \mathbb{Q}_2^T\mathbb{Q}_1\mathbb{R}_1 = \mathbb{D}\mathbb{R}_1, \tag{5.72}$$

kde jsme tentokrát v rovnosti označené $*$ využili předpoklad rovnost rozkladů. Nyní když dokážeme, že $\mathbb{D} = \mathbb{I}$, dostaneme rovnost mezi $\mathbb{R}_1 = \mathbb{R}_2$ a $\mathbb{Q}_1 = \mathbb{Q}_2$.

$$\mathbb{I} = \mathbb{Q}_2^T\mathbb{Q}_2 = (\mathbb{Q}_1\mathbb{D})^T(\mathbb{Q}_1\mathbb{D}) = \mathbb{D}^T\mathbb{Q}_1^T\mathbb{Q}_1\mathbb{D} = \mathbb{D}^T\mathbb{D} = \mathbb{D}^2 \tag{5.73}$$

Z toho plyne, že $\mathbb{D}$ je diagonální matice s prvky ± 1. Z předpokladu pro jednoznačnost ale víme, že $\mathbb{R}_1, \mathbb{R}_2$ mají kladnou diagonálu, a tedy z $\mathbb{R}_2 = \mathbb{D}\mathbb{R}_1$ plyne, že $\mathbb{D} = \mathbb{I}$. □

Poznámka 5.41. Existují tři základní způsoby, jak QR rozklad matice vypočítat:

- Gramův-Schmidtův ortonormalizační proces
- Householderovy transformace

- Givensovy rotace

5.5.1.1 Gramův-Schmidtův ortonormalizační proces

Gramův-Schmidtův proces je klasický algoritmus lineární algebry, který slouží k převedení sady lineárně nezávislých vektorů na sadu vektorů, které jsou vzájemně ortonormální a generují stejný vektorový podprostor. Mějme na vstupu sadu lineárně nezávislých vektorů $\{\vec{x}_1, \ldots, \vec{x}_n\}$. Cílem je zkonstruovat ortonormální sadu $\{\vec{q}_1, \ldots, \vec{q}_n\}$. Postupujeme iteračně:

1. **Krok 1:** První vektor $\vec{q}_1$ získáme jednoduchou normalizací prvního vstupního vektoru $\vec{x}_1$:

$$\vec{q}_1 = \frac{\vec{x}_1}{||\vec{x}_1||_2}. \tag{5.74}$$

   Definujme koeficient $r_{11} = ||\vec{x}_1||_2$. Potom platí $\vec{x}_1 = r_{11}\vec{q}_1$.

2. **Krok k:** Pro $k = 2, \ldots, n$ získáme k-tý ortonormální vektor $\vec{q}_k$ tak, že od vstupního vektoru $\vec{x}_k$ odečteme jeho průměty do směrů všech již zkonstruovaných ortonormálních vektorů $\vec{q}_1, \ldots, \vec{q}_{k-1}$. Tím získáme pomocný vektor $\tilde{\vec{q}}_k$, který je na všechny předchozí kolmý:

$$\tilde{\vec{q}}_k = \vec{x}_k - \sum_{j=1}^{k-1} (\vec{x}_k, \vec{q}_j)\vec{q}_j. \tag{5.75}$$

   Nový ortonormální vektor $\vec{q}_k$ pak dostaneme normalizací vektoru $\tilde{\vec{q}}_k$:

$$\vec{q}_k = \frac{\tilde{\vec{q}}_k}{||\tilde{\vec{q}}_k||_2}. \tag{5.76}$$

Poznámka 5.42 (Složitost)**.** Výpočetní složitost Gramova-Schmidtova procesu je řádově $O(n^3)$. Výpočet provádíme pro každý vektor, dále pro něj odečítáme průměty do všech předchozích vektorů, což je $O(n^2)$ operací. Skalární součin dvou vektorů uvnitř výpočtu průmětu přidá ještě $O(n)$ operací. Celkem tedy $O(n^3)$.

Poznámka 5.43 (Vztah ke QR rozkladu)**.** Definujme koeficienty r_{jk} jako $r_{jk} = (\vec{x}_k, \vec{q}_j)$ pro $j < k$ a $r_{kk} = ||\tilde{\vec{q}}_k||_2$. Z rovnice pro $\tilde{\vec{q}}_k$ můžeme vyjádřit původní vektor $\vec{x}_k$:

$$\vec{x}_k = \tilde{\vec{q}}_k + \sum_{j=1}^{k-1} r_{jk}\vec{q}_j = r_{kk}\vec{q}_k + \sum_{j=1}^{k-1} r_{jk}\vec{q}_j = \sum_{j=1}^{k} r_{jk}\vec{q}_j. \tag{5.77}$$

Tento vztah ukazuje, že každý původní vektor $\vec{x}_k$ lze vyjádřit jako lineární kombinaci prvních k ortonormálních vektorů. Zapíšeme-li tyto vztahy pro všechny $k = 1, \ldots, n$ do jedné maticové rovnice, dostaneme:

$$(\vec{x}_1, \ldots, \vec{x}_n) = (\vec{q}_1, \ldots, \vec{q}_n) \begin{pmatrix} r_{11} & r_{12} & \ldots & r_{1n} \\ 0 & r_{22} & \ldots & r_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & r_{nn} \end{pmatrix}. \tag{5.78}$$

Následující věta pouze formalizuje výše uvedený vztah s QR rozkladem.

Věta 5.44. Nechť $\mathbb{A} \in \mathbb{R}^{n,n}$ je regulární matice. Aplikujeme-li Gramův-Schmidtův ortonormalizační proces na sloupce matice $\mathbb{A}$, tj. položíme $\vec{x}^{(i)} = \vec{a}_{\cdot i}$ pro $i = 1, \ldots, n$, získáme ortonormální vektory $\vec{q}^{(1)}, \ldots, \vec{q}^{(n)}$ a koeficienty r_{ij}. Tento proces je ekvivalentní maticovému zápisu $\mathbb{A} = \mathbb{Q}\mathbb{R}$, kde $\mathbb{Q}$ je ortogonální matice tvořená vektory $\vec{q}^{(i)}$ ve sloupcích a $\mathbb{R}$ je horní trojúhelníková matice tvořená koeficienty r_{ij}. Diagonální prvky r_{ii} jsou navíc kladné, protože vznikly jako normy nenulových vektorů.

Poznámka 5.45 (Modifikovaný Gram-Schmidtův proces)**.** Klasický Gram-Schmidtův proces je náchylný k numerickým chybám, které mohou vést ke ztrátě ortogonality. Pro lepší numerickou stabilitu se v praxi používá

tzv. modifikovaný Gram-Schmidtův proces. Rozdíl je v tom, že projekce se v každém kroku vnitřního cyklu odečítá od již modifikovaného vektoru, nikoliv stále od původního.

5.5.1.2 Householderovy transformace

Druhým, a v praxi často nejpoužívanějším, způsobem výpočtu QR rozkladu je použití Householderových transformací. Jak jsme si již ukázali (viz definice (1.51) a věta (1.55)), Householderova matice $\mathbb{H}_{\vec{w}} = \mathbb{I} - 2\vec{w}\vec{w}^*$ je unitární a hermitovská transformace, jejíž aplikace na vektor $\vec{x}$ odpovídá zrcadlení (reflexi) tohoto vektoru vůči nadrovině L kolmé na Householderův vektor $\vec{w}$. Klíčovou vlastností pro nás je, že vhodnou volbou vektoru $\vec{w}$ umíme transformovat libovolný vektor $\vec{x}$ na vektor $\vec{y}$, který má stejnou normu. Z vlastností popsaných v (1.55) víme, že je-li $\vec{y} = \mathbb{H}_{\vec{w}}\vec{x}$ (tzn. $\vec{y}$ je obrazem vektoru $\vec{x}$ pod transformací $\mathbb{H}_{\vec{w}}$), pak platí $||\vec{y}||_2 = ||\vec{x}||_2$. Také víme, že rozdíl $\mathbb{H}_{\vec{w}}\vec{x} - \vec{x} = \vec{y} - \vec{x}$ je kolmý na nadrovinu L. Proto pro tuto transformaci zvolíme

$$\vec{w} = \frac{\vec{y} - \vec{x}}{||\vec{y} - \vec{x}||_2}, \tag{5.79}$$

kde jsme ještě přidali normování na 1, což je požadavek definice (1.51). Konkrétně, pokud chceme transformovat vektor $\vec{x}$ na vektor ležící na ose dané bázovým vektorem $\vec{e}_1$, volíme cílový vektor $\vec{y}$ jako $\vec{y} = \alpha\vec{e}_1$, kde z podmínky zachování normy musí platit $|\alpha| = ||\vec{x}||_2$. Tímto způsobem můžeme vynulovat všechny složky vektoru $\vec{x}$ kromě té první. Volba Householderova vektoru $\vec{w}$ by v tomto případě byla:

$$\vec{w} = \frac{\vec{x} - ||\vec{x}||_2\vec{e}_1}{||\vec{x} - ||\vec{x}||_2\vec{e}_1||_2}, \tag{5.80}$$

Poznámka 5.46 (Numerická stabilita volby Householderova vektoru)**.** Kdyby měl vektor $\vec{x}$ převládající první složku, tzn. $|x_i| \approx ||\vec{x}||_2$, potom bychom v rozdílu v $\vec{x} - \vec{y} = \vec{x} - \mathbb{H}_{\vec{w}}\vec{x} = \vec{x} - ||\vec{x}||_2\vec{e}_1$ dostávali hodně malá čísla. V takovém případě by docházelo k velkým numerickým chybám, které by mohly vést k nechtěnému ztrátě ortogonality. Nás ale nutně nezajímá směr výsledného vektoru, jde hlavně o to vynulovat všechny složky kromě té první. Proto můžeme zvolit $\vec{y} = -||\vec{x}||_2\vec{e}_1$, čímž zvětšíme velikost rozdílu a tím i numerickou stabilitu výpočtu. Pro zajištění numerické stability se tedy volba znaménka provádí tak, aby se předešlo odčítání dvou blízkých čísel. Je-li první složka vektoru $\vec{x}$ kladná, pak zobrazujeme do záporného směru, naopak je-li záporná, pak do kladného:

$$\vec{y} = -\mathrm{sgn}\,(x_1)||\vec{x}||_2\vec{e}_1, \tag{5.81}$$

Householderův vektor pro transformaci vektoru $\vec{x}$ na násobek $\vec{e}_1$ se tedy volí podle robustního vzorce (pozor na to, že došlo ke složení dvou -, výsledné znaménko je tedy +):

$$\vec{w} = \frac{\vec{x} + \mathrm{sgn}\,(x_1)||\vec{x}||_2\vec{e}_1}{||\vec{x} + \mathrm{sgn}\,(x_1)||\vec{x}||_2\vec{e}_1||_2}. \tag{5.82}$$

Poznámka 5.47 (Nulování specifických složek)**.** Obecně pokud pro $k \geq 1$ chceme zachovat prvních $k-1$ složek vektoru $\vec{x}$ a vynulovat složky od $k+1$ níže, použijeme speciální Householderovu matici, která působí pouze na posledních $n - k + 1$ složkách vektoru. Tato matice má blokovou strukturu:

$$\mathbb{Q}^{(k)} = \begin{pmatrix} \mathbb{I}^{(k-1)} & \mathbb{O} \\ \mathbb{O} & \tilde{\mathbb{Q}}^{(k)} \end{pmatrix}, \tag{5.83}$$

kde $\mathbb{I}^{(k-1)}$ je jednotková matice o rozměru $(k-1) \times (k-1)$ a $\tilde{\mathbb{Q}}^{(k)}$ je Householderova matice o rozměru $(n-k+1) \times (n-k+1)$. Matice $\tilde{\mathbb{Q}}^{(k)}$ je zkonstruována pro subvektor $\vec{x}^{(k)} \in \mathbb{R}^{n-k+1}$, který je tvořen posledními $n - k + 1$ složkami původního vektoru $\vec{x}$. Proto

$$\vec{w}^{(k)} = \frac{\vec{x}^{(k)} + \mathrm{sgn}\,(x_1^{(k)})||\vec{x}^{(k)}||_2\vec{e}_1}{||\vec{x}^{(k)} + \mathrm{sgn}\,(x_1^{(k)})||\vec{x}^{(k)}||_2\vec{e}_1||_2}, \tag{5.84}$$

a tedy

$$\tilde{\mathbb{Q}}^{(k)} = \mathbb{I} - 2\vec{w}^{(k)}(\vec{w}^{(k)})^T. \tag{5.85}$$

Aplikací takto zkonstruované matice na původní vektor $\vec{x}$ získáme vektor $\vec{y} = \mathbb{Q}^{(k)}\vec{x}$, jehož složky mají tvar:

$$y_j = \begin{cases} x_j & \text{pro } j = 1, \ldots, k-1 \\ \operatorname{sgn}(x_1^{(k)}) \cdot ||\vec{x}^{(k)}||_2 & \text{pro } j = k \\ 0 & \text{pro } j = k+1, \ldots, n \end{cases}$$

Prvních $k-1$ složek tedy zůstane nezměněno, k-tá složka se změní na (plus nebo mínus) normu zbytku vektoru a všechny následující složky jsou vynulovány. Tato transformace je podobná GEMu, nicméně zde se omezujeme pouze na unitární transformace, které zachovávají normu vektoru a ortogonalitu.

Je vidět, že pomocí Householderových transformací lze postupně matici $\mathbb{A}$ převést na matici v horním trojúhelníkovém tvaru. Protože je dále $\mathbb{H}_{\vec{w}}$ unitární, vše se děje pomocí unitárních transformací, a tedy máme dobrý základ pro QR rozklad. Myšlenka je tedy postupně nulovat prvky pod diagonálou matice $\mathbb{A}$ pomocí série Householderových reflexí.

1. **Krok 1:** Vezmeme první sloupec matice $\mathbb{A}$, vektor $\vec{a}_{\cdot 1}$. Najdeme takovou Householderovu matici $\mathbb{Q}^{(1)}$ podle předpisu výše (našli jsme ji pro obecné k), která tento vektor transformuje na násobek prvního bázového vektoru $\vec{e}_1$. Aplikací této transformace na celou matici $\mathbb{A}$ získáme:

$$\mathbb{Q}^{(1)}\mathbb{A} = \begin{pmatrix} r_{11} & r_{12} & \ldots & r_{1n} \\ 0 & & & \\ \vdots & & \mathbb{A}^{(1)} & \\ 0 & & & \end{pmatrix}. \tag{5.86}$$

   První sloupec má nyní požadovaný tvar s nulami pod diagonálou.

2. **Krok 2:** Nyní se zaměříme na submatici $\mathbb{A}^{(1)}$ o rozměru $(n-1) \times (n-1)$. Pro její první sloupec opět najdeme Householderovu transformaci $\tilde{\mathbb{Q}}^{(2)}$ (tentokrát o rozměru $(n-1) \times (n-1)$), která v něm vynuluje prvky pod diagonálou. Tuto transformaci vložíme do blokové matice, abychom neovlivnili již upravený první řádek a sloupec:

$$\mathbb{Q}^{(2)} = \begin{pmatrix} 1 & \vec{0}^T \\ \vec{0} & \tilde{\mathbb{Q}}^{(2)} \end{pmatrix}. \tag{5.87}$$

   Aplikací na výsledek z předchozího kroku dostaneme $\mathbb{Q}^{(2)}\mathbb{Q}^{(1)}\mathbb{A}$, kde již i druhý sloupec má nuly pod diagonálou.

3. **Opakování:** Tento proces opakujeme $n-1$ krát. V každém kroku k konstruujeme transformaci $\tilde{\mathbb{Q}}^{(k)}$ pro submatici $\mathbb{A}^{(k-1)}$ a vkládáme ji do větší matice $\mathbb{Q}^{(k)}$ s identitou v levém horním rohu.

Po $n-1$ krocích dostaneme výslednou horní trojúhelníkovou matici $\mathbb{R}$:

$$\mathbb{Q}^{(n-1)} \ldots \mathbb{Q}^{(2)}\mathbb{Q}^{(1)}\mathbb{A} = \mathbb{R}. \tag{5.88}$$

Jelikož je každá matice $\mathbb{Q}^{(k)}$ ortogonální, je i jejich součin ortogonální matice. Označíme-li $\mathbb{Q}^T = \mathbb{Q}^{(n-1)} \ldots \mathbb{Q}^{(2)}\mathbb{Q}^{(1)}$, pak z vlastnosti $\mathbb{Q}^T\mathbb{Q} = \mathbb{I}$ dostáváme $\mathbb{A} = \mathbb{Q}\mathbb{R}$.

Poznámka 5.48 (Složitost)**.** Aplikace jedné Householderovy transformace $\mathbb{H}_{\vec{w}}$ na matici $\mathbb{A}$ je reprezentováno násobením matic, tzn. má složitost $O(n^3)$. Protože musíme aplikovat $n-1$ Householderových transformací, celková složitost by se vyšplhala na $O(n^4)$, což by bylo neúnosné. Aplikace se pro efektivitu neprovádí výpočtem matice $\mathbb{H}_{\vec{w}}$ a následným maticovým násobením. Místo toho se využije vztahu $\mathbb{H}_{\vec{w}}\mathbb{A} = (\mathbb{I} - 2\vec{w}\vec{w}^T)\mathbb{A} = \mathbb{A} - 2\vec{w}(\vec{w}^T\mathbb{A})$. Výpočet tohoto výrazu vyžaduje pouze maticovo-vektorové a vektorovo-vektorové operace se složitostí $O(n^2)$. Protože tento postup opakujeme $(n-1)$-krát, je celková složitost QR rozkladu pomocí Householderových transformací $O(n^3)$.

5.5.1.3 Givensovy rotace

Třetím způsobem výpočtu QR rozkladu jsou Givensovy rotace. Motivace za nimi je podobná, jako u Householderových transformací. Tam, kde ty používaly k transformaci vektoru $\vec{x}$ na vektor $\vec{y}$ ortonormální transformace, tedy takové, které zachovávají úhly, dalším způsobem, jak transformaci provést je právě změnou úhlu. Odtud Givensova rotace. Jde o jemnější nástroj, než Householderovy transformace. Jsou to ortogonální transformace, které jsou, na rozdíl od Householderových reflexí, navrženy tak, aby eliminovaly pouze jeden specifický prvek vektoru.

Definice 5.49 (Givensova rotace)**.** Pro danou dvojici indexů i, j a úhel θ je Givensova matice rotace $\mathbb{G}(i, j, \theta)$ definována jako jednotková matice, u které je podmatice na průsečíku řádků a sloupců i a j nahrazena rotační maticí:

$$\mathbb{G}(i,j,\theta) = \begin{pmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ & & \cos\theta & & \sin\theta & & \\ & & & \ddots & & & \\ & & -\sin\theta & & \cos\theta & & \\ & & & & & \ddots & \\ & & & & & & 1 \end{pmatrix}. \tag{5.89}$$

Aplikace této matice na vektor $\vec{x}$ způsobí rotaci v rovině (i, j) a ovlivní pouze i-tou a j-tou složku výsledného vektoru $\vec{y}$. Platí tedy

$$y_k = \begin{cases} x_i \cos\theta + x_j \sin\theta & \text{pro } k = i \\ -x_i \sin\theta + x_j \cos\theta & \text{pro } k = j \\ x_k & \text{pro } k \neq i, j \end{cases} \tag{5.90}$$

Poznámka 5.50 (Aplikace při nulování složek)**.** Cílem je zvolit úhel θ tak, abychom vynulovali jednu ze dvou ovlivněných složek. Pokud chceme, aby se po transformaci $\vec{y} = \mathbb{G}(i, j, \theta)\vec{x}$ vynulovala j-tá složka (tj. $y_j = 0$), volíme kosinus a sinus úhlu θ následovně:

$$c^{ij} = \cos\theta = \frac{x_i}{\sqrt{x_i^2 + x_j^2}}, \quad s^{ij} = \sin\theta = \frac{x_j}{\sqrt{x_i^2 + x_j^2}}. \tag{5.91}$$

Aplikací této transformace se j-tá složka vynuluje a i-tá složka se změní na hodnotu $\sqrt{x_i^2 + x_j^2}$, zatímco všechny ostatní složky vektoru zůstanou nezměněny. To odpovídá rotaci o úhel $\theta = \arctan\left(-\frac{x_j}{x_i}\right)$. Označení konstant c^{ij} a s^{ij} se nám bude hodit později.

Poznámka 5.51 (Korektnost)**.** Givensovy rotace má smysl zkusit použít jako základ pro QR rozklad, jelikož jde o unitární transformace. To lze ověřit výpočtem $\mathbb{G}(i, j, \theta)\mathbb{G}(i, j, \theta)^T = \mathbb{I}$.

Matici $\mathbb{A}$ převedeme na horní trojúhelníkový tvar postupnou eliminací všech prvků pod hlavní diagonálou. Postupujeme systematicky, například po sloupcích:

1. **Eliminace v 1. sloupci:** Cílem je vynulovat všechny prvky v prvním sloupci pod diagonálou, tedy $a_{21}, a_{31}, \ldots, a_{n1}$.

   - Začneme s prvkem a_{21}. K jeho vynulování použijeme prvek a_{11} a rotaci v rovině $(1, 2)$. Sestrojíme matici $\mathbb{G}^{(21)}$, kde koeficienty $c^{(21)}$ a $s^{(21)}$ vypočteme jako:

   $$c^{(21)} = \frac{a_{11}}{\sqrt{a_{11}^2 + a_{21}^2}}, \quad s^{(21)} = \frac{a_{21}}{\sqrt{a_{11}^2 + a_{21}^2}}.$$

   Aplikací této transformace zleva, tj. výpočtem $\mathbb{G}^{(21)}\mathbb{A}$, se změní pouze první a druhý řádek, přičemž

prvek na pozici $(2,1)$ bude nulový a prvek na pozici $(1,1)$ bude mít novou hodnotu $a_{11}^{(21)} = \sqrt{a_{11}^2 + a_{21}^2}$.

$$\mathbb{G}^{(21)}\mathbb{A} = \begin{pmatrix} c^{(21)} & s^{(21)} & & & \\ -s^{(21)} & c^{(21)} & & & \\ & & 1 & & \\ & & & \ddots & \\ & & & & 1 \end{pmatrix} \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ a_{31} & a_{32} & \dots & a_{3n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix} = \begin{pmatrix} a_{11}^{(21)} & a_{12}^{(21)} & \dots & a_{1n}^{(21)} \\ 0 & a_{22}^{(21)} & \dots & a_{2n}^{(21)} \\ a_{31} & a_{32} & \dots & a_{3n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix}$$

- Dále nulujeme prvek a_{31} v původní matici. K tomu použijeme aktuální hodnotu prvku na pozici $(1,1)$ (tedy $a_{11}^{(21)}$) a sestrojíme novou rotační matici $\mathbb{G}^{(31)}$ v rovině $(1,3)$. Násobením $\mathbb{G}^{(31)}(\mathbb{G}^{(21)}\mathbb{A})$ se změní pouze první a třetí řádek. Prvek na pozici $(3,1)$ se vynuluje a prvek $(2,1)$ zůstane nulový, protože rotace neovlivní druhý řádek.
- Tímto způsobem postupně eliminujeme všechny prvky a_{i1} pro $i = 2, \dots, n$ pomocí rotací $\mathbb{G}^{(i1)}$. Po dokončení kroku pro první sloupec bude mít matice tvar:

$$\mathbb{G}_1\mathbb{A} = (\mathbb{G}^{(n1)} \dots \mathbb{G}^{(21)})\mathbb{A} = \begin{pmatrix} a_{11}^{(n1)} & a_{12}^{(n1)} & a_{13}^{(n1)} & \dots & a_{1n}^{(n1)} \\ 0 & a_{22}^{(21)} & a_{23}^{(21)} & \dots & a_{2n}^{(21)} \\ 0 & a_{32}^{(31)} & a_{33}^{(31)} & \dots & a_{3n}^{(31)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & a_{n-1,2}^{(n-1,1)} & a_{n-1,3}^{(n-1,1)} & \dots & a_{n-1,n}^{(n-1,1)} \\ 0 & a_{n2}^{(n1)} & a_{n3}^{(n1)} & \dots & a_{nn}^{(n1)} \end{pmatrix}.$$

  První řádek nyní odpovídá prvnímu řádku horní trojúhelníkové matice $\mathbb{R}$, tedy $r_{1j} = a_{1j}^{n1}$.

2. **Eliminace v 2. sloupci:** Nyní se zaměříme na druhý sloupec a nulujeme prvky pod diagonálou, tj. $a_{32}^{(31)}, a_{42}^{(41)}, \dots, a_{n2}^{(n1)}$. Použijeme k tomu rotace $\mathbb{G}^{(32)}, \mathbb{G}^{(42)}, \dots, \mathbb{G}^{(n2)}$ v rovinách $(2,i)$, které budou působit na druhý a i-tý řádek. Tyto rotace díky své struktuře neovlivní již vynulovaný první sloupec.
3. **Opakování:** Proces opakujeme pro všechny sloupce až do $n-1$, dokud nezískáme horní trojúhelníkovou matici $\mathbb{R}$.

Výsledkem je série maticových násobení $\mathbb{G}_N \dots \mathbb{G}_2\mathbb{G}_1\mathbb{A} = \mathbb{R}$, kde každá matice $\mathbb{G}_i$ je jedna z použitých Givensových rotací. Označíme-li součin všech rotačních matic jako $\mathbb{Q}^T = \mathbb{G}_N \dots \mathbb{G}_1$, dostáváme hledaný rozklad $\mathbb{A} = \mathbb{Q}\mathbb{R}$.

Poznámka 5.52 (Složitost)**.** Pro plnou matici řádu n je potřeba provést řádově $n^2/2$ Givensových rotací. Aplikace jedné Givensovy rotace v maticové podobě by mělo složitost $O(n^3)$. Protože ale víme, že rotace ovlivní pouze dva řádky, má tedy složitost $O(n)$. Celková složitost QR rozkladu pomocí Givensových rotací je proto $O(n^3)$. Pro řídké matice, kde je potřeba nulovat jen několik málo prvků, mohou být Givensovy rotace výhodnější než Householderovy transformace.

5.5.1.4 Shrnutí a srovnání metod QR rozkladu

Ukázali jsme si tři základní metody výpočtu QR rozkladu, z nichž každá má své výhody a nevýhody.

- **Gramův-Schmidtův proces:** Je koncepčně jednoduchý, ale v klasické podobě je numericky nestabilní. Jeho modifikovaná verze má lepší stabilitu a používá se v některých jiných numerických algoritmech.
- **Householderovy transformace:** Jsou numericky velmi stabilní a obecně efektivnější pro plné matice, protože každá transformace nuluje celý zbytek sloupce najednou.
- **Givensovy rotace:** Jsou rovněž numericky velmi stabilní. Jejich výhoda spočívá v možnosti cíleného nulování jednotlivých prvků, což je činí vhodnými pro práci s řídkými maticemi nebo pro paralelní implementace.

Z hlediska výpočetní složitosti jsou všechny tři metody pro plné matice řádu $O(n^3)$.

5.5.2 QR algoritmus

Definice 5.53 (QR algoritmus)**.** Metoda konstruuje posloupnost matic $\{\mathbb{T}^{(k)}\}_{k=1}^{\infty}$. Pro zadanou matici $\mathbb{A} \in \mathbb{R}^{n,n}$ a počáteční volbu $\mathbb{T}^{(1)} = \mathbb{A}$ má iterace pro $k = 1, 2, \ldots$ následující tvar:

1. Provedeme QR rozklad matice $\mathbb{T}^{(k)}$:
$$\mathbb{T}^{(k)} = \mathbb{Q}^{(k)}\mathbb{R}^{(k)}, \tag{5.92}$$
kde $\mathbb{Q}^{(k)}$ je ortogonální a $\mathbb{R}^{(k)}$ je horní trojúhelníková matice.
2. Následující člen posloupnosti $\mathbb{T}^{(k+1)}$ získáme vynásobením faktorů v opačném pořadí:
$$\mathbb{T}^{(k+1)} = \mathbb{R}^{(k)}\mathbb{Q}^{(k)}. \tag{5.93}$$

Poznámka 5.54 (Podobnostní transformace)**.** Každý krok QR algoritmu představuje ortogonální podobnostní transformaci, jelikož z definičních vztahů plyne $\mathbb{R}^{(k)} = (\mathbb{Q}^{(k)})^T\mathbb{T}^{(k)}$, a tedy:

$$\mathbb{T}^{(k+1)} = \mathbb{R}^{(k)}\mathbb{Q}^{(k)} = (\mathbb{Q}^{(k)})^T\mathbb{T}^{(k)}\mathbb{Q}^{(k)}. \tag{5.94}$$

Všechny matice v posloupnosti $\{\mathbb{T}^{(k)}\}$ jsou si tedy ortogonálně podobné a mají shodná vlastní čísla jako původní matice $\mathbb{A}$. V případě konvergence, tzn. $\prod_{k=0}^{k}\mathbb{Q}^{(k)} \to \mathbb{U}$ a $\mathbb{T}^{(k)} \to \mathbb{R}$ pro $k \to +\infty$, pak $\mathbb{R} = \mathbb{U}^*\mathbb{A}\mathbb{U}$, resp. $\mathbb{A} = \mathbb{U}^*\mathbb{R}\mathbb{U}$, což je rozklad ze Schurovy věty. Potom na diagonále matice $\mathbb{R}$ jsou vlastní čísla matice $\mathbb{A}$ (protože jde o podobnostní transformaci).

5.5.2.1 Konvergence QR algoritmu

Poznámka 5.55. QR rozklad je spojitou funkcí prvků matice $\mathbb{A}$, jak je vidět např. z definice QR rozkladu v Grammově-Schmidtově algoritmu.

Lemma 5.56. Existuje-li QR rozklad matice $\mathbb{A}^k = \mathcal{Q}^{(k)}\mathcal{R}^{(k)}$, pak platí

$$\mathcal{Q}^{(k)} = \mathbb{Q}^{(1)}\mathbb{Q}^{(2)}\ldots\mathbb{Q}^{(k)}$$
$$\mathcal{R}^{(k)} = \mathbb{R}^{(k)}\mathbb{R}^{(k-1)}\ldots\mathbb{R}^{(1)}$$

Důkaz. Analogický jako u LR alogritmu. □

Věta 5.57 (Konvergence QR algoritmu)**.** Nechť matice $\mathbb{A} \in \mathbb{R}^{n,n}$ má vlastní čísla λ_j, která splňují podmínku

$$|\lambda_1| > |\lambda_2| > \cdots > |\lambda_n| > 0.$$

Pak posloupnost matic $\{\mathbb{T}^{(k)}\}$ generovaná QR algoritmem konverguje k horní trojúhelníkové matici, která má na diagonále vlastní čísla matice $\mathbb{A}$ seřazená podle jejich absolutní hodnoty. Pokud je navíc matice $\mathbb{A}$ symetrická, pak posloupnost $\{\mathbb{T}^{(k)}\}$ konverguje k diagonální matici.

Důkaz. (TODO: ověřit správnost značení v tomto důkazu) Nechť $\mathbb{A} = \mathbb{X}^{-1}\mathbb{D}\mathbb{X}$, kde $\mathbb{D}$ je diagonální matice s vlastními čísly λ_j na diagonále a $\mathbb{X}$ je regulární. Potom $\mathbb{A}^k = \mathbb{X}\mathbb{D}^k\mathbb{X}^{-1}$. Z regularity matice $\mathbb{X}$ (TODO to nestačí) plyne, že existují rozklady $\mathbb{X} = \mathbb{Q}_X\mathbb{R}_X$ a $\mathbb{X}^{-1} = \mathbb{L}^Y\mathbb{R}^Y$. Potom

$$\mathbb{A}^k = \mathbb{Q}_X\mathbb{R}_X\mathbb{D}^k\mathbb{L}_Y\mathbb{R}_Y = \mathbb{Q}_X\mathbb{R}_X\mathbb{D}^k\mathbb{L}_Y\mathbb{D}^{-k}\mathbb{D}^k\mathbb{R}_Y \tag{5.95}$$

Analogicky jako v důkazu věty o konvergenci LR algoritmu dojdeme k tomu, že

$$[\mathbb{D}^k\mathbb{L}_Y\mathbb{D}^{-k}]_{ij} = \begin{cases} 0 & \text{pro } j > i \\ \lambda_i 1 \frac{1}{\lambda_i} = 1 & \text{pro } j = i \\ l_{ij}\frac{\lambda_i}{\lambda_j} & \text{pro } j < i \end{cases} \tag{5.96}$$

Dále víme, že pro $j < i$ je $|\lambda_j| > |\lambda_i|$ a proto $\frac{\lambda_i}{\lambda_j} < 0$ resp. $\frac{\lambda_i}{\lambda_j} \to 0$ pro $k \to \infty$. Proto $\mathbb{D}^k\mathbb{L}_Y\mathbb{D}^{-k} \to \mathbb{I}$ pro $k \to \infty$ a tedy lze napsat $\mathbb{D}^k\mathbb{L}_Y\mathbb{D}^{-k} = \mathbb{I} + \mathbb{F}^{(k)}$ kde $\mathbb{F}^{(k)} \to 0$ pro $k \to \infty$. Nyní

$$\begin{aligned} \mathbb{A}^k &= \mathbb{Q}_X\mathbb{R}_X(\mathbb{I} + \mathbb{F}^{(k)})\mathbb{D}^k\mathbb{R}_Y = \mathbb{Q}_X\mathbb{R}_X(\mathbb{I} + \mathbb{F}^{(k)})\mathbb{R}_X^{-1}\mathbb{R}_X\mathbb{D}^k\mathbb{R}_Y \\ &= \mathbb{Q}_X(\mathbb{I} + \mathbb{R}_X\mathbb{F}^{(k)}\mathbb{R}_X^{-1})\mathbb{R}_X\mathbb{D}^k\mathbb{R}_Y = \mathbb{Q}_X(\mathbb{I} + \mathbb{G}^{(k)})\mathbb{R}_X\mathbb{D}^k\mathbb{R}_Y, \end{aligned} \tag{5.97}$$

kde $\mathbb{G}^{(k)} = \mathbb{R}_X\mathbb{F}^{(k)}\mathbb{R}_X^{-1} \to \mathbb{O}$ pro $k \to +\infty$. Opět analogicky víme, že pro $\mathbb{I} + \mathbb{G}^{(k)} \to \mathbb{I}$ pro $k \to +\infty$ existuje QR rozklad, a tedy $\mathbb{I} + \mathbb{G}^{(k)} = \mathbb{Q}_G^{(k)}\mathbb{R}_G^{(k)}$, kde $\mathbb{Q}_G^{(k)} \to \mathbb{I}$ a $\mathbb{R}_G^{(k)} \to \mathbb{I}$ pro $k \to +\infty$. Celkově nyní dostáváme, že

$$\mathbb{A}^k = \mathbb{Q}_X\mathbb{Q}_G^{(k)}\mathbb{R}_G^{(k)}\mathbb{R}_X\mathbb{D}^k\mathbb{R}_Y. \tag{5.98}$$

Protože $\mathbb{Q}_G^{(k)} \to \mathbb{I}$, nutně $\mathbb{Q}_X\mathbb{Q}_G^{(k)} \to \mathbb{Q}_X$ pro $k \to +\infty$. Dále víme, že

$$\mathbb{T}^{(k)} = \left(\mathbb{Q}^{(1)}...\mathbb{Q}^{(k)}\right)^* \mathbb{A}\left(\mathbb{Q}^{(1)}...\mathbb{Q}^{(k)}\right) = \left(\mathcal{Q}^{(k)}\right)^* \mathbb{A}\mathcal{Q}^{(k)} \to \mathbb{Q}_X^*\mathbb{A}\mathbb{Q}_X, \tag{5.99}$$

čímž jsme dokázali konvergenci posloupnosti $\{\mathbb{T}^{(k)}\}$ k matici $\mathbb{Q}_X^*\mathbb{A}\mathbb{Q}_X = \mathbb{T}$. Dále bychom chtěli říct, že $\mathbb{T}^{(k)}$ je trojúhelníková. Vyjdeme opět z toho, že

$$\begin{aligned} \mathbb{T}^{(k)} &= \left(\mathbb{Q}^{(1)}...\mathbb{Q}^{(k)}\right)^* \mathbb{A}\left(\mathbb{Q}^{(1)}...\mathbb{Q}^{(k)}\right) = \left(\mathcal{Q}^{(k)}\right)^* \mathbb{A}\mathcal{Q}^{(k)} \\ = (\mathbb{Q}_X\mathbb{Q}_G^{(k)})^*\mathbb{A}(\mathbb{Q}_X\mathbb{Q}_G^{(k)}) &= \left(\mathbb{Q}_G^{(k)}\right)^* \mathbb{Q}_X^*\mathbb{A}\mathbb{Q}_X\mathbb{Q}_G^{(k)} = \left(\mathbb{Q}_G^{(k)}\right)^* \mathbb{Q}_X^*\mathbb{X}\mathbb{D}\mathbb{X}^{-1}\mathbb{Q}_X\mathbb{Q}_G^{(k)} \\ = \left(\mathbb{Q}_G^{(k)}\right)^* \mathbb{Q}_X^*\mathbb{Q}_X^*\mathbb{Q}_X\mathbb{R}_X\mathbb{D}\mathbb{R}_X^{-1}\mathbb{Q}_X^*\mathbb{Q}_X\mathbb{Q}_G^{(k)} &= \left(\mathbb{Q}_G^{(k)}\right)^* \mathbb{R}_X\mathbb{D}\mathbb{R}_X^{-1}\mathbb{Q}_G^{(k)} \to \mathbb{R}_X\mathbb{D}\mathbb{R}_X^{-1} \end{aligned} \tag{5.100}$$

a tedy dostáváme matici horní trojúhelníkovou, jejíž diagonála je shodná s $\mathbb{D}$, analogicky jako v důkazu konvergence LR algoritmu. Nechť nyní $\mathbb{A}$ je symetrická[7], tedy

$$\mathbb{Q}_X^*\mathbb{T}\mathbb{Q}_X = \mathbb{A} = \mathbb{A}^* = \mathbb{Q}_X^*\mathbb{T}^*\mathbb{Q}_X. \tag{5.101}$$

Odtud $\mathbb{T} = \mathbb{T}^*$, a tedy $\mathbb{T}$ je diagonální. □

5.5.3 QR algoritmus s Hessenbergovými maticemi

Základní QR algoritmus je sice numericky stabilní, ale výpočetně náročný. Každá iterace vyžaduje QR rozklad plné matice, což je operace se složitostí $O(n^3)$. Pro praktické použití je tedy nutné algoritmus zefektivnit. Toho se dosahuje převedením matice na speciální tvar, který je pro QR iterace výhodnější.

Definice 5.58 (Hessenbergův tvar)**.** Matice je v (horním) Hessenbergově tvaru, pokud jsou všechny její prvky pod první subdiagonálou nulové (tj. $a_{ij} = 0$ pro $i > j + 1$). Jedná se tedy o horní trojúhelníkovou matici, která může mít navíc nenulové prvky bezprostředně pod hlavní diagonálou.

Poznámka 5.59. Jde v jistém smyslu o nejjednodušší tvar, na který lze libovolnou regulární matici převést podobnostními transformacemi, které lze získat přímým algoritmem. Tím jsou zachovány spektrální vlastnosti, tedy i vlastní čísla a vlastní vektory. Samotný převod bude mít složitost $O(n^3)$, ale následné QR iterace na Hessenbergově matici mají složitost pouze $O(n^2)$, což je dramatické zrychlení. Převod navíc stačí provést jednou na začátku.

[7] Připomeňme, že jsme v realných číslech a tedy hvězdička a T mají stejný význam.

5.5.3.1 Převod na Hessenbergův tvar

K samotnému převodu se opět využijí Householderovy reflexe, avšak aplikované tak, aby byla zachována podobnost a zároveň se nenulovala první subdiagonála. Proces je přímý (neiterativní) a skládá se z $n-2$ kroků. Cílem prvního kroku je vynulovat prvky v prvním sloupci na pozicích $3, \ldots, n$. Toho dosáhneme Householderovou transformací, která působí na subvektor začínající až druhým prvkem sloupce.

1. Definujeme subvektor $\vec{x}^{(1)} = (a_{21}, a_{31}, \ldots, a_{n1})^T \in \mathbb{R}^{n-1}$.
2. Pro tento vektor zkonstruujeme $(n-1)$-rozměrnou Householderovu matici $\tilde{\mathbb{Q}}^{(1)}$.
3. Tuto menší transformaci vložíme do matice $\mathbb{Q}^{(1)}$ o plném rozměru $n \times n$:

$$\mathbb{Q}^{(1)} = \begin{pmatrix} 1 & \vec{0}^T \\ \vec{0} & \tilde{\mathbb{Q}}^{(1)} \end{pmatrix}. \tag{5.102}$$

4. Aplikujeme transformaci na matici $\mathbb{A}$ zleva. Výsledná matice $\mathbb{Q}^{(1)}\mathbb{A}$ bude mít v prvním sloupci nuly na pozicích $3, \ldots, n$. Tato matice však ještě není podobná matici $\mathbb{A}$.

$$\mathbb{Q}^{(1)}\mathbb{A} = \begin{pmatrix} a_{11} & \bar{a}_{12}^{(1)} & \bar{a}_{13}^{(1)} & \ldots & \bar{a}_{1n}^{(1)} \\ \bar{a}_{21}^{(1)} & \bar{a}_{22}^{(1)} & \bar{a}_{23}^{(1)} & \ldots & \bar{a}_{2n}^{(1)} \\ 0 & \bar{a}_{32}^{(1)} & \bar{a}_{33}^{(1)} & \ldots & \bar{a}_{3n}^{(1)} \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & \bar{a}_{n,2}^{(1)} & \bar{a}_{n,3}^{(1)} & \ldots & \bar{a}_{n,n}^{(1)} \end{pmatrix} \tag{5.103}$$

   TODO: myslím, že tady je chyba, že $\bar{a}_{1j}^{(1)}$ jsou stále původní hodnoty, ne ty transformované. Opravit. Transformace by neměla působit na první řádek. Proužky jsou zde matoucí. Znamenají, že se hodnoty ještě změní pro zachování podobnosti. Nahradit čárkami nebo aspoň popsat.

5. Abychom zachovali podobnost, musíme transformaci aplikovat i zprava pomocí $(\mathbb{Q}^{(1)})^{-1} = (\mathbb{Q}^{(1)})^T = \mathbb{Q}^{(1)}$. Vypočteme tedy:

$$\mathbb{H}^{(1)} = (\mathbb{Q}^{(1)}\mathbb{A})\mathbb{Q}^{(1)}. \tag{5.104}$$

   Díky struktuře matice $\mathbb{Q}^{(1)}$ (jednička a nuly v prvním řádku a sloupci) toto násobení zprava neovlivní již vynulované prvky v prvním sloupci. Výsledná matice $\mathbb{H}^{(1)}$ je podobná matici $\mathbb{A}$ a má tvar:

$$\mathbb{Q}^{(1)}\mathbb{A}\mathbb{Q}^{(1)} = \begin{pmatrix} a_{11} & a_{12}^{(1)} & a_{13}^{(1)} & \ldots & a_{1n}^{(1)} \\ \bar{a}_{21}^{(1)} & a_{22}^{(1)} & a_{23}^{(1)} & \ldots & a_{2n}^{(1)} \\ 0 & a_{32}^{(1)} & a_{33}^{(1)} & \ldots & a_{3n}^{(1)} \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & a_{n,2}^{(1)} & a_{n,3}^{(1)} & \ldots & a_{n,n}^{(1)} \end{pmatrix} = \begin{pmatrix} h_{11} & h_{12} & a_{13}^{(1)} & \ldots & a_{1n}^{(1)} \\ h_{21} & h_{22} & a_{23}^{(1)} & \ldots & a_{2n}^{(1)} \\ 0 & a_{32}^{(1)} & a_{33}^{(1)} & \ldots & a_{3n}^{(1)} \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & a_{n,2}^{(1)} & a_{n,3}^{(1)} & \ldots & a_{n,n}^{(1)} \end{pmatrix} = \mathbb{H}^{(1)} \tag{5.105}$$

V druhém kroku se proces opakuje pro druhý sloupec matice $\mathbb{H}^{(1)}$. Cílem je vynulovat prvky na pozicích $4, \ldots, n$. Householderova transformace $\tilde{\mathbb{Q}}^{(2)}$ se tentokrát konstruuje pro subvektor $(a_{32}^{(1)}, a_{42}^{(1)}, \ldots, a_{n2}^{(1)})^T$ a je o rozměru $(n-2)\times(n-2)$. Celková transformační matice $\mathbb{Q}^{(2)}$ má pak na diagonále dva jednotkové prvky. Opět provedeme oboustrannou aplikaci $\mathbb{H}^{(2)} = \mathbb{Q}^{(2)}\mathbb{H}^{(1)}\mathbb{Q}^{(2)}$ a získáme matici s vynulovanými prvky i ve druhém sloupci (přičemž první sloupec zůstane nezměněn). Tento postup opakujeme celkem $n-2$ krát. Výsledná matice $\mathbb{H} = \mathbb{H}^{(n-2)}$ je v Hessenbergově tvaru a je ortogonálně podobná původní matici $\mathbb{A}$. Celý tento převod má výpočetní složitost $O(n^3)$.

5.5.3.2 Vlastní QR iterace s Hessenbergovou maticí

Samotný zefektivněný algoritmus probíhá ve dvou fázích:

1. Matice $\mathbb{A}$ se jednorázově převede na podobnou matici $\mathbb{H}^{(1)}$ v Hessenbergově tvaru.

2. Na matici $\mathbb{H}^{(1)}$ se aplikuje QR algoritmus, tedy posloupnost iterací pro $k = 1, 2, \ldots$:

$$\begin{aligned} \mathbb{H}^{(k)} &= \mathbb{Q}^{(k)}\mathbb{R}^{(k)} \\ \mathbb{H}^{(k+1)} &= \mathbb{R}^{(k)}\mathbb{Q}^{(k)} \end{aligned} \tag{5.106}$$

Klíčové je, že QR rozklad Hessenbergovy matice $\mathbb{H}^{(k)}$ lze provést velmi efektivně. K vynulování $n-1$ prvků na subdiagonále stačí použít $n-1$ vhodně zvolených Givensových rotací $\mathbb{G}_1^{(k)}, \ldots, \mathbb{G}_{n-1}^{(k)}$. Matice $\mathbb{Q}^{(k)}$ je pak dána jejich součinem. Celá jedna iterace se tak skládá z:

- **Rozkladu:** Vypočteme $\mathbb{R}^{(k)} = (\mathbb{Q}^{(k)})^T\mathbb{H}^{(k)}$, kde $(\mathbb{Q}^{(k)})^T = \mathbb{G}_{n-1}^{(k)} \ldots \mathbb{G}_1^{(k)}$.
- **Složení:** Vypočteme nový iterant $\mathbb{H}^{(k+1)} = \mathbb{R}^{(k)}\mathbb{Q}^{(k)} = \mathbb{R}^{(k)}(\mathbb{G}_1^{(k)})^T \ldots (\mathbb{G}_{n-1}^{(k)})^T$.

Jedna Givensova rotace má složitost $O(n)$, protože ovlivní pouze dva řádky. Celková složitost jedné iterace QR algoritmu s Hessenbergovou maticí je tedy $O(n^2)$, což je výrazné zrychlení oproti původnímu QR algoritmu.

Lemma 5.60. Je-li matice $\mathbb{H}^{(k)}$ v Hessenbergově tvaru, pak je matice $\mathbb{H}^{(k+1)} = \mathbb{R}^{(k)}\mathbb{Q}^{(k)}$ rovněž v Hessenbergově tvaru.

Příklad 5.61. Platnost lemmatu můžeme nastínit na příkladu jedné iterace. Mějme matici $\mathbb{H}^{(k)}$ v Hessenbergově tvaru. První cílem je eliminovat nenulové prvky na subdiagonále, tedy $h_{21}, h_{32}, \ldots, h_{n,n-1}$. Použijeme k tomu sekvenci $n-1$ Givensových rotací.

- Nejprve eliminujeme prvek h_{21} pomocí rotace $\mathbb{G}^{(21)}$ v rovině $(1,2)$. Aplikací zleva na $\mathbb{H}^{(k)}$ získáme matici $\mathbb{G}^{(21)}\mathbb{H}^{(k)}$, která má na pozici $(2,1)$ nulu.
- Dále eliminujeme prvek h_{32} v upravené matici pomocí rotace $\mathbb{G}^{(32)}$ v rovině $(2,3)$. Aplikací $\mathbb{G}^{(32)}(\mathbb{G}^{(21)}\mathbb{H}^{(k)})$ se vynuluje prvek na pozici $(3,2)$, aniž by se obnovil nulový prvek na pozici $(2,1)$.
- Takto postupujeme dále, až poslední rotací $\mathbb{G}^{(n,n-1)}$ eliminujeme prvek $h_{n,n-1}$.

Výsledkem je horní trojúhelníková matice $\mathbb{R}^{(k)} = (\mathbb{G}^{(n,n-1)} \ldots \mathbb{G}^{(21)})\mathbb{H}^{(k)}$. Označme $\left(\mathbb{Q}^{(k)}\right)^T = \mathbb{G}^{(n,n-1)} \ldots \mathbb{G}^{(21)}$.
Nyní můžeme vše složit zpět a získat $\mathbb{H}^{(k+1)} = \mathbb{R}^{(k)}\mathbb{Q}^{(k)}$. Musíme tedy napočítat matici $\mathbb{R}^{(k)}(\mathbb{G}^{(21)})^T(\mathbb{G}^{(32)})^T \ldots (\mathbb{G}^{(n,n-1)})^T$.
Budeme postupně násobit zprava transponovanými rotačními maticemi.

- Násobení maticí $(\mathbb{G}^{(21)})^T$ ovlivní pouze první a druhý sloupec matice $\mathbb{R}^{(k)}$. Vytvoří nový (obecně nenulový) prvek na pozici $(2,1)$, ale ostatní sloupce nechá beze změny.
- Následné násobení maticí $(\mathbb{G}^{(32)})^T$ ovlivní pouze druhý a třetí sloupec. Vytvoří nový nenulový prvek na pozici $(3,2)$. Důležité je, že toto násobení neovlivní první sloupec, tedy prvek na pozici $(2,1)$ zůstane zachován a nevzniknou žádné další nenulové prvky pod subdiagonálou.
- Tento proces pokračuje. Každá další matice $(\mathbb{G}^{(k,k-1)})^T$ "vygeneruje"pouze jeden nenulový prvek na subdiagonále na pozici $(k, k-1)$.

Výsledná matice $\mathbb{H}^{(k+1)}$ má tedy nenulové prvky pouze na hlavní diagonále, nad ní a na první subdiagonále. Je tedy opět v Hessenbergově tvaru.

Tato klíčová vlastnost zaručuje, že pokud začneme s Hessenbergovou maticí, všechny další iteranty $\mathbb{H}^{(k)}$ si tento tvar udrží.

5.6 Otázky

- Trojíhelníková metoda
- LR algoritmus
- QR rozklad: Gram-Schmidt, Householderovy transformace, Givensovy rotace
- QR algoritmus: Hessenbergovy iterace

- Na čem závisí rychlost konvergence? Jak konvergenci urychlit? (zmenšit poměr lambda i lambda j)

Kapitola 6

Metody řešení nelineárních rovnic

V této kapitole se zaměříme na hledání řešení (kořenů) nelineární rovnice $f(x) = 0$, kde f je reálná funkce jedné reálné proměnné. Kořen rovnice budeme značit jako α. Řešení se typicky skládá ze dvou kroků:

1. **Separace kořenů:** Na rozdíl od metod pro řešení lineárních soustav, metody pro nelineární rovnice obecně nekonvergují globálně (tj. pro libovolnou počáteční aproximaci). Je proto nutné nejprve najít intervaly, z nichž každý obsahuje právě jeden kořen. Proč tomu tak je? U lineární soustavy $\mathbb{A}\vec{x} = \vec{b}$ (s regulární maticí $\mathbb{A}$) existuje vždy právě jedno řešení. Celý "prostor"úlohy tak směřuje k tomuto jedinému bodu. Iterační metody pro lineární rovnice, pokud splňují podmínky konvergence, najdou toto řešení bez ohledu na to, kde začneme. Nelineární rovnice $f(x) = 0$ je ale mnohem komplikovanější. Funkce $f(x)$ může být libovolně složitá – může mít více kořenů (např. polynom), nekonečně mnoho kořenů (např. $f(x) = \sin(x)$), nebo naopak žádný reálný kořen (např. $f(x) = x^2 + 1$). Může mít lokální extrémy, body nespojitosti a další problematické oblasti. Iterační metody pro hledání kořenů fungují zpravidla tak, že z nějakého bodu x_k se na základě lokální informace o funkci (typicky hodnota $f(x_k)$ a její derivace $f'(x_k)$) rozhodnou, kam se posunout pro další, lepší aproximaci x_{k+1}.

   - Pokud existuje více kořenů, záleží na počáteční aproximaci x_0, ke kterému z nich bude metoda konvergovat. Každý kořen má svou "spádovou oblast"(basin of attraction). Pokud začneme mimo spádovou oblast hledaného kořene, najdeme jiný kořen, nebo metoda nebude konvergovat vůbec.
   - Metoda může snadno selhat, pokud se počáteční bod nachází například v blízkosti lokálního minima, kde funkce neprotíná osu x, nebo pokud se iterace „zacyklí“ či začnou „utíkat“ do nekonečna.

   Separace kořenů je tedy klíčový první krok, kterým tuto nejistotu odstraníme. Nalezením malého intervalu $[a, b]$, kde máme zaručeno (např. pomocí věty o separaci kořenů (6.1)), že se v něm nachází právě jeden kořen, si vytvoříme "bezpečnou zónu". Následně můžeme aplikovat numerickou metodu s počáteční aproximací zvolenou uvnitř této zóny s jistotou, že pokud metoda zkonverguje, nalezne právě ten kořen, který hledáme.

2. **Výpočet kořene:** Po nalezení takového intervalu použijeme některou z numerických metod k výpočtu samotného kořene se zadanou přesností.

6.1 Separace kořenů

Pro obecnou funkci je separace kořenů netriviální problém, který často vyžaduje předběžnou znalost chování dané funkce. Pro některé speciální případy, jako jsou algebraické rovnice (polynomy), existují algoritmy (např. Sturmovy posloupnosti), které dokáží počet reálných kořenů v daném intervalu určit přesně. My se však budeme opírat o následující fundamentální větu.

Věta 6.1 (O separaci kořenů)**.** Nechť f je reálná funkce jedné reálné proměnné, která je spojitá na intervalu

$[a, b]$ a nechť platí $f(a)f(b) < 0$. Potom rovnice $f(x) = 0$ má v intervalu (a, b) alespoň jeden kořen. Pokud navíc první derivace $f'(x)$ na intervalu (a, b) nemění znaménko, pak je tento kořen jediný.

Důkaz. Jde o jednoduchou větu z kurzu MAN1, na přednášce nedokazováno. □

Poznámka 6.2 (Obecná ukončovací kritéria)**.** Ukončovací kritéria pro metody hledání kořenů nejsou tak přímočará, jak by se mohlo zdát, a volba nevhodného kritéria může vést k zavádějícím výsledkům. Uvažujme tři nejčastější kandidáty:

1. **Test rozdílu iterací:** $|x_{k+1} - x_k| < \varepsilon$. Tento test jsme používali u lineárních rovnic. Ačkoliv je intuitivní, v nelineárním případě může jednoduše selhat. Pokud se konvergence výrazně zpomalí a metoda se k řešení „plazí“ velmi pomalu, může být rozdíl mezi iteracemi nepatrný, i když jsme stále daleko od skutečného kořene α.
2. **Test rezidua:** $|f(x_k)| < \varepsilon$. Toto kritérium testuje, jak blízko je funkční hodnota nule. Problém nastává, pokud má funkce lokální minimum blízké 0, nikdy ale nepřesáne osu x. V takovém případě se metoda může "zaseknout"v tomto bodě, aniž by skutečně našla kořen.

V praxi se proto často kombinuje více kritérií. Spolehlivou strategií je sledovat jak absolutní hodnotu rezidua $|f(x_k)|$, tak i velikost kroku $|x_{k+1} - x_k|$, a výpočet ukončit, až když jsou obě tyto hodnoty pod zvolenou tolerancí, a zároveň nepřekročíme maximální povolený počet iterací. Pro metody, které garantují sevření kořene v intervalu, jako je bisekce, je nejspolehlivějším kritériem šířka tohoto intervalu.

6.1.1 Metoda půlení intervalu (bisekce)

Metoda bisekce je nejjednodušší, nejrobustnější, ale zároveň nejpomalejší metodou pro výpočet kořene. Její hlavní výhodou je zaručená konvergence, pokud jsou splněny předpoklady z předchozí věty. Metoda konstruuje posloupnost vnořených intervalů $[l_k, r_k]$, které všechny obsahují hledaný kořen α. V každém kroku se interval rozpůlí a pro další iteraci se vybere ta polovina, ve které se kořen nachází.

1. Zvolíme počáteční interval $[l_0, r_0] = [a, b]$, pro který platí $f(a)f(b) < 0$.
2. Pro $k = 0, 1, 2, \ldots$ počítáme střed intervalu:
$$x_k = \frac{l_k + r_k}{2}.$$
3. Nový, zúžený interval $[l_{k+1}, r_{k+1}]$ určíme následovně:
   - Pokud $f(l_k)f(x_k) < 0$, kořen leží v levé polovině: $[l_{k+1}, r_{k+1}] = [l_k, x_k]$.
   - Pokud $f(x_k)f(r_k) < 0$, kořen leží v pravé polovině: $[l_{k+1}, r_{k+1}] = [x_k, r_k]$.
   - Pokud $f(x_k) = 0$, našli jsme přesný kořen a výpočet končí.

Poznámka 6.3 (Konvergence a odhad chyby)**.** Protože se délka intervalu $\mathcal{I}_k = [l_k, r_k]$ v každém kroku půlí, platí $|\mathcal{I}_k| = \frac{b-a}{2^k}$. Jelikož kořen α vždy leží v aktuálním intervalu a x_k je jeho středem, můžeme chybu aproximace v k-tém kroku snadno odhadnout:
$$|\alpha - x_k| \leq |\mathcal{I}_k| = \frac{b-a}{2^k}. \tag{6.1}$$
Slovy tedy říkáme, že vzdálenost kořene od středu intervalu musí být menší než délka intervalu. Mohli bychom také říct, že vzdálenost od středu musí být menší než polovina délky intervalu. S rostoucím k jde chyba k nule, proto metoda vždy konverguje.

Příklad 6.4. Hledejme kořen polynomu $f(x) = x^3 + 4x^2 - 10$ v intervalu $[1, 2]$. Platí $f(1) = -5$ a $f(2) = 14$, takže podmínky jsou splněny. První čtyři iterace jsou:

k	l_k	r_k	$f(l_k)$	$f(r_k)$	x_k	$f(x_k)$
1	1.0	2.0	-5.0	14.0	1.5	2.375
2	1.0	1.5	-5.0	2.375	1.25	-1.796875
3	1.25	1.5	-1.796875	2.375	1.375	0.162109
4	1.25	1.375	-1.796875	0.162109	1.3125	-0.848389

Aproximace kořene po čtyřech iteracích je $x_4 = 1.3125$. Přesná hodnota je $\alpha \approx 1.3652$.

Poznámka 6.5 (Omezení metody)**.** Klíčovým předpokladem pro zaručení konvergence ke kořeni je spojitost funkce $f(x)$ na celém intervalu $[a, b]$. Pokud bychom například hledali řešení rovnice $\tan(x) = 0$ na intervalu $[1, 3]$, metoda bisekce by konvergovala k hodnotě $\pi/2$, což je bod nespojitosti, nikoliv kořen rovnice.

6.1.2 Obecná iterační metoda a podmínky konvergence

Metoda bisekce je sice robustní, ale její konvergence je pomalá. Pro urychlení výpočtu je třeba využít více informací o funkci f, nejen znaménko jejích funkčních hodnot. Základem rychlejších metod je myšlenka, že z aktuální aproximace x_k chceme najít další bod x_{k+1} tak, abychom se co nejlépe přiblížili skutečnému kořeni α. Pokud je funkce f diferencovatelná, můžeme použít Taylorův rozvoj:

$$0 = f(\alpha) = f(x_k) + f'(\xi)(\alpha - x_k), \tag{6.2}$$

kde ξ je nějaký bod mezi x_k a α. Z této rovnice můžeme vyjádřit přesnou hodnotu kořene:

$$\alpha = x_k - \frac{f(x_k)}{f'(\xi)}. \tag{6.3}$$

Problém je, že bod ξ a tedy ani hodnotu $f'(\xi)$ neznáme. Můžeme ji ale aproximovat nějakou hodnotou $q_k \approx f'(\xi)$. Tím získáme obecný předpis pro iterační metody:

$$x_{k+1} = x_k - \frac{f(x_k)}{q_k}. \tag{6.4}$$

Každá volba q_k vede na jinou numerickou metodu. Tento předpis můžeme zapsat ve tvaru $x_{k+1} = \varphi(x_k)$, kde φ nazýváme **iterační funkcí**. Zřejmě pro kořen α platí $\varphi(\alpha) = \alpha$, jedná se tedy o pevný bod zobrazení φ a je splněna podmínka konzistence.

6.1.2.1 Podmínky konvergence

Věta 6.6 (O konvergenci iterační metody)**.** Nechť iterační funkce $\varphi(x)$ splňuje $\varphi(\alpha) = \alpha$. Nechť je φ diferencovatelná na nějakém okolí kořene $V = [\alpha - r, \alpha + r]$ a nechť pro všechna $x \in V$ existuje konstanta $K < 1$ taková, že platí:

$$|\varphi'(x)| \leq K. \tag{6.5}$$

Potom posloupnost $\{x_k\}$ generovaná předpisem $x_{k+1} = \varphi(x_k)$ konverguje k α pro libovolnou počáteční aproximaci $x_0 \in V$.

Důkaz. TODO □

Poznámka 6.7. Pokud je derivace $\varphi'(x)$ spojitá v okolí kořene α, pak z předchozí věty plyne, že postačující podmínkou pro konvergenci je $|\varphi'(\alpha)| < 1$.

Definice 6.8 (Řád konvergence)**.** Řekneme, že iterační metoda daná vztahem $x_{k+1} = \varphi(x_k)$ má **řád konvergence** $m \geq 1$, jestliže pro chybu k-té $x_k - \alpha$ platí:

$$|x_{k+1} - \alpha| \leq C|x_k - \alpha|^m, \tag{6.6}$$

pro nějakou konstantu $C > 0$. Pro $m = 1$ mluvíme o lineární konvergenci, pro $m = 2$ o kvadratické atd.

Poznámka 6.9. Řád konvergence metody úzce souvisí s derivacemi iterační funkce φ v bodě kořene α. Konkrétně platí, že pokud jsou derivace φ na okolí α spojité až do řádu m včetně, až do řádu $m-1$ v bodě α nulové, tj. $\varphi'(\alpha) = \varphi''(\alpha) = \cdots = \varphi^{(m-1)}(\alpha) = 0$, a derivace řádu m je nenulová, $\varphi^{(m)}(\alpha) \neq 0$, pak má metoda řád konvergence právě m. Tento vztah můžeme odvodit pomocí Taylorova rozvoje. Označme chybu v k-té iteraci jako $e_k = x_k - \alpha$. Potom pro chybu v následujícím kroku platí:

$$e_{k+1} = x_{k+1} - \alpha = \varphi(x_k) - \varphi(\alpha)$$

Provedeme-li Taylorův rozvoj funkce $\varphi(x_k)$ v okolí bodu α, dostaneme:

$$e_{k+1} = \varphi'(\alpha)(x_k - \alpha) + \frac{\varphi''(\alpha)}{2!}(x_k - \alpha)^2 + \cdots + \frac{\varphi^{(m)}(\xi)}{m!}(x_k - \alpha)^m$$

kde ξ leží mezi x_k a α. Vzhledem k předpokladu, že první $m-1$ derivace jsou v bodě α nulové, se celý výraz zjednoduší na:

$$e_{k+1} = \frac{\varphi^{(m)}(\xi)}{m!}(x_k - \alpha)^m = \frac{\varphi^{(m)}(\xi)}{m!} e_k^m$$

Pokud přejdeme k absolutním hodnotám, získáme vztah $|e_{k+1}| = \left|\frac{\varphi^{(m)}(\xi)}{m!}\right| |e_k|^m$. Pro $k \to \infty$ se $x_k \to \alpha$, a tedy i $\xi \to \alpha$. Ze spojitosti $\varphi^{(m)}$ pak plyne, že se koeficient $\left|\frac{\varphi^{(m)}(\xi)}{m!}\right|$ blíží ke konstantě $C = \left|\frac{\varphi^{(m)}(\alpha)}{m!}\right|$, což přesně odpovídá definici metody s řádem konvergence m.

Poznámka 6.10 (Ukončovací kritérium)**.** Pokud je $f'(\alpha) \neq 0$, pak v okolí kořene platí, že chyba aproximace $|x_k - \alpha|$ je přibližně úměrná absolutní hodnotě funkce $|f(x_k)|$. Jako praktické kritérium pro zastavení výpočtu se proto často používá podmínka $|f(x_k)| < \epsilon$ pro nějakou malou toleranci ϵ. TODO: doplnit odvození přes Taylorův rozvoj.

6.1.3 Metoda regula falsi (metoda sečen)

Metoda regula falsi (v anglické literatuře často nazývaná "false position method") je první z metod založených na obecném iteračním schématu. Hodnotu derivace $f'(\xi)$ aproximuje směrnicí sečny vedené dvěma posledními známými body. Podobně jako metoda bisekce, i metoda regula falsi vychází z intervalu $[a, b]$, ve kterém leží kořen (tj. $f(a)f(b) < 0$). Princip metody je následující:

1. Body $(l_k, f(l_k))$ a $(r_k, f(r_k))$ proložíme přímku (sečnu). V první iteraci je $[l_0, r_0] = [a, b]$.
2. Další aproximaci kořene x_{k+1} získáme jako průsečík této sečny s osou x.
3. Podle znaménka funkční hodnoty $f(x_{k+1})$ zúžíme interval tak, aby kořen zůstal uvnitř. Nové hranice $[l_{k+1}, r_{k+1}]$ se určí následovně:

$$l_{k+1} := \begin{cases} x_{k+1} & \text{pokud } \operatorname{sgn} f(x_{k+1}) = \operatorname{sgn} f(l_k), \\ l_k & \text{jinak} \end{cases}$$

$$r_{k+1} := \begin{cases} x_{k+1} & \text{pokud } \operatorname{sgn} f(x_{k+1}) = \operatorname{sgn} f(r_k), \\ r_k & \text{jinak} \end{cases}$$

   Tímto postupem je zaručeno, že pro nový interval $[l_{k+1}, r_{k+1}]$ stále platí podmínka $f(l_{k+1})f(r_{k+1}) < 0$. Celý proces se opakuje, dokud není dosaženo požadované přesnosti.

Iterační vzorec pro x_{k+1} odvodíme z rovnice sečny. Nechť jsou dány dva body $(x_k, f(x_k))$ a $(x'_k, f(x'_k))$, kde x'_k je druhý bod, který držíme pro konstrukci sečny (jinými slovy, je to ten z bodů l_k, r_k, který nebyl v předchozím kroku nahrazen).

- Rovnice přímky (sečny) procházející těmito dvěma body má tvar:

$$y(x) = f(x_k) + \frac{f(x_k) - f(x'_k)}{x_k - x'_k}(x - x_k)$$

- Nová aproximace kořene x_{k+1} je průsečíkem této sečny s osou x. Dosadíme tedy $y(x_{k+1}) = 0$:

$$0 = f(x_k) + \frac{f(x_k) - f(x'_k)}{x_k - x'_k}(x_{k+1} - x_k)$$

- Vyřešením rovnice pro x_{k+1} dostaneme iterační vzorec pro metodu regula falsi:

$$x_{k+1} = x_k - \frac{x_k - x'_k}{f(x_k) - f(x'_k)} f(x_k). \tag{6.7}$$

Vidíme, že se jedná o obecnou iterační metodu $x_{k+1} = x_k - \frac{f(x_k)}{q_k}$, kde aproximace derivace je dána směrnicí sečny: $q_k = \frac{f(x_k)-f(x'_k)}{x_k-x'_k}$.

Příklad 6.11. Hledejme kořen polynomu $f(x) = x^3 + 4x^2 - 10$ v intervalu $[1, 2]$.

k	x_k	x'_k	$f(x_k)$
0	1.0	2.0	-5.0
1	1.263157...	2.0	-1.60227...
2	1.337206...	2.0	-0.43036...
3	1.358533...	2.0	-0.08903...
4	1.363547...	2.0	-0.01515...

Metoda konverguje viditelně rychleji než metoda bisekce, ale je patrné, že jeden z krajních bodů (zde bod 2.0) zůstává "zaseknutý".

Věta 6.12. Nechť funkce f je spojitě diferencovatelná na otevřeném okolí kořene α a nechť $f'(\alpha) \neq 0$. Potom existuje takové okolí V_α, že pro libovolné dva startovací body z tohoto okolí, které obklopují kořen, metoda regula falsi konverguje. Řád konvergence je obecně lineární ($m = 1$).

Důkaz. TODO □

6.1.4 Newtonova metoda (metoda tečen)

Newtonova metoda je jednou z nejznámějších a nejpoužívanějších metod pro hledání kořenů nelineárních rovnic. Oproti metodě regula falsi, která aproximuje funkci sečnou, Newtonova metoda využívá tečnu ke grafu funkce. Metoda vychází z počáteční aproximace x_0. V každém kroku k se postupuje následovně:

1. Sestrojíme tečnu ke grafu funkce $f(x)$ v bodě $(x_k, f(x_k))$.
2. Následující aproximaci x_{k+1} definujeme jako průsečík této tečny s osou x.
3. Postup opakujeme, dokud není dosaženo požadované přesnosti.

Rovnice tečny v bodě x_k má tvar:

$$y(x) = f(x_k) + f'(x_k)(x - x_k). \tag{6.8}$$

Hledáme její průsečík s osou x, tedy položíme $y(x_{k+1}) = 0$:

$$0 = f(x_k) + f'(x_k)(x_{k+1} - x_k). \tag{6.9}$$

Z této rovnice vyjádříme x_{k+1} a získáme tak iterační vzorec Newtonovy metody.

Definice 6.13 (Newtonova metoda)**.** Pro danou počáteční aproximaci x_0 jsou další členy posloupnosti gene-

rovány předpisem:

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}. \tag{6.10}$$

Tentokrát jako aproximaci derivace volíme přímo přesnou hodnotu derivace v daném bodě, tj. $q_k = f'(x_k)$. Iterační funkce má tedy tvar $\varphi(x) = x - \frac{f(x)}{f'(x)}$.

Příklad 6.14. Hledejme kořen polynomu $f(x) = x^3 + 4x^2 - 10$ v intervalu $[1, 2]$ s počáteční aproximací $x_0 = 1$. Derivace je $f'(x) = 3x^2 + 8x$.

k	x_k	$f(x_k)$	$f'(x_k)$
0	1.0	-5.0	11.0
1	1.454545...	1.540195...	19.43816...
2	1.375309...	0.167275...	16.67691...
3	1.365279...	0.000816...	16.51419...
4	1.365230...	$\approx 10^{-8}$	16.51339...

Vidíme, že metoda konverguje extrémně rychle. Po pouhých 4 iteracích jsme dosáhli velmi vysoké přesnosti (přesná hodnota je $\alpha \approx 1.365230013$).

Věta 6.15. Nechť funkce f je spojitě diferencovatelná na otevřeném okolí kořene α a nechť $f'(\alpha) \neq 0$. Potom existuje takové okolí

$$V_\alpha := \{x \in \mathbb{R} \,|\, |x - \alpha| < r\} \subset H_\alpha \tag{6.11}$$

že pro libovolné $x_0 \in V_\alpha$ Newtonova metoda konverguje. Řád konvergence je obecně lineární ($m = 1$).

Důkaz. TODO □

Věta 6.16. Pokud v předchozí větě budeme předpokládat, že f je dvakrát spojitě diferencovatelná na otevřeném okolí H_α, potom za stejných předpokladů Newtonova metoda konverguje pro libovolné $x_0 \in V_\alpha$ kvadraticky, tj. $m = 2$.

Poznámka 6.17. Newtonova metoda tedy za standardních podmínek konverguje výrazně rychleji než metoda bisekce nebo regula falsi. Cenou za tuto rychlost je však nutnost znát a počítat derivaci $f'(x)$ a potřeba přesnějšího počátečního odhadu řešení x_0. Okolí konvergence V_α může být pro Newtonovu metodu výrazně menší než u metod, které kořen trvale uzavírají v intervalu.

Poznámka 6.18. Konvergence byla v představených metodách vždy podmíněna tím, že počáteční aproximace x_0 leží v nějakém okolí kořene α. My ale v praxi často neznáme přesnou velikost tohoto okolí. V praxi často volíme počáteční aproximaci x_0 náhodně a doufáme, že se nachází v nějakém okolí kořene. Je dobré ale mít i nějakou podmínku, která zaručí, že pokud metoda nekonverguje, program skončí. Podobně jako jsme tedy dříve volili $|f(x_k)| < \epsilon$ pro rozhodnutí o dostatečné přesnosti, pro ukončení běhu z důvodu nekonvergence používáme např. $|f(x_k)| > C$.

Poznámka 6.19 (Metody vyšších řádů)**.** Lze odvodit i metody ještě vyšších řádů konvergence (např. Čebyševova metoda řádu 3). V praxi se však příliš nepoužívají, protože vyžadují výpočet ještě vyšších derivací funkce f a jsou ještě citlivější na volbu počáteční aproximace. Kvadratická konvergence Newtonovy metody je pro většinu praktických úloh naprosto postačující.

6.1.5 Globálně konvergující metody

Jak jsme viděli, rychlé metody jako je Newtonova často vyžadují velmi dobrou počáteční aproximaci, aby vůbec konvergovaly. Jejich typickým selháním je, že iterační krok je příliš velký, čímž "přestřelí"kořen a další aproximace je horší než ta předchozí. Tento problém řeší tzv. globálně konvergující metody, které se snaží délku kroku regulovat.

Definice 6.20 (Globálně konvergující metoda)**.** Pod pojmem globálně konvergující metody budeme rozumět

takové metody, které pro libovolnou volbu počátečního odhadu x_0 buď konvergují k řešení, nebo algoritmicky selžou (např. detekcí nulové derivace), ale nikdy neutečou do nekonečna ani neoscilují donekonečna.

Jednou ze základních strategií pro zajištění globální konvergence je **metoda zkracování kroku**.

1. V bodě x_k spočítáme směr kroku, například Newtonův krok $d = -f(x_k)/f'(x_k)$.
2. Otestujeme, zda krok tímto směrem a o této délce vede ke zlepšení, tj. zda platí $|f(x_k + d)| < |f(x_k)|$.
3. Pokud ano, přijmeme nový bod $x_{k+1} = x_k + d$.
4. Pokud ne, krok byl příliš dlouhý. Opakovaně ho zkracujeme (např. půlíme $d := d/2$) a testujeme znovu, dokud podmínka zlepšení není splněna.

Tímto postupem je zaručeno, že se v každé iteraci přiblížíme k řešení (alespoň z hlediska zmenšení hodnoty $|f(x)|$). Zmenšování d ale může naopak způsobit, že metoda bude velmi pomalá, zvláště pokud je počáteční odhad x_0 daleko od skutečného řešení α.

6.2 Řešení soustav nelineárních rovnic

Zobecněme nyní naše metody pro případ soustavy n nelineárních rovnic o n neznámých. Hledáme řešení $\vec{\alpha} \in \mathbb{R}^n$ soustavy:

$$\begin{cases} f_1(x_1, \ldots, x_n) = 0 \\ f_2(x_1, \ldots, x_n) = 0 \\ \vdots \\ f_n(x_1, \ldots, x_n) = 0 \end{cases} \quad \text{což lze zapsat vektorově jako} \quad \vec{f}(\vec{x}) = \vec{0}. \tag{6.12}$$

Separace kořenů je v tomto případě ještě výrazně obtížnější než v jedné dimenzi. V dalším textu budeme předpokládat, že máme k dispozici dostatečně dobrou počáteční aproximaci $\vec{x}^{(0)}$.

6.2.1 Newtonova metoda pro soustavy

Newtonovu metodu pro jednu rovnici $x_{k+1} = x_k - [f'(x_k)]^{-1} f(x_k)$ můžeme přímo zobecnit pro soustavy. Roli derivace přebírá Jacobiho matice a roli dělení násobení inverzní maticí.

Definice 6.21 (Jacobiho matice)**.** Jacobiho matice zobrazení $\vec{f}(\vec{x})$ je matice prvních parciálních derivací:

$$(\mathbb{J}_{\vec{f}}(\vec{x}))_{ij} = \frac{\partial f_i(\vec{x})}{\partial x_j}. \tag{6.13}$$

Definice 6.22 (Newtonova metoda pro soustavy)**.** Pro danou počáteční aproximaci $\vec{x}^{(0)}$ jsou další členy posloupnosti generovány předpisem:

$$\vec{x}^{(k+1)} = \vec{x}^{(k)} - [\mathbb{J}_{\vec{f}}(\vec{x}^{(k)})]^{-1} \vec{f}(\vec{x}^{(k)}). \tag{6.14}$$

Poznámka 6.23 (Praktická implementace)**.** Výpočet inverzní matice v každém kroku je výpočetně náročný. Proto se předchozí rovnice přepisuje do tvaru, který místo inverze vyžaduje řešení soustavy lineárních rovnic:

$$\mathbb{J}_{\vec{f}}(\vec{x}^{(k)})(\vec{x}^{(k+1)} - \vec{x}^{(k)}) = -\vec{f}(\vec{x}^{(k)}). \tag{6.15}$$

Označíme-li si přírůstek $\Delta\vec{x}^{(k)} = \vec{x}^{(k+1)} - \vec{x}^{(k)}$, můžeme jednu iteraci Newtonovy metody rozdělit do dvou kroků:

1. **Vyřeš lineární soustavu:** $\mathbb{J}_{\vec{f}}(\vec{x}^{(k)})\Delta\vec{x}^{(k)} = -\vec{f}(\vec{x}^{(k)})$ pro neznámý vektor $\Delta\vec{x}^{(k)}$.
2. **Aktualizuj řešení:** $\vec{x}^{(k+1)} = \vec{x}^{(k)} + \Delta\vec{x}^{(k)}$.

Výhodou je, že pro řešení lineární soustavy můžeme použít efektivní iterační metody a nemusíme ji řešit s

maximální přesností, zvláště v počátečních fázích výpočtu.

Lemma 6.24. Nechť funkce $f \in \mathbb{C}^1(H)$, kde H je konvexní oblast. Potom pro každé $\vec{u}, \vec{v} \in H$ existuje $\vec{\epsilon} \in H$ takové, že

$$f(\vec{u}) - f(\vec{v}) = \nabla f(\vec{\epsilon}) \cdot (\vec{u} - \vec{v}). \tag{6.16}$$

Důkaz. TODO □

Věta 6.25 (Konvergence Newtonovy metody pro soustavy)**.** Nechť pro kořen $\vec{\alpha}$ soustavy $\vec{f}(\vec{x}) = \vec{0}$ existuje otevřené okolí $H_{\vec{\alpha}}$ takové, že :

- zobrazení $\vec{f}$ je na $H_{\vec{\alpha}}$ spojitě diferencovatelné,
- Jacobiho matice v bodě kořene $\mathbb{J}_{\vec{f}}(\vec{\alpha})$ je regulární.

Pak existuje takové okolí

$$V_{\vec{\alpha}} \coloneqq \{\vec{x} \in \mathbb{R}^n \mid ||\vec{x} - \vec{\alpha}|| < r\} \subset H_{\vec{\alpha}}, \tag{6.17}$$

že pro libovolnou počáteční aproximaci $\vec{x}^{(0)} \in V_{\vec{\alpha}}$ Newtonova metoda konverguje, a to s řádem konvergence $m = 1$.

Důkaz. TODO □

Věta 6.26. Pokud předpoklady věty výše zpřísníme tak, že požadujeme aby $\vec{f}$ byla na $H_{\vec{\alpha}}$ dvakrát spojitě diferencovatelná, potom Newtonova metoda konverguje pro libovolnou počáteční aproximaci $\vec{x}^{(0)} \in V_{\vec{\alpha}}$ kvadraticky, tj. s řádem konvergence $m = 2$.

Důkaz. Nepřednáší se. □

Poznámka 6.27. I na Newtonovou metodu pro soustavy nelineárních rovnic lze metodu zkracování kroku.

6.3 Otázky

- metoda bisekcí
- metoda regula falsi
- Newtonova metoda
- jak se při praktických výpočtech liší podmínky konvergence (okolí u bisekcí vs regula falsi)
- jaké jsou výhody a nevýhody metod vyšších řádů
- globálně konvergující metody
- Newtonova metoda pro systémy nelineárních rovnic, jak se dá efektivně vyhnout napočítávání inverzní matice

Kapitola 7

Numerická interpolace funkcí

V numerické matematice často pracujeme s funkcemi, které nelze vyjádřit analyticky, nebo známe jejich hodnoty pouze v několika diskrétních bodech, například z experimentálního měření. V takových případech je nutné funkci aproximovat jinou, jednodušší funkcí. Pro svou jednoduchost, snadné derivování a integrování se nejčastěji volí polynomiální aproximace.

7.1 Interpolační polynom

Jednou z možností, jak aproximovat funkci, je Taylorův polynom. Ten však vyžaduje znalost derivací funkce v jednom bodě, což je v případě experimentálních dat jen zřídka možné. My se naopak zaměříme na konstrukci polynomu, který prochází několika zadanými body (uzly) s danými funkčními hodnotami.

Poznámka 7.1 (Matematická formulace problému)**.** Mějme funkci $f : \mathbb{R} \to \mathbb{R}$, jejíž hodnoty známe v $n+1$ vzájemně různých bodech $x_0, x_1, \ldots, x_n$. Hledáme polynom $L_n(x)$ co nejnižšího stupně tak, aby platilo:

$$L_n(x_i) = f(x_i) \quad \text{pro } i = 0, \ldots, n. \tag{7.1}$$

Za vhodných podmínek můžeme doufat, že $L_n(x)$ bude dobrou aproximací funkce $f(x)$ i v ostatních bodech. Této úloze říkáme interpolace.

7.1.1 Obecná konstrukce a existence

Hledáme-li polynom ve tvaru $L_n(x) = \sum_{i=0}^{n} a_i x^i$, pak $n+1$ interpolačních podmínek vede na soustavu $n+1$ lineárních rovnic pro neznámé koeficienty $a_0, \ldots, a_n$:

$$\begin{pmatrix} 1 & x_0 & x_0^2 & \ldots & x_0^n \\ 1 & x_1 & x_1^2 & \ldots & x_1^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \ldots & x_n^n \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} f(x_0) \\ f(x_1) \\ \vdots \\ f(x_n) \end{pmatrix}. \tag{7.2}$$

Tuto soustavu můžeme zapsat jako $\mathbb{V}\vec{a} = \vec{f}$.

Definice 7.2 (Vandermondova matice)**.** Matice $\mathbb{V}$ z předchozí soustavy se nazývá Vandermondova matice.

Věta 7.3. Jsou-li body $x_0, \ldots, x_n \in \mathbb{R}$ navzájem různé, pak je příslušná Vandermondova matice $\mathbb{V}$ regulární.

Důkaz. TODO □

Věta 7.4 (O existenci a jednoznačnosti interpolačního polynomu)**.** Nechť jsou dány body $x_0, \ldots, x_n \in D_f$, které jsou navzájem různé. Pak existuje právě jeden polynom $P(x)$ stupně nejvýše n, který splňuje interpolační

podmínky $P(x_i) = f(x_i)$ pro všechna $i = 0, \ldots, n$.

Důkaz. Důkaz plyne z regularity Vandermondovy matice, která zaručuje jednoznačné řešení soustavy pro koeficienty a_i. □

Poznámka 7.5. Interpolace polynomem stupně nižšího než n obecně není možná, protože bychom dostali přeurčenou soustavu lineárních rovnic, která obecně nemá řešení. Naopak interpolace polynomem stupně vyššího než n není jednoznačná, protože soustava pro koeficienty by byla nedourčená a měla by nekonečně mnoho řešení.

Poznámka 7.6. Přímé řešení soustavy s Vandermondovou maticí se v praxi nepoužívá, protože tato matice bývá často špatně podmíněná, což vede k numerické nestabilitě. Proto se pro konstrukci téhož (jednoznačně určeného) polynomu používají jiné, numericky stabilnější tvary.

7.1.2 Lagrangeův tvar interpolačního polynomu

Lagrangeův tvar je založen na konstrukci speciálních bázových polynomů.

Definice 7.7 (Lagrangeovy bázové polynomy)**.** Pro $i = 0, \ldots, n$ definujeme Lagrangeovy bázové polynomy $l_i(x)$ stupně n vztahem:

$$l_i(x) = \prod_{j=0, j\neq i}^{n} \frac{x - x_j}{x_i - x_j}. \tag{7.3}$$

Tyto polynomy mají klíčovou vlastnost $l_i(x_j) = \delta_{ij}$ (pro $i = j$ je hodnota 1, pro $i \neq j$ je 0).

Definice 7.8 (Lagrangeův tvar polynomu)**.** Lagrangeův interpolační polynom je pak definován jako lineární kombinace těchto bázových polynomů:

$$L_n(x) = \sum_{i=0}^{n} f(x_i) l_i(x). \tag{7.4}$$

Díky vlastnosti bázových polynomů je zřejmé, že $L_n(x_i) = f(x_i)$.

Příklad 7.9. Nalezněme Lagrangeův interpolační polynom pro body $(-1, 1), (0, 0), (1, 1)$. Máme tedy $n = 2$, $x_0 = -1, x_1 = 0, x_2 = 1$. Bázové polynomy jsou:

$$\begin{aligned} l_0(x) &= \frac{(x-0)(x-1)}{(-1-0)(-1-1)} = \frac{1}{2}x(x-1) \\ l_1(x) &= \frac{(x-(-1))(x-1)}{(0-(-1))(0-1)} = -(x+1)(x-1) \\ l_2(x) &= \frac{(x-(-1))(x-0)}{(1-(-1))(1-0)} = \frac{1}{2}x(x+1) \end{aligned}$$

Výsledný polynom je:

$$L_2(x) = 1 \cdot l_0(x) + 0 \cdot l_1(x) + 1 \cdot l_2(x) = \frac{1}{2}x(x-1) + \frac{1}{2}x(x+1) = x^2.$$

Poznámka 7.10 (Složitost výpočtu)**.** Lagrangeův tvar polynomu je elegantní, ale pro praktický výpočet nepříliš vhodný. Pro vyčíslení hodnoty polynomu v novém bodě x je nutné znovu vypočítat všech $n + 1$ bázových polynomů (protože jsou závislé na x), přičemž každý vyžaduje n násobení. Dohromady máme složitost $O(n^2)$. Efektivnější způsob konstrukce i vyčíslení nabízí Newtonova formule.

7.1.3 Newtonova formule

Newtonův tvar interpolačního polynomu představuje efektivnější způsob jeho konstrukce a vyhodnocování. Je založen na rekurzivní myšlence, kdy polynom vyššího stupně vzniká úpravou polynomu nižšího stupně.

Předpokládejme, že známe interpolační polynom $L_{k-1}(x)$ stupně nejvýše $k-1$, který prochází body

$$(x_0, f(x_0)), \ldots, (x_{k-1}, f(x_{k-1})). \tag{7.5}$$

Polynom $L_k(x)$ pro $k+1$ bodů můžeme sestrojit tak, že k $L_{k-1}(x)$ přičteme korekční člen:

$$L_k(x) = L_{k-1}(x) + c_k(x - x_0)(x - x_1)\ldots(x - x_{k-1}). \tag{7.6}$$

Tento nový polynom stále správně interpoluje prvních k bodů, protože přidaný člen je v bodech $x_0, \ldots, x_{k-1}$ nulový. Neznámý koeficient c_k určíme z poslední interpolační podmínky $L_k(x_k) = f(x_k)$:

$$f(x_k) = L_{k-1}(x_k) + c_k(x_k - x_0)(x_k - x_1)\ldots(x_k - x_{k-1}). \tag{7.7}$$

Odtud můžeme c_k přímo vyjádřit:

$$c_k = \frac{f(x_k) - L_{k-1}(x_k)}{(x_k - x_0)(x_k - x_1)\ldots(x_k - x_{k-1})}. \tag{7.8}$$

Opakovaným rozvojem této rekurze získáme finální tvar polynomu.

Definice 7.11 (Newtonův tvar polynomu)**.** Newtonův tvar interpolačního polynomu stupně n je dán vztahem:

$$L_n(x) = c_0 + c_1(x - x_0) + c_2(x - x_0)(x - x_1) + \cdots + c_n(x - x_0)\ldots(x - x_{n-1}), \tag{7.9}$$

což lze zapsat jako

$$L_n(x) = \sum_{i=0}^{n} c_i \prod_{j=0}^{i-1}(x - x_j). \tag{7.10}$$

Podívejme se podrobněji, jaký význam má přidaný člen $c_k(x - x_0)...(x - x_{k-1})$.

- Polynom stupně nejvýše 0, který interpoluje první bod $(x_0, f(x_0))$, je zjevně konstanta:

  $$L_0(x) = f(x_0).$$

  Pro koeficient c_0 tedy platí $c_0 = f(x_0)$.

- Nyní hledejme polynom $L_1(x)$ stupně nejvýše 1, který navíc prochází bodem $(x_1, f(x_1))$. Sestrojíme ho přičtením lineární korekce k $L_0(x)$, která nezmění hodnotu v bodě x_0:

  $$L_1(x) = L_0(x) + c_1(x - x_0) = c_0 + c_1(x - x_0).$$

  Neznámý koeficient c_1 určíme z interpolační podmínky $L_1(x_1) = f(x_1)$:

  $$\begin{aligned} f(x_1) &= c_0 + c_1(x_1 - x_0) \\ f(x_1) &= f(x_0) + c_1(x_1 - x_0) \\ c_1 &= \frac{f(x_1) - f(x_0)}{x_1 - x_0}. \end{aligned}$$

Vidíme, že koeficient c_1 je směrnice sečny procházející body $(x_0, f(x_0))$ a $(x_1, f(x_1))$. Přidaný člen $c_1(x - x_0)$ je tedy korekční člen, jehož úkolem je „napravit“ hodnotu konstantního polynomu $L_0(x)$ tak, aby v bodě x_1 byla splněna interpolační podmínka. V ostatních bodech hodnotu polynomu nezmění, neboť je tam nulový.

7.1.3.1 Poměrné diference

Přímý výpočet koeficientů c_k z rekurzivního vztahu by byl zdlouhavý. Ukazuje se však, že je můžeme efektivně určit, pokud z podmínek $L_n(x_i) = f(x_i)$ pro $i = 0, \ldots, n$ sestavíme soustavu lineárních rovnic. Tuto soustavu

lze zapsat v maticovém tvaru $\mathbb{B}\vec{c} = \vec{f}$. Matice $\mathbb{B}$ je přitom definována vztahy:

$$\mathbb{B}_{ij} = \begin{cases} \prod_{k=0}^{j-1}(x_i - x_k) & \text{pro } j \leq i, \\ 0 & \text{pro } j > i \end{cases} \tag{7.11}$$

Celá soustava má tedy následující podobu:

$$\begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ 1 & x_1 - x_0 & 0 & \cdots & 0 \\ 1 & x_2 - x_0 & (x_2 - x_0)(x_2 - x_1) & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n - x_0 & (x_n - x_0)(x_n - x_1) & \cdots & \prod_{k=0}^{n-1}(x_n - x_k) \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \\ c_2 \\ \vdots \\ c_n \end{pmatrix} = \begin{pmatrix} f(x_0) \\ f(x_1) \\ f(x_2) \\ \vdots \\ f(x_n) \end{pmatrix} \tag{7.12}$$

Z tvaru matice $\mathbb{B}$ je zřejmé, že se jedná o dolní trojúhelníkovou matici. Tuto soustavu lze tedy snadno řešit dopřednou substitucí. Z této struktury přímo vyplývá, že:

- koeficient c_0 závisí pouze na hodnotě $f(x_0)$,
- koeficient c_1 závisí pouze na hodnotách $f(x_0)$ a $f(x_1)$,
- obecně koeficient c_k závisí pouze na hodnotách funkce v bodech $x_0, \ldots, x_k$.

Díky této vlastnosti můžeme pro koeficienty zavést speciální značení $c_k = f[x_0, \ldots, x_k]$, které nazýváme poměrnou diferencí. Newtonův polynom pak můžeme psát jako:

$$L_n(x) = f[x_0] + f[x_0, x_1](x - x_0) + \cdots + f[x_0, \ldots, x_n] \prod_{j=0}^{n-1}(x - x_j). \tag{7.13}$$

Definice 7.12 (Poměrná diference)**.** Výraz $f[x_i, \ldots, x_{i+k}]$ nazýváme **poměrnou diferencí (divided difference)** k-tého řádu. Poměrnou diferenci nultého řádu definujeme jako funkční hodnotu:

$$f[x_i] = f(x_i). \tag{7.14}$$

Věta 7.13. Pro koeficienty v Newtonově formuli (poměrné diference) platí následující rekurzivní vztah:

$$f[x_i, \ldots, x_{i+k}] = \frac{f[x_{i+1}, \ldots, x_{i+k}] - f[x_i, \ldots, x_{i+k-1}]}{x_{i+k} - x_i}. \tag{7.15}$$

Důkaz. TODO □

Poměrné diference vyšších řádů se počítají rekurzivně z diferencí nižších řádů. se tedy na základě předchozí věty počítají rekurzivně z diferencí nižších řádů.

$$\begin{aligned} f[x_0, x_1] &= \frac{f[x_1] - f[x_0]}{x_1 - x_0} \\ f[x_1, x_2] &= \frac{f[x_2] - f[x_1]}{x_2 - x_1} \\ f[x_0, x_1, x_2] &= \frac{f[x_1, x_2] - f[x_0, x_1]}{x_2 - x_0} \\ f[x_1, x_2, x_3] &= \frac{f[x_2, x_3] - f[x_1, x_2]}{x_3 - x_1} \end{aligned}$$

Tento rekurzivní výpočet lze přehledně uspořádat do tabulky poměrných diferencí. V prvním sloupci jsou uzlové body x_i, ve druhém funkční hodnoty (diference 0. řádu). Každý další prvek v tabulce se vypočítá jako podíl

rozdílu dvou sousedních prvků z předchozího sloupce a rozdílu odpovídajících krajních uzlových bodů.

$$\begin{array}{c|cccccc}
x_0 & f[x_0] & & & & & \\
 & & f[x_0,x_1] & & & & \\
x_1 & f[x_1] & & f[x_0,x_1,x_2] & & & \\
 & & f[x_1,x_2] & & \ddots & & \\
x_2 & f[x_2] & & f[x_1,x_2,x_3] & & & \\
 & & f[x_2,x_3] & & & & \\
x_3 & f[x_3] & & \vdots & & & \\
\vdots & \vdots & \vdots & & & f[x_0,\ldots,x_n] & \\
x_{n-1} & f[x_{n-1}] & & & & & \\
 & & f[x_{n-1},x_n] & & & & \\
x_n & f[x_n] & & & & &
\end{array}$$

Koeficienty $c_k = f[x_0, \ldots, x_k]$ hledaného Newtonova interpolačního polynomu jsou pak přímo prvky na horní diagonále této tabulky, tedy $f[x_0], f[x_0, x_1], f[x_0, x_1, x_2], \ldots, f[x_0, \ldots, x_n]$.

Příklad 7.14. Nalezněme znovu interpolační polynom pro body $(-1, 1), (0, 0), (1, 1)$ pomocí Newtonovy formule. Sestavíme tabulku poměrných diferencí:

x_i	$f[x_i]$	$f[x_i, x_{i+1}]$	$f[x_i, x_{i+1}, x_{i+2}]$
$x_0 = -1$	$\mathbf{1}$		
		$\frac{0-1}{0-(-1)} = \mathbf{-1}$	
$x_1 = 0$	0		$\frac{1-(-1)}{1-(-1)} = \mathbf{1}$
		$\frac{1-0}{1-0} = 1$	
$x_2 = 1$	1		

Koeficienty pro Newtonův polynom jsou hodnoty na horní diagonále: $c_0 = 1, c_1 = -1, c_2 = 1$. Dosadíme je do Newtonovy formule:

$$\begin{aligned}
L_2(x) &= c_0 + c_1(x - x_0) + c_2(x - x_0)(x - x_1) \\
&= 1 + (-1)(x - (-1)) + 1(x - (-1))(x - 0) \\
&= 1 - (x + 1) + (x + 1)x = 1 - x - 1 + x^2 + x = x^2.
\end{aligned}$$

Dostali jsme stejný polynom jako při použití Lagrangeova tvaru.

Poznámka 7.15 (Složitost vyhodnocení)**.** Při naivním vyhodnocování Newtonova polynomu by se pro každý člen zvlášť počítal součin $(x - x_0) \ldots (x - x_{k-1})$, což by vedlo ke složitosti $O(n^2)$. Tuto složitost lze ale snadno snížit na $O(n)$ využitím faktu, že součin pro $(k + 1)$-ní člen obsahuje celý součin pro k-tý člen. Při výpočtu hodnoty polynomu $L_n(x)$ si zavedeme pomocnou proměnnou t, ve které budeme iterativně držet hodnotu součinu.

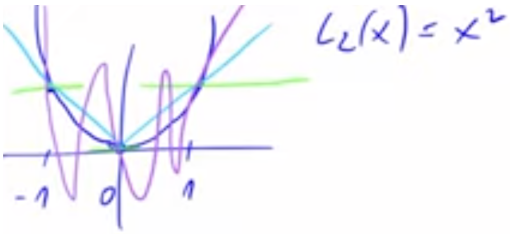
1. Inicializujeme výsledek $y = c_0$ a pomocnou proměnnou $t = 1$.
2. Pro $k = 1, \ldots, n$ postupně provádíme:
   - Aktualizujeme součin: $t := t \cdot (x - x_{k-1})$.
   - Přičteme další člen k výsledku: $y := y + c_k \cdot t$.

Každý krok cyklu vyžaduje pouze konstantní počet operací (jedno násobení, jedno odčítání, jedno násobení a jedno sčítání). Jelikož cyklus proběhne n-krát, je celková složitost vyhodnocení polynomu lineární, tedy $O(n)$.

7.2 Analýza interpolačního polynomu

7.2.1 Chyba aproximace

Klíčovou otázkou je, jak dobře interpolační polynom $L_n(x)$ aproximuje původní funkci $f(x)$ v bodech, které nebyly použity pro interpolaci. Když se totiž podíváme na určování interpolačního polynomu procházejícího body $(-1,1),(0,0),(1,1)$ v předchozích příkladech, vidíme, že ač jsme dostali jednoznačně určený polynom $L_2(x) = x^2$, funkcí procházející těmito body je obecně nekonečně mnoho (viz obrázek). Abychom tedy mohli mluvit o interpolaci, je potřeba o funkci předpokládat něco víc. Následující věta nám říká, že pokud je samotná



Obrázek 7.1: existence nekonečně mnoha funkcí procházejících třemi danými body

funkce blízká polynomu, pak je interpolační polynom dobrá aproximace.

Věta 7.16 (O chybě interpolačního polynomu)**.** Nechť f je funkce, která má na intervalu I_x, obsahujícím body $x, x_0, \ldots, x_n$, spojitou derivaci řádu $n+1$. Nechť $L_n(x)$ je interpolační polynom příslušný k funkci f a bodům $x_0, \ldots, x_n$. Pak pro chybu interpolace $R_n(x) = f(x) - L_n(x)$ existuje bod $\xi \in I_x$ takový, že platí:

$$R_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!}\omega_n(x), \tag{7.16}$$

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

Důkaz. TODO □

Poznámka 7.17. Je důležité si všimnout, že zatímco pro samotnou konstrukci interpolačního polynomu (např. v Lagrangeově nebo Newtonově tvaru) nepotřebujeme znát žádné derivace funkce f, pro analýzu jeho chyby a pro ospravedlnění, proč by měl být dobrou aproximací, je existence $(n+1)$-ní derivace klíčová. Pokud tato derivace neexistuje nebo není omezená, nelze zaručit, že se chyba $R_n(x)$ bude s rostoucím n zmenšovat.

Poznámka 7.18. Dalším důležitým pozorováním je, že chyba není jen odhadem, ale přímo je nabývána rovnost. Nabízela by se otázka, zda můžeme interpolační polynom o tuto chybu zlepšit. Problém je v tom, že chyba je vyjádřena v bodě $\xi \in I_x$, a my nevíme, kde tento bod leží.

Poznámka 7.19. Všimněme si také, že člen $\omega_n(x)$ nám říká, že chyba interpolace je nulová v uzlových bodech $x_0, \ldots, x_n$. To je logické, protože v těchto bodech interpolační polynom přesně odpovídá funkční hodnotě $f(x_i)$.

Chybu interpolace lze vyjádřit také pomocí poměrných diferencí.

Věta 7.20. Za stejných předpokladů jako v předchozí větě platí pro chybu interpolace alternativní vztah:

$$R_n(x) = f[x_0, x_1, \ldots, x_n, x] \cdot \omega_n(x). \tag{7.17}$$

Důkaz. TODO □

Poznámka 7.21 (Srovnání s Taylorovým polynomem)**.** Porovnáním obou tvarů chyby dostáváme důležitý vztah mezi poměrnou diferencí a derivací:

$$f[x_0, x_1, \ldots, x_n, x] = \frac{f^{(n+1)}(\xi)}{(n+1)!}. \tag{7.18}$$

Vzpomeneme-li si dále na jeden z možných tvarů zápisu Lagrangeova polynomu:

$$L_n(x) = f[x_0] + f[x_0, x_1](x - x_0) + \cdots + f[x_0, \ldots, x_n] \prod_{j=0}^{n-1}(x - x_j). \tag{7.19}$$

pokud do něj dosadíme, dostáváme:

$$L_n(x) = \sum_{i=0}^{n} \frac{f^{(i)}(\xi_i)}{i!} \prod_{j=0}^{i-1}(x - x_j), \tag{7.20}$$

kde $\xi_i \in [x_0, x_i]$ (TODO: proč). Z tohoto vztahu je vidět hluboká podobnost mezi Newtonovým tvarem interpolačního polynomu a Taylorovým polynomem,

$$T_n(x) = \sum_{i=0}^{n} \frac{f^{(i)}(x_0)}{i!}(x - x_0)^i. \tag{7.21}$$

Zatímco Taylorův polynom používá derivace vyčíslené v jediném bodě x_0, Newtonův polynom je používá ve smyslu poměrných diferencí, které odpovídají derivacím v různých, předem neznámých bodech ξ_i. To stejné platí i pro celou chybu $\mathbb{R}_n(x)$, která připomíná Lagrangeův tvar zbytku Taylorova polynomu. V něm ale máme v součinu $(x - x_0)^n$, v našem případě je to $\omega_n(x) = \prod_{i=0}^{n}(x - x_i)$.

7.2.2 Řád aproximace

Definice 7.22 (Řád aproximace)**.** Řekneme, že funkce $g(x)$ aproximuje funkci $f(x)$ na okolí bodu x_0 s **přesností řádu** r, právě když platí

$$\lim_{x \to x_0} \frac{|f(x) - g(x)|}{|x - x_0|^r} = C, \tag{7.22}$$

kde C je kladná nenulová konstanta.

Poznámka 7.23. Aplikujeme-li tuto definici na chybu Lagrangeovy interpolace[1] v jednom z uzlových bodů, například x_0, dostáváme:

$$\lim_{x \to x_0} \frac{|R_n(x)|}{|x - x_0|} = \lim_{x \to x_0} \left| \frac{f^{(n+1)}(\xi(x))}{(n+1)!} \prod_{i=1}^{n}(x - x_i) \right| = \left| \frac{f^{(n+1)}(\xi(x_0))}{(n+1)!} \prod_{i=1}^{n}(x_0 - x_i) \right| = C.$$

To znamená, že v okolí[2] interpolačních bodů je Lagrangeův polynom aproximací prvního řádu ($r = 1$), a to nezávisle na počtu bodů n. Zvyšování počtu bodů tedy nezaručuje lepší aproximaci "všude". Zvyšování n má smysl pouze tehdy, pokud je hodnota $(n+1)$-ní derivace na celém intervalu velmi malá[3], tj. pokud se funkce $f(x)$ chová "podobně jako polynom[4]". V opačném případě může zvyšování stupně polynomu vést ke zhoršení aproximace a k oscilacím, což je jev známý jako Rungův fenomén.

7.2.3 Rungův jev

Z analýzy chyby interpolačního polynomu by se mohlo zdát, že pro lepší aproximaci stačí jednoduše zvýšit počet interpolačních uzlů n. To by vedlo k vyšší mocnině v členu $\omega_n(x)$, ale také k vyššímu řádu derivace $f^{(n+1)}(\xi)$, jejíž velikost může růst. Ukazuje se, že zvyšování stupně polynomu může přesnost aproximace naopak dramaticky zhoršit.

Poznámka 7.24 (Rungův jev)**.** Rungův jev popisuje situaci, kdy při interpolaci funkce na intervalu s ekvidistantně rozmístěnými uzly dochází s rostoucím stupněm polynomu n k výrazným oscilacím poblíž krajů inter-

[1] Důležité je zde vzpomenout si, že $\xi \in I_x$, a tedy ξ závisí při aplikaci limity na hodnotě x.

[2] a jen na něm, jinde o vztahu funkce a polynomu nic nevíme

[3] Když se vrátíme ke znění věty o chybě interpolačního polynomu, to že hodnota $n+1$-ní derivace bude velmi malá nám pomůže mitigovat oscilaci, které vytváří zvýšení stupně ω_n.

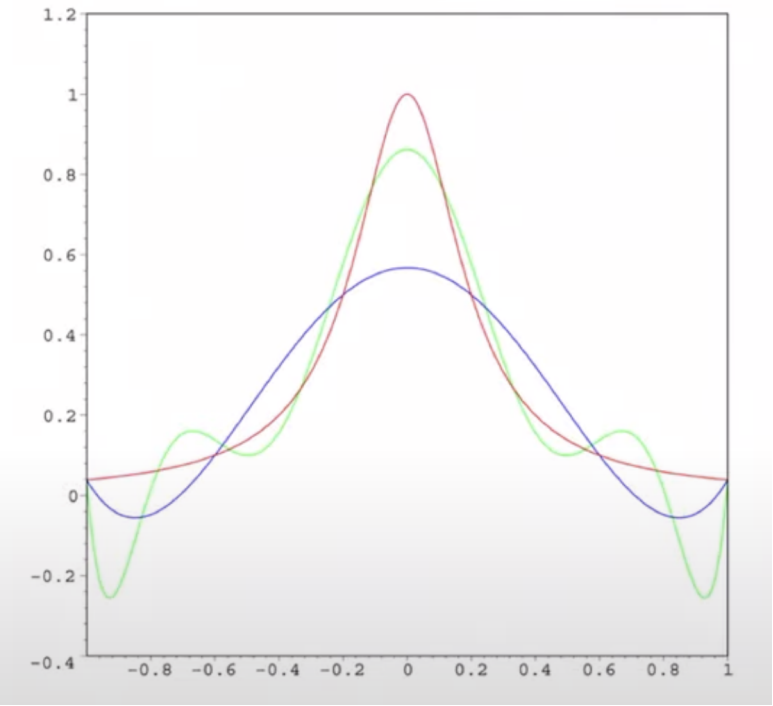
[4] Polynom má totiž od určitého n výše nulové všechny derivace.

valu. Tyto oscilace jsou způsobeny rychlým růstem hodnoty polynomu $\omega_n(x)$ mimo střed intervalu. Důsledkem je, že chyba aproximace $|f(x) - L_n(x)|$ v těchto oblastech neroste k nule, ale naopak se může zvětšovat.

Příklad 7.25. Klasickým příkladem je interpolace tzv. Rungovy funkce $f(x) = \frac{1}{1+25x^2}$ na intervalu $[-1, 1]$ s ekvidistantně rozloženými uzly $x_i = \frac{2i}{n} - 1$ pro $i \in \hat{n}$. Lze ukázat, že v tomto případě platí:

$$\lim_{n\to\infty} \left(\max_{x\in[-1,1]} |f(x) - L_n(x)| \right) = +\infty.$$

Následující obrázek ilustruje tento jev. Zatímco polynom 5. stupně (modře) funkci aproximuje rozumně, polynom 9. stupně (zeleně) již na krajích intervalu silně osciluje a od původní funkce (červeně) se výrazně odchyluje.



Obrázek 7.2: Rungův jev při interpolaci Rungovy funkce

7.3 Interpolace po částech

Dostáváme se do sporné situace: Lagrangeův polynom je obecně dobrou aproximací pouze v okolí zadaných uzlů, ale přidáním dalších uzlů se kvůli Rungovu jevu může celková chyba na intervalu naopak zvýšit. Řešením tohoto problému je **interpolace po částech**. Princip spočívá v tom, že původní interval $[x_0, x_n]$ rozdělíme na několik menších podintervalů. Na každém z těchto podintervalů pak sestrojíme interpolační polynom nízkého stupně (např. lineární nebo kubický) s využitím pouze uzlů, které leží v daném podintervalu. Výsledná aproximace je pak "slepena"z těchto polynomiálních kousků. Pokud krajní body podintervalů tvoří zadané uzly, je výsledná funkce spojitá. Obecně však není v místech napojení diferencovatelná.

7.3.1 Hermitova-Birkhoffova interpolace*

Chceme-li dosáhnout hladšího navázání jednotlivých interpolačních polynomů (tj. shody nejen ve funkčních hodnotách, ale i v derivacích), musíme použít obecnější typ interpolace.

Definice 7.26 (Hermitova-Birkhoffova interpolace)**.** Mějme funkci f, pro kterou v zadaných uzlech $x_0, \ldots, x_n$ známe nejen funkční hodnoty, ale i hodnoty jejích derivací až do řádu m_i. Hledáme polynom $H_{N-1}(x)$ stupně

nejvýše $N-1$, kde $N = \sum_{i=0}^{n}(m_i+1)$ je celkový počet zadaných podmínek, takový, že platí:

$$H_{N-1}^{(k)}(x_i) = f^{(k)}(x_i) \quad \text{pro } i = 0,\dots,n \text{ a } k = 0,\dots,m_i. \tag{7.23}$$

Lze ukázat, že takový polynom existuje a je jednoznačně určen.

Věta 7.27 (O chybě Hermitovy interpolace)**.** Nechť funkce f má na intervalu I_x obsahujícím body $x, x_0, \dots, x_n$ spojitou derivaci řádu N. Pak pro chybu Hermitovy interpolace platí:

$$f(x) - H_{N-1}(x) = \frac{f^{(N)}(\xi)}{N!}\prod_{i=0}^{n}(x-x_i)^{m_i+1}, \tag{7.24}$$

kde $\xi \in I_x$.

7.4 Interpolace ve vyšších dimenzích

Poznámka 7.28 (Interpolace v $\mathbb{R}^2$)**.** Myšlenku Lagrangeovy interpolace lze rozšířit i do vyšších dimenzí. Například pro funkci dvou proměnných $f(x,y)$ a data zadaná na pravoúhlé síti bodů (x_i, y_j) můžeme zkonstruovat 1D Lagrangeovy bázové polynomy pro každou souřadnici zvlášť:

$$l_i^x(x) = \prod_{k=0,k\neq i}^{n}\frac{x-x_k}{x_i-x_k}$$
$$l_j^y(y) = \prod_{k=0,k\neq j}^{m}\frac{y-y_k}{y_j-y_k}$$

Interpolační polynom je pak dán tenzorovým součinem těchto bází:

$$L_{n,m}(x,y) = \sum_{i=0}^{n}\sum_{j=0}^{m} f(x_i,y_j)l_i^x(x)l_j^y(y). \tag{7.25}$$

7.4.1 Otázky

- Lagrangeův polynom: konstrukce, existence, jednoznačnost
- Newtonova formule
- chyba aproximace: pokud sestrojím polynom, $L_4(x)$ k funkci f, ktera není diferencovatelná, co lze říct o tom, jak polynom $L_4(x)$ bude aproximovat funkci f? (nic)
- jak't je řád aproximace funkce f Lagrangeovým polynomem
- podle čeho volit stupeň Lagrangeova polynomu
- interpolace funkce po částech a navazování Lagrangeových polynomů
- interpolace s vyšším řádem přesnosti: jen existence a obecný princip

Kapitola 8

Numerický výpočet derivace

V této kapitole se budeme zabývat úlohou numerického výpočtu derivace funkce. Často se stává, že funkci, kterou chceme derivovat, neznáme analyticky, ale máme k dispozici pouze její hodnoty v několika diskrétních bodech (uzlech). Cílem je z těchto dat aproximovat hodnotu derivace v některém z těchto bodů, případně i jinde. Numerický výpočet derivace je klíčovou součástí mnoha jiných numerických metod, například:

- v Newtonově metodě pro řešení nelineárních rovnic,
- při numerickém řešení obyčejných i parciálních diferenciálních rovnic.

8.1 Výpočet derivace pomocí interpolačního polynomu

Základní myšlenka numerické derivace je jednoduchá: pokud umíme funkci $f(x)$ dobře aproximovat interpolačním polynomem $L_n(x)$, můžeme derivaci funkce aproximovat derivací tohoto polynomu. Ze vztahu pro chybu interpolace

$$f(x) = L_n(x) + R_n(x)$$

přímo plyne derivováním:

$$f^{(k)}(x) = L_n^{(k)}(x) + R_n^{(k)}(x). \tag{8.1}$$

Úloha se tedy rozpadá na dva kroky: najít derivaci interpolačního polynomu $L_n^{(k)}(x)$ a odhadnout chybu, tedy derivaci chybového členu $R_n^{(k)}(x)$.

8.1.1 Derivace interpolačního polynomu

Lemma 8.1. Pro derivaci bazického Lagrangeova polynomu $l_j(x)$ platí:

$$l_j'(x) = \sum_{i=0, i\neq j}^{n} \frac{1}{x_j - x_i} \prod_{k=0, k\neq j}^{n} \frac{x - x_k}{x_j - x_k}. \tag{8.2}$$

Důkaz. TODO □

Poznámka 8.2. S využitím předchozího lemmatu můžeme první derivaci Lagrangeova polynomu zapsat jako $L_n'(x) = \sum_{j=0}^{n} f(x_j) l_j'(x)$. Odvození obecných vzorců pro vyšší derivace je však tímto způsobem technicky velmi pracné.

8.1.2 Chyba numerické derivace

Pro odvození chyby derivace budeme potřebovat Leibnizovo pravidlo pro derivaci součinu.

Lemma 8.3 (Leibnizovo pravidlo)**.** Pro funkce f, g, které jsou n-krát diferencovatelné, platí:

$$(f \cdot g)^{(n)} = \sum_{k=0}^{n} \binom{n}{k} f^{(k)} g^{(n-k)}. \tag{8.3}$$

Důkaz. TODO □

Věta 8.4 (O chybě numerické derivace)**.** Nechť funkce f má na intervalu I_x obsahujícím body $x, x_0, \ldots, x_n$ spojitou derivaci řádu $k+n+1$. Pak pro chybu k-té derivace platí:

$$f^{(k)}(x) - L_n^{(k)}(x) = R_n^{(k)}(x) = \frac{1}{(n+1)!} \sum_{i=0}^{k} \binom{k}{i} \left(f^{(n+1)}(\xi(x))\right)^{(k-i)} \omega_n^{(i)}(x). \tag{8.4}$$

Důkaz. Tvrzení plyne aplikací Leibnizova pravidla na vztah pro chybu interpolace $R_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!}\omega_n(x)$, kde označíme

$$F(x) = f^{(n+1)}(\xi(x)), \quad G(x) = \omega_n(x). \tag{8.5}$$

Poté

$$R_n^{(k)}(x) = \frac{1}{(n+1)!}(F \cdot G)^{(k)}(x) = \frac{1}{(n+1)!} \sum_{i=0}^{k} \binom{k}{i} F^{(k-i)}(x) G^{(i)}(x), \tag{8.6}$$

což odpovídá tvrzení věty. □

Poznámka 8.5. Z předchozí věty je zřejmé, že chyba derivace interpolačního polynomu závisí na derivacích členu $\omega_n(x)$. Následující lemma ukáže, že obecně $\omega_n'(x_i) \neq 0$ pro uzlové body x_i. Z toho plyne, že ani v uzlových bodech, kde je chyba samotné interpolace nulová ($R_n(x_i) = 0$), není chyba derivace nulová. Důvodem je, že derivace v bodě závisí na chování funkce v nekonečně malém okolí tohoto bodu, zatímco interpolační polynom je konstruován z bodů s konečnými rozestupy.

Lemma 8.6. Platí:

$$\omega_n^{(k)}(x) = \sum_{i_1=0}^{n} \sum_{\substack{i_2=0 \\ i_2 \neq i_1}}^{n} \cdots \sum_{\substack{i_k=0 \\ i_k \neq i_1, \ldots, i_k \neq i_{k-1}}}^{n} \prod_{\substack{j=0 \\ j \neq i_1, \ldots, j \neq i_k}}^{n} (x - x_j). \tag{8.7}$$

Důkaz. TODO □

Příklad 8.7. Pro obecné n a k se výraz špatně představuje. Podívejme se na

$$\omega_1(x) = (x - x_0)(x - x_1) = x^2 - (x_0 + x_1)x + x_0 x_1. \tag{8.8}$$

Potom

$$\omega_1'(x) = 2x - (x_0 + x_1) = (x - x_1) + (x - x_0) \tag{8.9}$$

Z toho je vidět, že opravdu nelze navolit x tak, aby pro $\omega_1'(x)$ platilo $\omega_1'(x) = 0$. (Uzly jsou různé, tj. $x_0 \neq x_1$.)

Když tedy derivace v bodě závisí na chování funkce v nekonečně malém okolí tohoto bodu, intuitivně by se nabízelo pokusit se napočítat ji přesně tím, že budeme zmenšovat rozestupy mezi uzly. Chceme-li tedy aproximovat derivaci v bodě x_0, zvolme si symetrickou sadu uzlů kolem tohoto bodu s konstantním rozestupem (krokem) $h > 0$. Body tedy budou mít tvar:

$$x_i = x_0 + ih \quad \text{pro } i = -m_1, \ldots, m_2,$$

kde m_1, m_2 jsou nezáporná celá čísla. Celkový počet bodů je $n+1$, kde $n = m_1 + m_2$. Pro tyto body pak můžeme zkonstruovat interpolační polynom $L_n(x)$. Pro zjednodušení analýzy chování pro $h \to 0$ zavedeme substituci $x = x_0 + th$. Derivace v bodě x_0 pak odpovídá derivaci v bodě $t = 0$. Touto transformací se polynom $L_n(x)$

převede na polynom $L_n(t)$, který je funkcí proměnné t a má tvar:

$$L_n(t) = \sum_{i=-m_1}^{m_2} f(x_i) l_i(t), \tag{8.10}$$

kde

$$l_i(t) = l_i(x(t)) = \prod_{\substack{j=-m_1 \\ j\neq i}}^{m_2} \frac{x(t)-x_j}{x_i-x_j} = \prod_{\substack{j=-m_1 \\ j\neq i}}^{m_2} \frac{x_0+th-x_0-jh}{x_0+ih-x_0-jh} = \prod_{\substack{j=-m_1 \\ j\neq i}}^{m_2} \frac{t-j}{i-j}. \tag{8.11}$$

Dále se polynom $\omega_n(x) = \prod_{i=-m_1}^{m_2}(x-x_i)$ převede na tvar, kde je zjevná jeho závislost na kroku h:

$$\omega_n(t) = \omega_n(x(t)) = \prod_{i=-m_1}^{m_2} (x_0+th-(x_0+ih)) = \prod_{i=-m_1}^{m_2} h(t-i) = h^{n+1} \prod_{i=-m_1}^{m_2} (t-i). \tag{8.12}$$

Dosazením do odvozeného vztahu pro k-tou derivaci ω_n:

$$\omega_n^{(k)}(t) = h^{n+1-k} \sum_{i_1=-m_1}^{m_2} \sum_{\substack{i_2=-m_1 \\ i_2\neq i_1}}^{m_2} \cdots \sum_{\substack{i_k=-m_1 \\ i_k\neq i_1,\ldots,i_k\neq i_{k-1}}}^{m_2} \prod_{\substack{j=-m_1 \\ j\neq i_1,\ldots,j\neq i_k}}^{m_2} (t-j). \tag{8.13}$$

Z tohoto skutečně plyne, že pro $h \to 0$ získáme $\omega_n^{(k)}(t) \to 0$. Tento poznatek je klíčový pro transformaci původní věty o chybě numerické derivace na nové znění.

Věta 8.8 (Řád chyby numerické derivace)**.** Nechť funkce f má na intervalu I_x obsahujícím body $x, x_{-m_1}, \ldots, x_{m_2}$ zavedené výše spojitou derivaci řádu $k+n+1$. Nechť $L_n(x)$ je její interpolační polynom sestrojený pro tyto uzly. Pak existuje $\xi(t) \in I_x$

$$f^{(k)}(t) - L_n^{(k)}(t) = R_n^{(k)}(t) = \frac{\sum_{i=0}^{k} \left(f^{(n+1)}(\xi(t))\right)^{(k-i)}}{(n+1)!} \omega_n^{(i)}(t). \tag{8.14}$$

Protože $\omega_n^{(i)}(t)$ má tvar odvozený výše.

Závěr z této věty je zásadní: chyba, které se dopouštíme při aproximaci k-té derivace pomocí polynomu sestrojeného z $n+1$ ekvidistantních bodů, klesá s $(n+1-k)$-tou mocninou kroku h. To znamená, že použitím více bodů (zvýšením n a tedy $h \to 0$) můžeme zkonstruovat vzorce pro numerickou derivaci libovolně vysokého řádu přesnosti.

Definice 8.9 (Landauova notace)**.** Symbolem $O(h^r)$ budeme značit třídu všech funkcí $g : \mathbb{R} \to \mathbb{R}$ takových, že pro jejich limitu v nule platí:

$$\lim_{h\to 0} \frac{|g(h)|}{|h|^r} = C, \tag{8.15}$$

kde $C \in \mathbb{R}$ je nenulová konstanta. Zápis $f(h) = O(h^r)$ tedy znamená, že pro malé h se funkce $f(h)$ chová úměrně h^r. Místo $g \in O(h^r)$ se často používá i značení $g = O(h^r)$.

Poznámka 8.10. V předchozí části jsme tedy ukázali, že chyba numerické derivace k-té řádu přesnosti je $O(h^{n+1-k})$.

8.1.3 Konečné diference

Vzorce pro numerickou derivaci, odvozené z interpolačního polynomu na ekvidistantních uzlech, se nazývají **konečné diference**. Tyto vzorce se liší podle:

- řádu derivace k, kterou aproximují,
- řádu přesnosti (chyby) aproximace r,
- konfigurace uzlů, které využívají (dopředné, zpětné, centrální).

Ukážeme si odvození několika základních typů konečných diferencí.

8.1.3.1 Dopředná diference 1. řádu přesnosti

Věta 8.11. Nechť $f \in C^3([x_0, x_1])$ a $h = x_1 - x_0$. Pak pro tzv. dopřednou konečnou diferenci platí:

$$f'(x_0) = \frac{f(x_1) - f(x_0)}{h} + O(h). \tag{8.16}$$

Aproximace $\frac{f(x_1)-f(x_0)}{h}$ má tedy řád přesnosti 1.

Důkaz. Vyjdeme z interpolačního polynomu prvního stupně v Newtonově tvaru pro uzly x_0, x_1:

$$\begin{aligned} f(x) &= L_1(x) + R_1(x) \\ f(x) &= f(x_0) + f[x_0, x_1](x - x_0) + \frac{f''(\xi(x))}{2!}\omega_n(x) \\ f(x) &= f(x_0) + \frac{f(x_1) - f(x_0)}{x_1 - x_0}(x - x_0) + \frac{f''(\xi(x))}{2!}(x - x_0)(x - x_1). \end{aligned} \tag{8.17}$$

Derivováním celého výrazu podle x dostaneme:

$$f'(x) = \frac{f(x_1) - f(x_0)}{h} + \frac{d}{dx}\left(\frac{f''(\xi(x))}{2}\right)\omega_1(x) + \frac{f''(\xi(x))}{2}\omega_1'(x).$$

Dosadíme-li $x = x_0$, člen s $\omega_1(x_0)$ se vynuluje. Pro derivaci $\omega_1'(x) = (x - x_1) + (x - x_0)$ v bodě x_0 platí $\omega_1'(x_0) = x_0 - x_1 = -h$. Získáme tedy:

$$f'(x_0) = \frac{f(x_1) - f(x_0)}{h} + \frac{f''(\xi(x_0))}{2}(-h).$$

Přerovnáním dostaneme výraz pro chybu:

$$f'(x_0) - \frac{f(x_1) - f(x_0)}{h} = -\frac{f''(\xi)}{2}h.$$

Absolutní hodnota chyby je tedy řádu $O(h)$. □

8.1.3.2 Zpětná diference 1. řádu přesnosti

Věta 8.12. Nechť $f \in C^3([x_{-1}, x_0])$ a $h = x_0 - x_{-1}$. Pak pro tzv. zpětnou konečnou diferenci platí:

$$f'(x_0) = \frac{f(x_0) - f(x_{-1})}{h} + O(h). \tag{8.18}$$

Důkaz. Důkaz je zcela analogický k dopředné diferenci, pouze se použijí uzly x_{-1}, x_0. □

8.1.3.3 Centrální diference 2. řádu přesnosti

Věta 8.13. Nechť $f \in C^4([x_{-1}, x_1])$ a uzly x_{-1}, x_0, x_1 jsou ekvidistantní s krokem h. Pak pro tzv. centrální konečnou diferenci platí:

$$f'(x_0) = \frac{f(x_1) - f(x_{-1})}{2h} + O(h^2). \tag{8.19}$$

Aproximace má tedy řád přesnosti 2.

Důkaz. Vyjdeme z interpolačního polynomu druhého stupně pro uzly x_{-1}, x_0, x_1:

$$f(x) = L_2(x) + \frac{f^{(3)}(\xi(x))}{3!}(x - x_{-1})(x - x_0)(x - x_1).$$

Zderivujeme-li tento výraz a dosadíme $x = x_0$, klíčové je chování derivace členu $\omega_2(x) = (x-x_{-1})(x-x_0)(x-x_1)$ v bodě x_0:

$$\omega_2'(x_0) = (x_0 - x_0)(x_0 - x_1) + (x_0 - x_{-1})(x_0 - x_1) + (x_0 - x_{-1})(x_0 - x_0) = (h)(-h) = -h^2.$$

Po dosazení a úpravách, které zahrnují vyjádření poměrných diferencí, získáme:

$$f'(x_0) = \frac{f(x_1) - f(x_{-1})}{2h} - \frac{f^{(3)}(\xi(x_0))}{6}h^2.$$

Chyba aproximace je tedy řádu $O(h^2)$. K vyššímu řádu přesnosti dochází díky symetrickému rozložení uzlů kolem bodu x_0, což způsobí vyrušení chybového členu řádu $O(h)$. □

8.2 Výpočet derivace pomocí Taylorova polynomu

8.2.1 Konečné diference

Vzorce pro konečné diference lze alternativně odvodit pomocí Taylorova rozvoje. Tento přístup je často jednodušší a elegantnější než derivování interpolačního polynomu a vystačí si se slabšími předpoklady na hladkost funkce f.

8.2.1.1 Dopředná diference

Věta 8.14. Nechť $f \in C^2([x_0, x_1])$ a $h = x_1 - x_0$. Pak pro tzv. dopřednou konečnou diferenci platí:

$$f'(x_0) = \frac{f(x_1) - f(x_0)}{h} + O(h). \tag{8.20}$$

Důkaz. Sestrojíme Taylorův rozvoj funkce f v bodě x_0 se zbytkem v Lagrangeově tvaru:

$$f(x) = f(x_0) + f'(x_0)(x - x_0) + \frac{f''(\xi)}{2!}(x - x_0)^2,$$

kde ξ leží mezi x a x_0. Dosadíme-li do tohoto rozvoje bod $x = x_1$, dostaneme:

$$f(x_1) = f(x_0) + f'(x_0)(x_1 - x_0) + \frac{f''(\xi)}{2}(x_1 - x_0)^2 \quad \text{pro } \xi \in [x_0, x_1].$$

S využitím $h = x_1 - x_0$ můžeme rovnici přepsat a vyjádřit z ní derivaci $f'(x_0)$:

$$f(x_1) = f(x_0) + hf'(x_0) + \frac{h^2}{2}f''(\xi)$$
$$\frac{f(x_1) - f(x_0)}{h} = f'(x_0) + \frac{h}{2}f''(\xi).$$

Chyba aproximace je tedy:

$$\left|\frac{f(x_1) - f(x_0)}{h} - f'(x_0)\right| = \left|\frac{h}{2}f''(\xi)\right| = O(h).$$

□

8.2.1.2 Zpětná diference

Věta 8.15. Nechť $f \in C^2([x_{-1}, x_0])$ a $h = x_0 - x_{-1}$. Pak pro tzv. zpětnou konečnou diferenci platí:

$$f'(x_0) = \frac{f(x_0) - f(x_{-1})}{h} + O(h). \tag{8.21}$$

Důkaz. Důkaz je analogický. Do Taylorova rozvoje v bodě x_0 dosadíme $x = x_{-1}$:

$$f(x_{-1}) = f(x_0) + f'(x_0)(x_{-1} - x_0) + \frac{f''(\xi)}{2}(x_{-1} - x_0)^2 = f(x_0) - hf'(x_0) + \frac{h^2}{2}f''(\xi).$$

Úpravou opět získáme chybu řádu $O(h)$. □

8.2.1.3 Centrální diference pro první derivaci

Věta 8.16. Nechť $f \in C^3([x_{-1}, x_1])$ a uzly x_{-1}, x_0, x_1 jsou ekvidistantní s krokem h. Pak pro tzv. centrální konečnou diferenci platí:

$$f'(x_0) = \frac{f(x_1) - f(x_{-1})}{2h} + O(h^2). \tag{8.22}$$

Důkaz. Sestavíme dva Taylorovy rozvoje v bodě x_0, tentokrát do třetího řádu:

$$f(x_1) = f(x_0) + hf'(x_0) + \frac{h^2}{2!}f''(x_0) + \frac{h^3}{3!}f^{(3)}(\xi_1)$$
$$f(x_{-1}) = f(x_0) - hf'(x_0) + \frac{h^2}{2!}f''(x_0) - \frac{h^3}{3!}f^{(3)}(\xi_2),$$

kde $\xi_1 \in [x_0, x_1]$ a $\xi_2 \in [x_{-1}, x_0]$. Odečtením druhé rovnice od první dostaneme:

$$f(x_1) - f(x_{-1}) = 2hf'(x_0) + \frac{h^3}{6}(f^{(3)}(\xi_1) + f^{(3)}(\xi_2)).$$

Po přerovnání získáme výraz pro chybu:

$$\left|\frac{f(x_1) - f(x_{-1})}{2h} - f'(x_0)\right| = \left|\frac{h^2}{12}(f^{(3)}(\xi_1) + f^{(3)}(\xi_2))\right| = O(h^2).$$

Díky symetrické volbě bodů se členy s druhou derivací vyrušily, což vedlo k vyššímu řádu přesnosti. □

8.2.1.4 Aproximace druhé derivace

Věta 8.17. Nechť $f \in C^4([x_{-1}, x_1])$ a uzly x_{-1}, x_0, x_1 jsou ekvidistantní s krokem h. Pak pro tzv. centrální konečnou diferenci druhé derivace platí:

$$f''(x_0) = \frac{f(x_1) - 2f(x_0) + f(x_{-1})}{h^2} + O(h^2). \tag{8.23}$$

Důkaz. Vyjdeme z Taylorových rozvojů do čtvrtého řádu:

$$f(x_1) = f(x_0) + hf'(x_0) + \frac{h^2}{2}f''(x_0) + \frac{h^3}{6}f^{(3)}(x_0) + \frac{h^4}{24}f^{(4)}(\xi_1)$$
$$f(x_{-1}) = f(x_0) - hf'(x_0) + \frac{h^2}{2}f''(x_0) - \frac{h^3}{6}f^{(3)}(x_0) + \frac{h^4}{24}f^{(4)}(\xi_2).$$

Tentokrát obě rovnice sečteme. Členy s první a třetí derivací se vyruší:

$$f(x_1) + f(x_{-1}) = 2f(x_0) + h^2f''(x_0) + \frac{h^4}{24}(f^{(4)}(\xi_1) + f^{(4)}(\xi_2)).$$

Vyjádřením $f''(x_0)$ a úpravou získáme chybu řádu $O(h^2)$. □

Poznámka 8.18. Řád přesnosti aproximace druhé derivace závisí na hladkosti funkce. Pokud by funkce f byla pouze třídy C^3, aproximace by měla řád přesnosti pouze $O(h)$.

8.2.1.5 Obecná konstrukce konečných diferencí

Doposud jsme odvodili několik základních vzorců. Následující věta a její konstruktivní důkaz ukazují, jak lze systematicky odvodit formuli libovolného řádu přesnosti pro libovolnou derivaci.

Věta 8.19. Nechť jsou dány ekvidistantní uzly $x_i = x_0 + ih$ pro $i = -m_1, \ldots, m_2$ (celkem $n+1$ bodů, kde $n = m_1 + m_2$). Pokud je funkce $f \in C^{n+1}(\langle x_{-m_1}, x_{m_2}\rangle)$, pak existuje konečná diference pro aproximaci k-té derivace v bodě x_0 s řádem přesnosti $n+1-k$ pro $k = 1, \ldots, n$.

Konstruktivní důkaz. Princip konstrukce spočívá v nalezení takové lineární kombinace funkčních hodnot $f(x_i)$, která po dosazení Taylorových rozvojů vyeliminuje všechny derivace kromě té požadované. Rozepíšeme si celkem $n = m_1 + m_2$ Taylorových rozvojů pro funkci f v bodech $x_i = x_0 + ih$ pro $i \in \{-m_1, \ldots, m_2\}, i \neq 0$. Středem všech rozvojů je bod x_0. Pro přehlednost budeme značit $f_j = f(x_j)$ a $f_0^{(k)} = f^{(k)}(x_0)$.

$$
\begin{aligned}
f_{-m_1} &= f_0 + (-m_1 h) f_0^{(1)} + \cdots + \frac{(-m_1 h)^n}{n!} f_0^{(n)} + \frac{(-m_1 h)^{n+1}}{(n+1)!} f^{(n+1)}(\xi_{-m_1}) \\
f_{-m_1+1} &= f_0 + ((-m_1+1)h) f_0^{(1)} + \cdots + \frac{((_1+1)h)^n}{n!} f_0^{(n)} + \ldots \\
&\vdots \\
f_{m_2} &= f_0 + (m_2 h) f_0^{(1)} + \cdots + \frac{(m_2 h)^n}{n!} f_0^{(n)} + \frac{(m_2 h)^{n+1}}{(n+1)!} f^{(n+1)}(\xi_{m_2})
\end{aligned}
$$

Hledáme takové koeficienty α_i, aby jejich lineární kombinace $\sum_{i\neq 0} \alpha_i f_i$ dala aproximaci k-té derivace. Konkrétně požadujeme, aby po sečtení vynásobených rozvojů byly koeficienty u derivací $f_0^{(j)}$ nulové pro $j \neq k$ a koeficient u $f_0^{(k)}$ byl roven jedné. Tento požadavek vede přímo na soustavu n lineárních rovnic pro n neznámých koeficientů α_i kde $i \in \{-m_1, \ldots, -1, 1, \ldots, m_2\}$.

$$
\begin{pmatrix}
(-m_1) & (-m_1)^2/2! & \ldots & (-m_1)^n/n! \\
(-m_1+1) & (-m_1+1)^2/2! & \ldots & (-m_1+1)^n/n! \\
\vdots & \vdots & \ddots & \vdots \\
(m_2) & (m_2)^2/2! & \ldots & (m_2)^n/n!
\end{pmatrix}
\begin{pmatrix}
\alpha_{-m_1} \\ \alpha_{-m_1+1} \\ \vdots \\ \alpha_{m_2}
\end{pmatrix}
=
\begin{pmatrix}
0 \\ \vdots \\ 1/h^k \\ \vdots \\ 0
\end{pmatrix}
\leftarrow k\text{-tá pozice}
$$

Po vyřešení této soustavy (jejíž matice je regulární) získáme koeficienty α_i a výsledný aproximační vzorec je:

$$
f^{(k)}(x_0) \approx \sum_{i=-m_1, i\neq 0}^{m_2} \alpha_i f(x_i)
$$

Chyba této aproximace je dána součtem zbytkových členů z Taylorových rozvojů:

$$
\text{Chyba} = \sum_{i\neq 0} \alpha_i \frac{(ih)^{n+1}}{(n+1)!} f^{(n+1)}(\xi_i)
$$

Z tvaru soustavy je vidět, že řešení α_i je úměrné $1/h^k$. Celková chyba je tedy řádu:

$$
\text{Chyba} \sim O(h^{-k}) \cdot O(h^{n+1}) = O(h^{n+1-k}).
$$

To odpovídá tvrzení věty. □

Poznámka 8.20. Všiměme si, že dostáváme stejnou přesnost $n+1-k$ jako při odvození pomocí interpolačního polynomu.

8.3 Shrnutí

Následující tabulka shrnuje odvozené vzorce a porovnává minimální požadavky na hladkost funkce f pro zaručení daného řádu chyby při odvození pomocí Lagrangeova polynomu a Taylorova rozvoje.

Diference	**Vzorec**	**k**	**r**	**Předpoklady ($f \in C^m$)** Lagrange / Taylor
Dopředná	$\frac{f(x_1)-f(x_0)}{h}$	1	1	C^3 / C^2
Zpětná	$\frac{f(x_0)-f(x_{-1})}{h}$	1	1	C^3 / C^2
Centrální	$\frac{f(x_1)-f(x_{-1})}{2h}$	1	2	C^4 / C^3
Centrální	$\frac{f(x_1)-2f(x_0)+f(x_{-1})}{h^2}$	2	2	C^6 / C^4

Viděli jsme dva hlavní přístupy k odvození vzorců pro numerickou derivaci:

- **Pomocí interpolačního polynomu:** Tento přístup je obecnější, ale technicky pracnější a pro důkaz řádu chyby vyžaduje vyšší hladkost funkce.
- **Pomocí Taylorova rozvoje:** Tento přístup je pro ekvidistantní uzly jednodušší, názorně ukazuje vztah mezi řádem přesnosti a hladkostí funkce a vystačí si se slabšími předpoklady.

Odvozené vzorce, zejména centrální diference, jsou základním stavebním kamenem pro metodu konečných diferencí, která se hojně využívá při řešení diferenciálních rovnic.

8.4 Otázky

- Jaké jsou dva hlavní způsoby odvození vzorců pro konečné diference a jaké jsou jejich výhody a nevýhody?
- Jaký je vztah mezi počtem použitých uzlů, řádem derivace a řádem přesnosti výsledného vzorce?
- Proč je centrální diference pro první derivaci přesnější než dopředná nebo zpětná?
- Jak byste obecně zkonstruovali formuli pro třetí derivaci pomocí čtyř bodů?

Kapitola 9

Numerická integrace

TODO

Bibliografie

[1] doc. Lubomíra Dvořáková. *Lineární algebra 1*. ČVUT, 2014. ISBN: 9788001053461.

[2] doc. Lubomíra Dvořáková. *Lineární algebra 2*. ČVUT, 2020. ISBN: 9788001067215.

[3] doc. Tomáš Oberhuber. *Numerická matematika 1*. přednáška na FJFI ČVUT v Praze. Dostupné z: `https://mmg.fjfi.cvut.cz/~oberhuber/node/num`. 2024.