

UNIVERZITA PALACKÉHO V OLOMOUCI
PŘÍRODOVĚDECKÁ FAKULTA
KATEDRA MATEMATICKÉ ANALÝZY A APLIKACÍ MATEMATIKY

DIPLOMOVÁ PRÁCE

Vlastní čísla matic



Vedoucí diplomové práce:
RNDr. Jitka Machalová, Ph.D.
Rok odevzdání: 2011

Vypracovala:
Bc. Michaela Lepišová
AME, II. ročník

Prohlášení

Prohlašuji, že jsem vytvořila tuto diplomovou práci samostatně za vedení RNDr. Jitky Machalové, Ph.D. a že jsem v seznamu použité literatury uvedla všechny zdroje užití při zpracování práce.

V Olomouci dne 31. března 2011

Poděkování

Ráda bych na tomto místě poděkovala vedoucí diplomové práce RNDr. Jitce Machalové, Ph.D. za obětavou spolupráci i za čas, který mi věnovala při konzultacích. V neposlední řadě je mou milou povinností poděkovat svým rodičům a blízkým za morální a materiální podporu, které se mi dostávalo po celou dobu studia.

Obsah

Označení	5
Úvod	6
1 Základní pojmy	7
1.1 Matice a vektory, invariantní podprostory	7
1.2 Normy vektorů a matic	9
1.3 Vlastní čísla a vlastní vektory	11
2 Vlastní čísla nesymetrické matice	21
2.1 Citlivost výpočtu	21
2.1.1 Citlivost výpočtu vlastního čísla	21
2.1.2 Citlivost výpočtu invariantního podprostoru	27
2.1.3 Citlivost výpočtu vlastního vektoru	30
2.2 QR rozklad	31
2.2.1 Householderovy matice	32
2.2.2 Výpočet QR rozkladu pomocí Householderovy transformace	35
2.2.3 Givensovy matice rotace	38
2.2.4 Výpočet QR rozkladu pomocí Givensových matic rotace .	41
2.3 Mocninná metoda	43
2.4 Metoda QR transformace	47
2.5 Hessenbergovy a reálné Schurovy tvary matice	51
2.5.1 Reálný Schurův rozklad	51
2.5.2 Hessenbergova redukce	52
2.5.3 Hessenbergův QR rozklad	54
2.5.4 QR transformace z matice v horním Hessenbergově tvaru .	56
3 Vlastní čísla symetrické matice	61
3.1 Vlastnosti a rozklady	61
3.1.1 Citlivost výpočtu vlastního čísla	62
3.1.2 Citlivost výpočtu invariantního podprostoru	66
3.2 Jacobiho metoda	70
3.2.1 Teoretická podstata Jacobiho metody	70
3.2.2 Symetrický Schurův rozklad druhého řádu	73
3.2.3 Klasická Jacobiho metoda	75
Závěr	78

Přílohy	79
Procedura house.m	79
Procedura QRHouse.m	80
Procedura givens.m	81
Procedura QRGivens.m	82
Procedura MocMet.m	83
Procedura QRrozklad.m	84
Procedura HessHous.m	86
Procedura QRHess.m	88
Procedura RQHess.m	89
Procedura QRHrozklad.m	90
Procedura jacobi.m	93
Procedura JacMet.m	94
Literatura	96

Označení

\mathbb{N}	množina přirozených čísel
\mathbb{R}	množina reálných čísel
\mathbb{C}	množina komplexních čísel
\mathbb{C}^n	n rozměrný vektorový prostor nad \mathbb{C}
$\mathbb{C}^{m,n}$	lineární prostor řádu $m \times n$
$\mathbf{x} = (x_1, x_2, \dots, x_n)^T, x_i \in \mathbb{C}$	sloupcový vektor
\mathbf{x}^T	řádkový vektor
$\mathbf{x}^H = (\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n)$	řádkový vektor Hermitovsky sdružený k vektoru \mathbf{x}
$\mathbf{o} = (0, 0, \dots, 0)^T$	nulový vektor
$\mathbf{A} = (a_{ij})_{i,j=1}^n, a_{ij} \in \mathbb{C}$	čtvercová matice n -tého řádu
$\mathbf{A}^H = (\bar{a}_{ji})_{j,i=1}^n$	Hermitovsky sdružená matice k matici \mathbf{A}
$ a , a \in \mathbb{C}$	absolutní hodnota čísla a
$ \mathbf{A} = (a_{ij})_{i,j=1}^n$	prvky matice \mathbf{A} jsou v absolutní hodnotě
\mathbf{A}^{-1}	inverzní matice k matici \mathbf{A}
$\mathbf{0}$	nulová matice n -tého řádu
\mathbf{I}	jednotková matice příslušného řádu
\mathbf{I}_n	jednotková matice n -tého řádu
$\mathbf{A}(i, :)$	i -tý řádek matice \mathbf{A}
$\mathbf{A}(:, i)$	i -tý sloupec matice \mathbf{A}
$\mathbf{x}(i)$	i -tá složka vektoru \mathbf{x}
$\mathbf{x}(i : n)$	i -tá až n -tá složka vektoru $\mathbf{x} \in \mathbb{R}^n$
$\lambda_i, i = 1, \dots, n, \lambda_i \in \mathbb{C}^n$	vlastní čísla matice
$\rho(\mathbf{A})$	spektrální poloměr matice \mathbf{A}
$\kappa_p(\mathbf{A})$	číslo podmíněnosti matice \mathbf{A}
$\det \mathbf{A}$	determinant matice \mathbf{A}
$\text{rank}(\mathbf{A})$	hodnost matice \mathbf{A}
$\text{ran}(\mathbf{A})$	obor hodnot matice \mathbf{A}
δ_{ij}	Kroneckerovo delta

Úvod

Tato diplomová práce navazuje na bakalářskou práci, ve které byly zpracovány základní metody výpočtu vlastních čísel. Cílem práce je nastudovat některé další metody pro výpočet vlastních čísel a prohloubit si teoretické znalosti již studovaných metod.

Diplomová práce je pro přehlednost rozčleněna do tří kapitol s následujícím obsahem. V první kapitole jsou uvedeny základní pojmy, potřebné při studiu dalšího textu. Druhá kapitola se zabývá problematikou výpočtu vlastních čísel nesymetrické matice. Úvod kapitoly je věnován problému citlivosti výpočtu vlastního čísla. V další části jsou uvedeny jednotlivé metody výpočtu vlastních čísel. Třetí kapitola se zabývá metodami výpočtu vlastních čísel symetrické matice. V úvodu kapitoly je opět popsán problém citlivosti výpočtu vlastního čísla. Další podkapitola je věnována konkrétním metodám výpočtu.

Metody jsou nejdříve vysvětleny teoreticky, poté následuje stručné shrnutí v algoritmu. Pro přehlednost jsou jednotkové matice v textu značeny jako obecné jednotkové matice \mathbf{I} , v algoritmech je již ale uvažována jednotková matice konkrétního řádu \mathbf{I}_n . Popsaná teorie je ilustrována na příkladech. Tyto jsou počítány povětšinou pomocí matematického softwaru MATLAB verze 7.5.0.342 (R2007b). Výpočty jsou prováděny buď užitím příkazu $eig(\mathbf{A})$, sloužícího k výpočtu vlastních čísel a vlastních vektorů matice \mathbf{A} , který nabízí program MATLAB, nebo mnou vytvořenými m-fily, jejichž přesnost výpočtu se blíží k přesnosti výpočtu pomocí výše zmíněného příkazu $eig(\mathbf{A})$. Některé jednodušší příklady jsou počítány ručně.

Při tvorbě práce bylo nutné prohloubit si znalosti z numerické matematiky o vlastních číslech a vlastních vektorech, o programu MATLAB a v neposlední řadě i o programu LaTeX, kterým je práce vysázena. Diplomová práce byla zpracována na notebooku ASUS F5RL s parametry: procesor Intel Core 2 Duo T5750 s frekvencí 2 GHz, čipset ATI Radeon Xpress 1100, operační paměť 3 GB DDR2 667 MHz, HDD 320 GB, 5400 RPM.

1 Základní pojmy

Cílem této kapitoly je ve stručnosti shrnout nejdůležitější pojmy, které budeme dále užívat. Pokud nebude řečeno jinak, budeme v celé práci maticí \mathbf{A} rozumět komplexní čtvercovou matici n -tého řádu, tj. $\mathbf{A} = (a_{ij})$, $i, j = 1, \dots, n$, $a_{ij} \in \mathbb{C}$.

1.1 Matice a vektory, invariantní podprostory

V této části si připomeneme základní pojmy a vztahy z lineární algebry, týkající se matic a vektorů.

Definice 1.1 Matice $\mathbf{A} = (a_{ij})$, $i, j = 1, \dots, n$ s komplexními prvky se nazývá **hermitovská**, jestliže platí $\mathbf{A}^H = \mathbf{A}$, tj. $a_{ij} = \bar{a}_{ji}$, $i, j = 1, \dots, n$.

Definice 1.2 Hermitovská matice $\mathbf{A} = (a_{ij})$, $i, j = 1, \dots, n$ se nazývá **pozitivně definitní**, jestliže pro každý nenulový komplexní vektor $\mathbf{x} = (x_1, x_2, \dots, x_n)^T$, platí

$$\mathbf{x}^H \mathbf{A} \mathbf{x} = \sum_{i=1}^n \sum_{j=1}^n a_{ij} \bar{x}_i x_j > 0.$$

Definice 1.3 Řekneme, že matice \mathbf{A} je **normální**, jestliže platí $\mathbf{A}^H \mathbf{A} = \mathbf{A} \mathbf{A}^H$.

Definice 1.4 Matice \mathbf{A} se nazývá **unitární**, jestliže platí $\mathbf{A}^H \mathbf{A} = \mathbf{A} \mathbf{A}^H = \mathbf{I}$. Reálná matice \mathbf{A} , pro kterou platí $\mathbf{A}^H = \mathbf{A}^{-1}$, tj. $\mathbf{A}^T = \mathbf{A}^{-1}$, se nazývá **ortogonální**.

Definice 1.5 Čtvercová matice $\mathbf{A} = (a_{ij})$, pro niž platí $a_{ij} = 0$, pro $i \neq j$, a_{ij} je libovolné komplexní číslo pro $i = j$, kde $i, j = 1, \dots, n$, se nazývá **diagonální matice**. Budeme ji stručně zapisovat $\mathbf{A} = \text{diag}(a_{11}, a_{22}, \dots, a_{nn})$, kde a_{ii} , $i = 1, \dots, n$ jsou prvky na hlavní diagonále.

Věta 1.1 (Singulární rozklad) Je-li $\mathbf{A} \in \mathbb{C}^{m,n}$, pak existují unitární matice $\mathbf{U} \in \mathbb{C}^{m,m}$ a $\mathbf{V} \in \mathbb{C}^{n,n}$ takové, že platí

$$\mathbf{U}^H \mathbf{A} \mathbf{V} = \begin{pmatrix} \mathbf{D} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix},$$

kde $\mathbf{D} = \text{diag}(\sigma_1, \dots, \sigma_p)$, pro $p = \min\{m, n\}$ a $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_p \geq 0$.

Důkaz: viz [5] str. 70.

Poznámka 1.1 Číslům σ_i , kde $i = 1, \dots, p$, zmíněným ve větě 1.1, říkáme **singulární čísla** matice \mathbf{A} a sloupcům matic \mathbf{U} a \mathbf{V} **levé** a **pravé singulární vektory**.

Definice 1.6 Řekneme, že čtvercová matice \mathbf{A} je **regulární**, je-li $\det \mathbf{A} \neq 0$. Je-li $\det \mathbf{A} = 0$, pak matici \mathbf{A} nazveme **singulární**.

Definice 1.7 Matice $\mathbf{A} = (a_{ij})$, $i, j = 1, \dots, n$ se nazývá **horní trojúhelníková**, resp. **ostře horní trojúhelníková** matice, jestliže $a_{ij} = 0$ pro $i > j$, resp. $i \geq j$. Matice se nazývá **dolní trojúhelníková**, resp. **ostře dolní trojúhelníková**, jestliže $a_{ij} = 0$ pro $i < j$, resp. $i \leq j$.

Definice 1.8 Řekneme, že matice $\mathbf{A} = (a_{ij})$ je v **horním Hessenbergově tvaru**, jestliže platí $a_{ij} = 0$ pro $i > j + 1$, kde $i, j = 1, \dots, n$.

Definice 1.9 Řekneme, že vektory $\mathbf{x}, \mathbf{y} \in \mathbb{C}^n$ jsou **ortogonální**, jestliže platí $\mathbf{x}^H \mathbf{y} = 0$. Vektory $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n$ se nazývají vzájemně **ortonormální**, jestliže

$$\mathbf{x}_i^H \mathbf{x}_j = \delta_{ij} = \begin{cases} 1 & \text{pro } i = j \\ 0 & \text{pro } i \neq j. \end{cases}$$

Definice 1.10 Podprostor $S \subseteq \mathbb{C}^n$ se nazývá **invariantní** (pro \mathbf{A}), jestliže platí

$$\mathbf{x} \in S \Rightarrow \mathbf{A}\mathbf{x} \in S.$$

Poznámka 1.2 Podprostor prostoru \mathbb{C}^n definovaný vztahem

$$\text{ran}(A) = \{\mathbf{y} \mid \exists \mathbf{x} : \mathbf{y} = \mathbf{A}\mathbf{x}\}$$

nazýváme **obor hodnot matice \mathbf{A}** .

1.2 Normy vektorů a matic

Definice 1.11 Vektorová norma na \mathbb{C}^n je funkce $\|\cdot\| : \mathbb{C}^n \rightarrow \mathbb{R}$ s následujícími vlastnostmi

1. $\|\mathbf{x}\| \geq 0$, $\mathbf{x} \in \mathbb{C}^n$
2. $\|\mathbf{x}\| = 0 \Leftrightarrow \mathbf{x} = \mathbf{o}$, $\mathbf{o} = (0, \dots, 0)^T$
3. $\|\alpha\mathbf{x}\| = |\alpha| \|\mathbf{x}\|$, $\alpha \in \mathbb{C}$, $\mathbf{x} \in \mathbb{C}^n$
4. $\|\mathbf{x} + \mathbf{y}\| \leq \|\mathbf{x}\| + \|\mathbf{y}\|$, $\mathbf{x}, \mathbf{y} \in \mathbb{C}^n$.

Existuje řada možností, jak normu vektoru definovat. Mezi nejznámější vektorové normy patří

1. $\|\mathbf{x}\|_2 = \left(\sum_{i=1}^n |x_i|^2 \right)^{\frac{1}{2}}$
2. $\|\mathbf{x}\|_1 = \sum_{i=1}^n |x_i|$
3. $\|\mathbf{x}\|_\infty = \max_{1 \leq i \leq n} |x_i|$

Poznámka 1.3 Vektorové normy $\|\cdot\|_1$ resp. $\|\cdot\|_2$ jsou speciální případy vektorové normy

$$\|\mathbf{x}\|_p = \left(\sum_{i=1}^n |x_i|^p \right)^{\frac{1}{p}}, \quad p \geq 1,$$

pro $p = 1$ resp. $p = 2$.

Definice 1.12 Maticová norma na $\mathbb{C}^{m,n}$ je funkce $\|\cdot\| : \mathbb{C}^{m,n} \rightarrow \mathbb{R}$ s těmito vlastnostmi

1. $\|\mathbf{A}\| \geq 0$, $\mathbf{A} \in \mathbb{C}^{m,n}$
2. $\|\mathbf{A}\| = 0 \Leftrightarrow \mathbf{A}$ je nulová matice
3. $\|\alpha\mathbf{A}\| = |\alpha| \|\mathbf{A}\|$, $\alpha \in \mathbb{C}$, $\mathbf{A} \in \mathbb{C}^{m,n}$

$$4. \|\mathbf{A} + \mathbf{B}\| \leq \|\mathbf{A}\| + \|\mathbf{B}\|, \quad \mathbf{A}, \mathbf{B} \in \mathbb{C}^{m,n}$$

$$5. \|\mathbf{AB}\| \leq \|\mathbf{A}\| \|\mathbf{B}\|, \quad \mathbf{A} \in \mathbb{C}^{m,n}, \mathbf{B} \in \mathbb{C}^{n,m}$$

Někdy je vhodné požadovat, aby norma matice nějakým způsobem „souvisela“ s normou vektoru. Tuto vlastnost nazýváme **souhlasnost** a její definice je následující.

Definice 1.13 Řekneme, že maticová norma $\|\cdot\|$ je **souhlasná** s danou vektorovou normou $\|\cdot\|_\varphi$, jestliže

$$\|\mathbf{Ax}\|_\varphi \leq \|\mathbf{A}\| \|\mathbf{x}\|_\varphi, \quad \mathbf{x} \in \mathbb{C}^n, \quad \mathbf{A} \in \mathbb{C}^{m,n}.$$

Věta 1.2 *Nechť $\|\cdot\|_\varphi$ je vektorová norma na \mathbb{C}^n . Pak číslo*

$$\|\mathbf{A}\|_\varphi = \max_{\|\mathbf{x}\|_\varphi=1} \|\mathbf{Ax}\|_\varphi$$

*je maticová norma souhlasná s danou vektorovou normou $\|\cdot\|_\varphi$. Tato norma se nazývá **přidružená** k dané vektorové normě.*

Důkaz: viz [7] str. 16.

Věta 1.3 *Nechť $\mathbf{A} \in \mathbb{C}^{m,n}$. Přidružené maticové normy k vektorovým normám $\|\cdot\|_1, \|\cdot\|_\infty, \|\cdot\|_2$ jsou dány vztahy*

$$1. \|\mathbf{A}\|_1 = \max_{1 \leq j \leq n} \sum_{i=1}^m |a_{ij}|,$$

$$2. \|\mathbf{A}\|_\infty = \max_{1 \leq i \leq m} \sum_{j=1}^n |a_{ij}|,$$

$$3. \|\mathbf{A}\|_2 = \sqrt{\varrho(\mathbf{A}^H \mathbf{A})}, \quad \text{kde } \varrho(\mathbf{A}^H \mathbf{A}) \text{ je spektrální poloměr } \mathbf{A}^H \mathbf{A}.$$

Důkaz: viz [6] str. 215.

Poznámka 1.4 Norma $\|\mathbf{A}\|_2$ se nazývá **spektrální norma** matice \mathbf{A} . Důležitou normou souhlasnou s vektorovou normou $\|\cdot\|_2$ je Frobeniova norma

$$\|\mathbf{A}\|_F = \left(\sum_{i=1}^m \sum_{j=1}^n |a_{ij}|^2 \right)^{\frac{1}{2}}.$$

1.3 Vlastní čísla a vlastní vektory

Abychom se mohli zabývat metodami výpočtu vlastních čísel a vlastních vektorů, potřebujeme ještě zadefinovat základní pojmy týkající se teorie vlastních čísel a vlastních vektorů.

Definice 1.14 Nechť \mathbf{A} je čtvercová matice. Nenulový vektor \mathbf{x} se nazývá **pravý vlastní (charakteristický) vektor** matice \mathbf{A} , jestliže platí

$$\mathbf{A}\mathbf{x} = \lambda\mathbf{x}$$

pro nějaké číslo $\lambda \in \mathbb{C}$. Toto číslo λ se nazývá **vlastní (charakteristické) číslo** matice \mathbf{A} , odpovídající vlastnímu vektoru \mathbf{x} .

Poznámka 1.5

1. **Levý vlastní vektor** \mathbf{y} , odpovídající vlastnímu číslu λ , je řešením rovnice $\mathbf{y}^H \mathbf{A} = \lambda \mathbf{y}^H$. Levý vlastní vektor matice \mathbf{A} je tedy pravým vlastním vektorem matice \mathbf{A}^H .
2. Vlastní vektor není určen jednoznačně, protože platí

$$\mathbf{A}\mathbf{x} = \lambda\mathbf{x} \Leftrightarrow \mathbf{A}(\alpha\mathbf{x}) = \lambda(\alpha\mathbf{x})$$

pro každé $\alpha \in \mathbb{C}$.

Poznámka 1.6 V dalším textu budeme o pravém vlastním vektoru hovořit jako o vlastním vektoru.

Definice 1.15 Nechť \mathbf{A} je komplexní čtvercová matice. Polynom

$$p_{\mathbf{A}}(\lambda) = \det(\mathbf{A} - \lambda\mathbf{I})$$

nazveme **charakteristický polynom matice \mathbf{A}** .

Věta 1.4 *Vlastním číslem čtvercové matice \mathbf{A} řádu n je právě takové číslo $\lambda \in \mathbb{C}$, které je kořenem charakteristického polynomu*

$$p_{\mathbf{A}}(\lambda) = \det(\mathbf{A} - \lambda\mathbf{I}) = (-1)^n \lambda^n + b_1 \lambda^{n-1} + \dots + b_{n-1} \lambda + b_n,$$

kde koeficient b_k je součet všech hlavních minorů k -téko řádu matice \mathbf{A} , pro $k = 1, \dots, n$.

Důkaz: viz [4] str. 36.

Poznámka 1.7 Každá čtvercová matice n -tého řádu má právě n vlastních čísel $\lambda_1, \lambda_2, \dots, \lambda_n$, přičemž každé vlastní číslo počítáme tolikrát, kolik je jeho násobnost.

Definice 1.16 Nechť je dána matice \mathbf{A} řádu n a nechť $\lambda_1, \lambda_2, \dots, \lambda_n$ jsou její vlastní čísla. Pak matici $\bar{\mathbf{A}} = \text{diag}(\lambda_1, \dots, \lambda_n)$ nazýváme **spektrální maticí matice \mathbf{A}** . Množina všech vlastních čísel matice \mathbf{A} se nazývá **spektrum matice \mathbf{A}** . Budeme jej označovat $\lambda(\mathbf{A})$.

Věta 1.5 *Je-li \mathbf{x}_i vlastní vektor matice \mathbf{A} odpovídající vlastnímu číslu λ_i , potom z definice 1.14 plynou rovnosti $\mathbf{A}\mathbf{x}_i = \lambda_i\mathbf{x}_i$, $i = 1, 2, \dots, n$, které můžeme zapsat ve tvaru maticové rovnosti*

$$\mathbf{A}\mathbf{X} = \mathbf{X}\bar{\mathbf{A}},$$

kde sloupce matice $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n)$ jsou vlastní vektory matice \mathbf{A} .

Důkaz: Rozepíšeme-li vztah $\mathbf{A}\mathbf{X} = \mathbf{X}\bar{\mathbf{A}}$ dostáváme rovnost

$$\mathbf{A}(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n) = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n) \begin{pmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \lambda_n \end{pmatrix},$$

odkud je ihned vidět platnost tvrzení. ■

Lemma 1.1 *Nechť je dána matice $\mathbf{A} \in \mathbb{C}^{n,n}$. Dále nechť λ_i resp. \mathbf{x}_i , pro $i = 1, \dots, n$, jsou vlastní čísla, resp. vlastní vektory matice \mathbf{A} . Potom platí*

$$\mathbf{A}^k \mathbf{x}_i = \lambda_i^k \mathbf{x}_i,$$

pro $i = 1, \dots, n, k = 1, 2, \dots$

Důkaz: Podle definice 1.14 platí $\mathbf{A}\mathbf{x}_i = \lambda_i\mathbf{x}_i$, $i = 1, \dots, n$. Tedy lze psát

$$\mathbf{A}^k \mathbf{x}_i = \mathbf{A}^{k-1}(\mathbf{A}\mathbf{x}_i) = \mathbf{A}^{k-1}(\lambda_i\mathbf{x}_i) = \lambda_i \mathbf{A}^{k-2}(\mathbf{A}\mathbf{x}_i) = \lambda_i^2 \mathbf{A}^{k-3}(\mathbf{A}\mathbf{x}_i) = \dots = \lambda_i^k \mathbf{x}_i,$$

což je vztah, který jsme měli dokázat. ■

Definice 1.17 Vlastní vektor definuje jednorozměrný podprostor S , který je invariantní vzhledem k pronásobení s maticí \mathbf{A} , tj. $\mathbf{x} \in S \Rightarrow \mathbf{A}\mathbf{x} \in S$.

Poznámka 1.8 Jestliže

$$\mathbf{A}\mathbf{X} = \mathbf{X}\mathbf{B},$$

kde $\mathbf{A} \in \mathbb{C}^{n,n}$, $\mathbf{B} \in \mathbb{C}^{k,k}$ a $\mathbf{X} \in \mathbb{C}^{n,k}$. Potom $\text{ran}(\mathbf{X})$ je invariantní pro matici \mathbf{A} a $\mathbf{B}\mathbf{y} = \lambda\mathbf{y} \Rightarrow \mathbf{A}(\mathbf{X}\mathbf{y}) = \lambda(\mathbf{X}\mathbf{y})$. Tedy, jestliže \mathbf{X} má plnou sloupcovou hodnotu, potom $\mathbf{A}\mathbf{X} = \mathbf{B}\mathbf{X}$ znamená, že $\lambda(\mathbf{B}) \subseteq \lambda(\mathbf{A})$. Jestliže je matice \mathbf{X} čtvercová a regulární, pak $\lambda(\mathbf{A}) = \lambda(\mathbf{B})$ a říkáme, že matice \mathbf{A} a $\mathbf{B} = \mathbf{X}^{-1}\mathbf{A}\mathbf{X}$ jsou podobné.

Definice 1.18 Řekneme, že matice \mathbf{A} a \mathbf{B} jsou **podobné**, jestliže existuje regulární matice \mathbf{P} taková, že

$$\mathbf{A} = \mathbf{P}\mathbf{B}\mathbf{P}^{-1}.$$

Věta 1.6 *Nechť jsou dány podobné matice \mathbf{A} , \mathbf{B} a regulární matice \mathbf{P} taková, že $\mathbf{A} = \mathbf{P}\mathbf{B}\mathbf{P}^{-1}$. Podobné matice mají stejný charakteristický polynom a tedy stejná vlastní čísla a to i co do násobnosti. Je-li \mathbf{x} vlastní vektor matice \mathbf{B} , potom $\mathbf{P}\mathbf{x}$ je vlastní vektor matice \mathbf{A} , odpovídající témuž vlastnímu číslu.*

Důkaz: viz [4] str. 37.

Pokud vlastní vektory $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n$ matice \mathbf{A} jsou lineárně nezávislé, bude matice \mathbf{X} v rovnosti $\mathbf{A}\mathbf{X} = \mathbf{X}\bar{\mathbf{A}}$, kde sloupce matice \mathbf{X} jsou vlastní vektory matice \mathbf{A} , regulární a plyne odtud

$$\mathbf{X}^{-1}\mathbf{A}\mathbf{X} = \bar{\mathbf{A}},$$

tzn. matice \mathbf{A} je v tomto případě podobná spektrální matici $\bar{\mathbf{A}}$. Tato situace nastává např., když matice \mathbf{A} má všechna vlastní čísla vzájemně různá, nebo když je symetrická. Ale ne každá matice je podobná matici spektrální. Dá se však dokázat, že libovolná čtvercová matice \mathbf{A} je podobná matici v tzv. **Jordanově tvaru**.

Definice 1.19 Řekneme, že matice \mathbf{J} je v **Jordanově tvaru**

$$\mathbf{J} = \begin{pmatrix} \mathbf{J}_1(\lambda_1) & 0 & \dots & 0 \\ 0 & \mathbf{J}_2(\lambda_2) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \mathbf{J}_r(\lambda_r) \end{pmatrix},$$

je-li blokově diagonální a její každý diagonální blok je Jordanův blok, kde čísla λ_i , $i = 1, \dots, r$ jsou vzájemně různá vlastní čísla matice \mathbf{A} . **Jordanovým blokem** $\mathbf{J}_i(\lambda_i)$, $i = 1, \dots, r$ nazveme čtvercovou matici, jejíž prvky na hlavní diagonále jsou rovny vlastnímu číslu λ_i , na diagonále nad hlavní diagonálou jsou samé jedničky a ostatní prvky má rovny nule, tj.

$$\mathbf{J}_i(\lambda_i) = \begin{pmatrix} \lambda_i & 1 & 0 & \dots & 0 \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & \ddots & 1 \\ 0 & \dots & \dots & 0 & \lambda_i \end{pmatrix}.$$

Přitom rozměr matice $\mathbf{J}_i(\lambda_i)$ je roven násobnosti vlastního čísla λ_i .

Definice 1.20 Determinant

$$\det(\mathbf{J}_i(\lambda_i) - \lambda \mathbf{I}) = \begin{vmatrix} (\lambda_i - \lambda) & 1 & \dots & 0 \\ 0 & \ddots & \ddots & 0 \\ 0 & 0 & \ddots & 1 \\ 0 & 0 & \dots & (\lambda_i - \lambda) \end{vmatrix} = (\lambda_i - \lambda)^{v_i},$$

kde v_i je řád matice $\mathbf{J}_i(\lambda_i)$, se nazývá **elementární dělitel** matice \mathbf{A} . Je-li $v_i = 1$, řekneme, že příslušný elementární dělitel je lineární.

Mnoho výpočtů vlastních čísel zahrnuje rozklad daného problému do souboru menších vlastních problémů. Následující lemma je základem pro tyto redukce.

Lemma 1.2 *Nechť je dána matice $\mathbf{T} \in \mathbb{C}^{n,n}$ ve tvaru*

$$\mathbf{T} = \begin{pmatrix} \mathbf{T}_{11} & \mathbf{T}_{12} \\ \mathbf{0} & \mathbf{T}_{22} \end{pmatrix},$$

kde $\mathbf{T}_{11} \in \mathbb{C}^{p,p}$, $\mathbf{T}_{12} \in \mathbb{C}^{p,n-p}$, $\mathbf{T}_{22} \in \mathbb{C}^{n-p,n-p}$. Pak platí $\lambda(\mathbf{T}) = \lambda(\mathbf{T}_{11}) \cup \lambda(\mathbf{T}_{22})$.

Důkaz: viz [5] str. 311.

Pomocí podobnostních transformací je možné redukovat dané matice na jeden z možných kanonických tvarů. Kanonické tvary se liší v zobrazení vlastních čísel vzhledem k typu invariantního podprostoru.

Jak se lze dočíst v [5], [12], je výhodné diagonalizovat danou matici pomocí ortogonální transformace, nebo unitární transformace.

Lemma 1.3 *Nechť jsou dány matice $\mathbf{A} \in \mathbb{C}^{n,n}$, $\mathbf{B} \in \mathbb{C}^{p,p}$ a $\mathbf{X} \in \mathbb{C}^{n,p}$ takové, že*

$$\mathbf{A}\mathbf{X} = \mathbf{X}\mathbf{B}, \quad \text{rank}(\mathbf{X}) = p.$$

Pak existuje unitární matice $\mathbf{Q} \in \mathbb{C}^{n,n}$ taková, že

$$\mathbf{Q}^H \mathbf{A} \mathbf{Q} = \mathbf{T} = \begin{pmatrix} \mathbf{T}_{11} & \mathbf{T}_{12} \\ \mathbf{0} & \mathbf{T}_{22} \end{pmatrix} \quad (1)$$

kde $\mathbf{T}_{11} \in \mathbb{C}^{p,p}$, $\mathbf{T}_{12} \in \mathbb{C}^{p,n-p}$, $\mathbf{T}_{22} \in \mathbb{C}^{n-p,n-p}$ a $\lambda(\mathbf{T}_{11}) = \lambda(\mathbf{A}) \cap \lambda(\mathbf{B})$.

Důkaz: viz [5] str. 312.

Lemma 1.3 říká, že matice \mathbf{A} může být redukována do blokového trojúhelníkového tvaru užitím unitární podobnostní transformace, jestliže známe jeden z jejích invariantních podprostorů. Užitím matematické indukce lze matici \mathbf{T} z lemmatu 1.3 zformulovat obecněji. Tedy konkrétně $\mathbf{T} = \bar{\mathbf{A}} + \mathbf{N}$, kde $\bar{\mathbf{A}}$ je spektrální matice a matice \mathbf{N} je ostře horní trojúhelníková. Tomuto rozkladu říkáme **Schurův rozklad**. Na základě těchto poznatků můžeme vyslovit následující větu.

Věta 1.7 (Schurův rozklad)

Nechť $\mathbf{A} \in \mathbb{C}^{n,n}$ a $\lambda_1, \dots, \lambda_n$ jsou její vlastní čísla. Pak existuje unitární matice $\mathbf{Q} \in \mathbb{C}^{n,n}$ taková, že

$$\mathbf{Q}^H \mathbf{A} \mathbf{Q} = \mathbf{T}, \quad \text{kde } \mathbf{T} = \bar{\mathbf{A}} + \mathbf{N}, \quad (2)$$

přičemž $\bar{\mathbf{A}} = \text{diag}(\lambda_1, \dots, \lambda_n)$ a $\mathbf{N} \in \mathbb{C}^{n,n}$ je ostře horní trojúhelníková. Matice \mathbf{Q} může být zvolena tak, že vlastní čísla λ_i , $i = 1, \dots, n$ jsou na diagonále matice $\bar{\mathbf{A}}$ umístěna libovolně.

Důkaz: viz [5] str. 313.

Poznámka 1.9 Sloupce matice \mathbf{Q} nazýváme **Schurovy vektory**.

Příklad 1.1 Najděme Schurův rozklad matice \mathbf{A} , jestliže

$$\mathbf{A} = \begin{pmatrix} 2 & 5 \\ -1 & 2 \end{pmatrix} \quad a \quad \mathbf{Q} = \begin{pmatrix} 0.9129i & 0.4082 \\ -0.4082 & -0.9129i \end{pmatrix}.$$

Jednoduše lze ověřit, že matice \mathbf{Q} je unitární a

$$\mathbf{Q}^H \mathbf{A} \mathbf{Q} = \begin{pmatrix} 2 + 2.2361i & -4 \\ 0 & 2 - 2.2361i \end{pmatrix},$$

kde

$$\bar{\mathbf{A}} = \text{diag}(2 + 2.2361i, 2 - 2.2361i) \quad a \quad \mathbf{N} = \begin{pmatrix} 0 & -4 \\ 0 & 0 \end{pmatrix}.$$

Nechť $\mathbf{Q} = (\mathbf{q}_1, \dots, \mathbf{q}_n)$ je unitární matice ze vztahu (2), přičemž \mathbf{q}_i jsou Schurovy vektory. Porovnáním sloupců matic v rovnosti $\mathbf{A}\mathbf{Q} = \mathbf{Q}\mathbf{T}$ (viz vztah (2)) dostaneme pro Schurovy vektory následující vztah

$$\mathbf{A}\mathbf{q}_k = \lambda_k \mathbf{q}_k + \sum_{i=1}^{k-1} n_{ik} \mathbf{q}_i \quad k = 1, \dots, n, \quad (3)$$

kde $n_{ik} \in \mathbb{C}$. Odtud plyne (viz [5] str. 314), že podprostory

$$\mathbf{S}_k = \text{span}\{\mathbf{q}_1, \dots, \mathbf{q}_k\} \quad k = 1, \dots, n \quad (4)$$

jsou invariantní. Označíme-li, pro $k = 1, \dots, n$, matice $\mathbf{Q}_k = (\mathbf{q}_1, \dots, \mathbf{q}_k)$, pak $\lambda(\mathbf{Q}_k^H \mathbf{A} \mathbf{Q}_k) = \{\lambda_1, \dots, \lambda_k\}$. Protože je možné vlastní čísla ve vztahu (2) libovolně seřadit, lze dokázat, že existuje alespoň jeden k -rozměrný podprostor spojený s každou podmnožinou k vlastních čísel. Dalším zjištěním vycházejícím ze vztahu (3) je, že Schurův vektor \mathbf{q}_k je vlastní vektor matice \mathbf{A} tehdy a jen tehdy, když k -tý sloupec matice \mathbf{N} je nulový, pro $k = 1, \dots, n$. To platí vždy, když je matice \mathbf{A} normální.

Důsledek 1.1 *Matice $\mathbf{A} \in \mathbb{C}^{n,n}$ je normální tehdy a jen tehdy, když existuje unitární matice $\mathbf{Q} \in \mathbb{C}^{n,n}$ taková, že platí $\mathbf{Q}^H \mathbf{A} \mathbf{Q} = \text{diag}(\lambda_1, \dots, \lambda_n)$.*

Důkaz: viz [5] str. 314.

Jestliže $\mathbf{Q}^H \mathbf{A} \mathbf{Q} = \mathbf{T}$, kde $\mathbf{T} = \text{diag}(\lambda_1, \dots, \lambda_n) + \mathbf{N}$ je Schurův rozklad matice $\mathbf{A} \in \mathbb{C}^{n,n}$, potom $\|\mathbf{N}\|_F$ je nezávislá na volbě \mathbf{Q}

$$\|\mathbf{N}\|_F^2 = \|\mathbf{A}\|_F^2 - \sum_{i=1}^n |\lambda_i|^2 \equiv \Delta^2(\mathbf{A}).$$

Tato veličina je označována jako **odklon od normality matice \mathbf{A}** . Jestliže chceme vytvořit matici \mathbf{T} „více diagonální“, využijeme transformaci neunitární podobnosti. Následující lemma se zabývá redukcí neunitární podobnosti.

Lemma 1.4 *Nechť $\mathbf{T} \in \mathbb{C}^{n,n}$ je dána ve tvaru*

$$\mathbf{T} = \begin{pmatrix} \mathbf{T}_{11} & \mathbf{T}_{12} \\ \mathbf{0} & \mathbf{T}_{22} \end{pmatrix},$$

kde $\mathbf{T}_{11} \in \mathbb{C}^{p,p}$, $\mathbf{T}_{12} \in \mathbb{C}^{p,n-p}$, $\mathbf{T}_{22} \in \mathbb{C}^{n-p,n-p}$. Definujme lineární transformaci $\phi : \mathbb{C}^{p,n-p} \rightarrow \mathbb{C}^{p,n-p}$ vztahem

$$\phi(\mathbf{X}) = \mathbf{T}_{11}\mathbf{X} - \mathbf{X}\mathbf{T}_{22},$$

kde $\mathbf{X} \in \mathbb{C}^{p,n-p}$. Potom ϕ je regulární tehdy a jen tehdy, když $\lambda(\mathbf{T}_{11}) \cap \lambda(\mathbf{T}_{22}) = \emptyset$. Jestliže ϕ je regulární a \mathbf{Y} je definovaná vztahem

$$\mathbf{Y} = \begin{pmatrix} \mathbf{I}_p & \mathbf{Z} \\ \mathbf{0} & \mathbf{I}_{n-p} \end{pmatrix}, \quad \text{přičemž } \phi(\mathbf{Z}) = -\mathbf{T}_{12}$$

potom $\mathbf{Y}^{-1}\mathbf{T}\mathbf{Y} = \text{diag}(\mathbf{T}_{11}, \mathbf{T}_{22})$.

Důkaz: viz [5] str. 314.

Příklad 1.2 *Nechť je dána matice*

$$\mathbf{T} = \begin{pmatrix} 2 & 3 & 1 \\ 0 & 1 & 9 \\ 0 & -3 & 1 \end{pmatrix},$$

kde

$$\mathbf{T}_{11} = (2), \quad \mathbf{T}_{12} = (3 \ 1), \quad a \quad \mathbf{T}_{22} = \begin{pmatrix} 1 & 9 \\ -3 & 1 \end{pmatrix}.$$

Ručním výpočtem vlastních čísel matic \mathbf{T}_{11} a \mathbf{T}_{22} zjistíme, že $\lambda(\mathbf{T}_{11}) = \{2\}$ a $\lambda(\mathbf{T}_{22}) = \{1 + 5.1962i, 1 - 5.1962i\}$. Z toho vidíme, že platí $\lambda(\mathbf{T}_{11}) \cap \lambda(\mathbf{T}_{22}) = \emptyset$. Můžeme tedy říct, že lineární transformace ϕ z lemmatu 1.4 je regulární a dosazením do vztahu $\phi(\mathbf{Z}) = \mathbf{T}_{11}\mathbf{Z} - \mathbf{Z}\mathbf{T}_{22} = -\mathbf{T}_{12}$ dostaneme

$$\mathbf{Y} = \begin{pmatrix} 1 & 0 & -1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad a \quad \mathbf{Y}^{-1}\mathbf{T}\mathbf{Y} = \begin{pmatrix} 2 & 0 & 0 \\ 0 & 1 & 9 \\ 0 & -3 & 1 \end{pmatrix},$$

tj. pomocí redukce neunitární podobnosti jsme získali matici, která má na diagonále bloky \mathbf{T}_{11} a \mathbf{T}_{22} .

Opakovaným užitím lemmatu 1.4 lze zformulovat následující obecnější výsledek.

Věta 1.8 (Blokově diagonální rozklad) *Předpokládejme, že*

$$\mathbf{Q}^H \mathbf{A} \mathbf{Q} = \mathbf{T} = \begin{pmatrix} \mathbf{T}_{11} & \mathbf{T}_{12} & \dots & \mathbf{T}_{1q} \\ \mathbf{0} & \mathbf{T}_{22} & \dots & \mathbf{T}_{2q} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{T}_{qq} \end{pmatrix} \quad (5)$$

je Schurův rozklad matice $\mathbf{A} \in \mathbb{C}^{n,n}$ a předpokládejme, že \mathbf{T}_{ii} , $i = 1, \dots, q$ jsou čtvercové. Jestliže $\lambda(\mathbf{T}_{ii}) \cap \lambda(\mathbf{T}_{jj}) = \emptyset$, kde $i \neq j$, pro $i, j = 1, \dots, q$, potom existuje regulární matice $\mathbf{Y} \in \mathbb{C}^{n,n}$ taková, že

$$(\mathbf{Q}\mathbf{Y})^{-1} \mathbf{A} (\mathbf{Q}\mathbf{Y}) = \text{diag}(\mathbf{T}_{11}, \dots, \mathbf{T}_{qq}). \quad (6)$$

Důkaz: viz [5] str. 316.

Důsledek 1.2 *Jestliže $\mathbf{A} \in \mathbb{C}^{n,n}$, pak existuje regulární matice \mathbf{X} taková, že*

$$\mathbf{X}^{-1} \mathbf{A} \mathbf{X} = \text{diag}(\lambda_1 \mathbf{I} + \mathbf{N}_1, \dots, \lambda_q \mathbf{I} + \mathbf{N}_q) \quad \mathbf{N}_i \in \mathbb{C}^{n_i, n_i}, \quad (7)$$

kde $\lambda_1, \dots, \lambda_q$ jsou vzájemně různá vlastní čísla matice \mathbf{A} , n_1, \dots, n_q celá kladná čísla, pro která platí $n_1 + \dots + n_q = n$ a každá \mathbf{N}_i je ostře horní trojúhelníková.

S rozkladem ve tvaru (7) je spojeno mnoho důležitých pojmů, viz následující poznámka.

Poznámka 1.10 Celé číslo n_i , z důsledku 1.2 se nazývá **algebraická násobnost** vlastního čísla λ_i . Jestliže $n_i = 1$, potom říkáme, že λ_i je **jednoduché**. **Geometrická násobnost** vlastního čísla λ_i je rovna počtu lineárně nezávislých

vlastních vektorů, odpovídajících vlastnímu číslu λ_i . Jestliže algebraická násobnost λ_i převyšuje jeho geometrickou násobnost, pak říkáme, že λ_i je **defektní** vlastní číslo. Matice s defektním vlastním číslem se nazývá **defektní matice**. **Nedefektním** maticím se také říká **diagonalizovatelné**.

Důsledek 1.3 (Diagonální tvar) *Matice $\mathbf{A} \in \mathbb{C}^{n,n}$ je nedefektní tehdy a jen tehdy, když existuje regulární matice $\mathbf{X} \in \mathbb{C}^{n,n}$ taková, že*

$$\mathbf{X}^{-1}\mathbf{A}\mathbf{X} = \text{diag}(\lambda_1, \dots, \lambda_q). \quad (8)$$

Důkaz: viz [5] str. 316.

Příklad 1.3 Nechť jsou dány matice

$$\mathbf{A} = \begin{pmatrix} 1 & 3 & 0 \\ 3 & -2 & -1 \\ 0 & -1 & 1 \end{pmatrix} \quad a \quad \mathbf{X} = \begin{pmatrix} -3 & 1 & 3 \\ 5 & 0 & 2 \\ 1 & 3 & -1 \end{pmatrix}.$$

Potom $\mathbf{X}^{-1}\mathbf{A}\mathbf{X} = \text{diag}(-4, 1, 3)$. Výpočtem kořenů charakteristického polynomu matice \mathbf{A} lze ověřit, že vlastní čísla matice jsou skutečně $\lambda_1 = -4$, $\lambda_2 = 1$, $\lambda_3 = 3$. Z důsledku 1.3 tedy plyne, že matice \mathbf{A} je nedefektní.

2 Vlastní čísla nesymetrické matice

Cílem této kapitoly je představit problém vlastních čísel nesymetrické matice. V první části se zaměříme na to, jak jsou vlastní čísla citlivá na změny v matici. V druhé části si představíme metodu QR rozkladu, která patří mezi univerzální metody pro výpočet vlastních čísel matice.

2.1 Citlivost výpočtu

Výpočet vlastních čísel matice $\mathbf{A} \in \mathbb{C}^{n,n}$ spočívá ve výpočtu kořenů charakteristického polynomu. Jestliže $n > 4$, doporučuje se při hledání těchto kořenů užití iteračních metod, neboť při přímém výpočtu dochází k nárůstu chyb. Abychom mohli stanovit vhodné zastavovací kritérium pro iterační proces, je třeba prozkoumat citlivost výpočtu vlastního čísla. Výpočet vlastních čísel může být citlivý na změny v matici \mathbf{A} . Říkáme, že výpočet vlastních čísel je dobře podmíněn, jestliže malá změna v matici \mathbf{A} vyvolá malou změnu hodnot vlastních čísel. Číslo $\kappa_p(\mathbf{A}) = \|\mathbf{A}\|_p \|\mathbf{A}^{-1}\|_p$ se nazývá **číslem podmíněnosti** matice \mathbf{A} . Lze snadno ukázat, že $\kappa_p(\mathbf{A}) \geq 1$. Je-li $\kappa_p(\mathbf{A})$ „malé“, úloha je dobře podmíněná, tím lépe podmíněná, čím je číslo podmíněnosti $\kappa_p(\mathbf{A})$ blíže 1. Je-li $\kappa_p(\mathbf{A})$ „velké“, říkáme, že úloha je špatně podmíněná.

2.1.1 Citlivost výpočtu vlastního čísla

Několik metod výpočtu vlastních čísel vytváří posloupnosti transformací $\{\mathbf{X}_k\}$ s takovou vlastností, že s rostoucím k jsou matice $\mathbf{X}_k^{-1} \mathbf{A} \mathbf{X}_k$ postupně „více diagonální“. Vzniká tedy otázka, jak dobře aproximují diagonální prvky těchto matic vlastní čísla matice \mathbf{A} .

Věta 2.1 (Gershgorinovy kruhy)

Nechť je dána matice \mathbf{A} a nechť $\lambda(\mathbf{A})$ je její spektrum. Dále nechť matice $\mathbf{X}^{-1} \mathbf{A} \mathbf{X} = \mathbf{D} + \mathbf{F}$, kde $\mathbf{D} = \text{diag}(d_1, \dots, d_n)$ a \mathbf{F} má nulové diagonální prvky,

potom

$$\lambda(\mathbf{A}) \subseteq \bigcup_{i=1}^n \mathbf{D}_i,$$

kde $\mathbf{D}_i = \{z \in \mathbb{C} : |z - d_i| \leq \sum_{j=1}^n |f_{ij}|\}$, $\mathbf{F} = (f_{ij})_{i,j=1}^n$.

Důkaz: viz [5] str. 320.

Poznámka 2.1 Je několik možností, jak zvolit matice \mathbf{X} , \mathbf{D} a \mathbf{F} z věty 2.1. V následujícím příkladě uvedeme jednu z nich.

Příklad 2.1 Užitím věty 2.1 odhadněme polohu vlastních čísel matice

$$\mathbf{A} = \begin{pmatrix} 7 & 2 & 1 \\ -3 & 1 & 2 \\ 3 & -2 & 0 \end{pmatrix}.$$

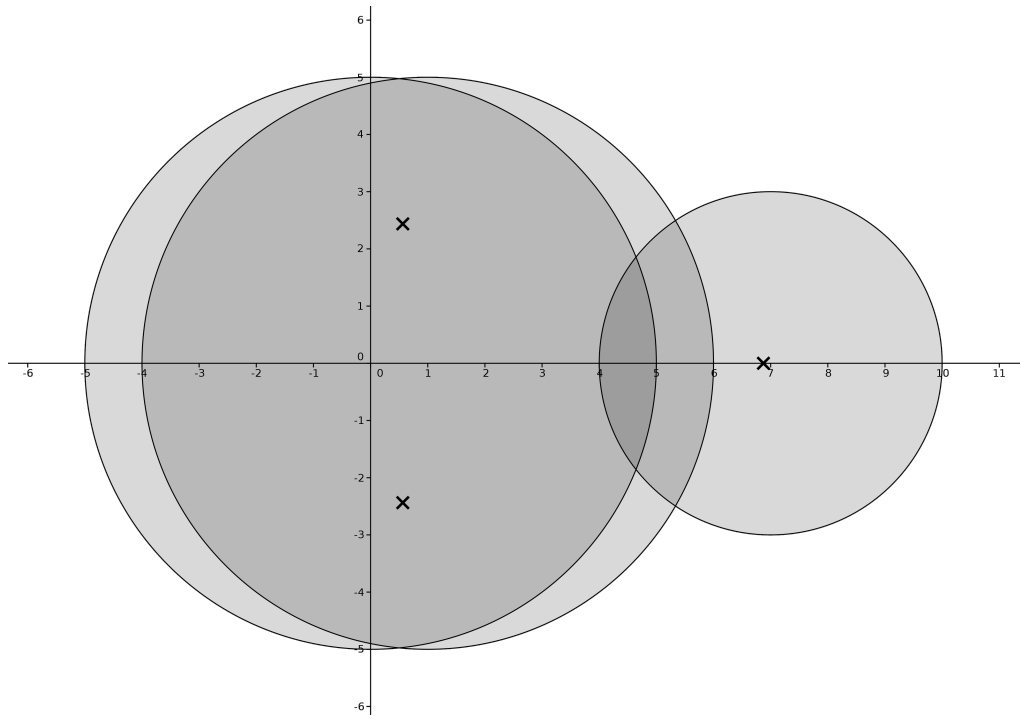
Lze zvolit např.

$$\mathbf{D} = \begin{pmatrix} 7 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad \mathbf{F} = \begin{pmatrix} 0 & 2 & 1 \\ -3 & 0 & 2 \\ 3 & -2 & 0 \end{pmatrix} \quad \text{a} \quad \mathbf{X} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Dle vztahu $\mathbf{D}_i = \{z \in \mathbb{C} : |z - d_i| \leq \sum_{j=1}^n |f_{ij}|\}$ určíme Gershgorinovy kruhy ve tvaru

$$\begin{aligned} \mathbf{D}_1 &= \{|z| : |z - 7| \leq |2| + |1| = 3\}, \\ \mathbf{D}_2 &= \{|z| : |z - 1| \leq |-3| + |2| = 5\}, \\ \mathbf{D}_3 &= \{|z| : |z| \leq |3| + |-2| = 5\}. \end{aligned}$$

Výpočtem kořenů charakteristického polynomu matice \mathbf{A} lze zjistit, že vlastní čísla jsou $\lambda_1 = 6.8733$, $\lambda_2 = 0.5633 + 2.4369i$ a $\lambda_3 = 0.5633 - 2.4369i$ (na obrázku jsou vyznačena znakem \times).



Pro některé výpočetní metody vlastních čísel je možné ukázat, že vypočtená vlastní čísla jsou přesně rovna vlastním číslům matice $\mathbf{A} + \mathbf{E}$, přičemž $\|\mathbf{E}\|$ je malá hodnota. Otázkou je, jak mohou být vlastní čísla matice ovlivněna malými změnami. Tuto otázku objasní následující věty.

Věta 2.2 (Bauer-Fike)

Je-li μ vlastní číslo matice $\mathbf{A} + \mathbf{E} \in \mathbb{C}^{n,n}$ a $\mathbf{X}^{-1}\mathbf{A}\mathbf{X} = \bar{\mathbf{A}} = \text{diag}(\lambda_1, \dots, \lambda_n)$, potom

$$\min_{\lambda \in \lambda(\mathbf{A})} |\lambda - \mu| \leq \kappa_p(\mathbf{X}) \|\mathbf{E}\|_p,$$

kde $\|\cdot\|_p$ značí některou z p -norm.

Důkaz: viz [5] str. 321.

Podobný výsledek obdržíme užitím Schurova rozkladu matice \mathbf{A} .

Věta 2.3 Necht' $\mathbf{Q}^H \mathbf{A} \mathbf{Q} = \bar{\mathbf{A}} + \mathbf{N}$ je Schurův rozklad matice $\mathbf{A} \in \mathbb{C}^{n,n}$ ze vztahu (2). Je-li $\mu \in \lambda(\mathbf{A} + \mathbf{E})$ a p je nejmenší celé kladné číslo takové, že

$|\mathbf{N}|^p = \mathbf{0}$, potom

$$\min_{\lambda \in \lambda(\mathbf{A})} |\lambda - \mu| \leq \max\{\theta, \theta^{1/p}\},$$

kde

$$\theta = \|\mathbf{E}\|_2 \sum_{k=0}^{p-1} \|\mathbf{N}\|_2^k.$$

Důkaz: viz [5] str. 322.

Příklad 2.2 Nechť jsou dány matice

$$\mathbf{A} = \begin{pmatrix} 1 & 3 & 1 \\ 0 & 3 & 2 \\ 0 & 0 & 3.001 \end{pmatrix} \quad \text{a} \quad \mathbf{E} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0.001 & 0 & 0 \end{pmatrix}.$$

Je zřejmé, že $\lambda(\mathbf{A}) = \{1, 3, 3.001\}$. Výpočtem charakteristického polynomu matice $\mathbf{A} + \mathbf{E}$ zjistíme její spektrum $\lambda(\mathbf{A} + \mathbf{E}) = \{1.001, 3.0548, 2.9452\}$. Matice vlastních vektorů matice \mathbf{A} je ve tvaru

$$\mathbf{X} = \begin{pmatrix} 1 & 0.8321 & 0.8320 \\ 0 & 0.5547 & 0.5548 \\ 0 & 0 & 0.0003 \end{pmatrix}$$

a má číslo podmíněnosti $\kappa_F(\mathbf{X}) = 0.8831 \times 10^4$. Potom podle Bauer-Fikeho věty 2.2 je $\min_{\lambda \in \lambda(\mathbf{A})} |\lambda - \mu| \leq 0.8831 \times 10^1$. Pro srovnání vypočítáme také horní

odhad $\min_{\lambda \in \lambda(\mathbf{A})} |\lambda - \mu|$ z věty 2.3. Tedy necht' $\mathbf{Q}^H \mathbf{A} \mathbf{Q} = \bar{\mathbf{A}} + \mathbf{N}$ je Schurův rozklad matice \mathbf{A} , kde

$$\bar{\mathbf{A}} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 3.001 \end{pmatrix}, \quad \mathbf{N} = \begin{pmatrix} 0 & 3 & 1 \\ 0 & 0 & 2 \\ 0 & 0 & 0 \end{pmatrix} \quad \text{a} \quad \mathbf{Q} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Potom $|\mathbf{N}|^p = \mathbf{0}$, pro $p = 3$, $\theta = \|\mathbf{E}\|_2 \sum_{k=0}^2 \|\mathbf{N}\|_2^k = 0.0149$ a $\theta^{1/3} = 0.2459$. Tedy

$$\min_{\lambda \in \lambda(\mathbf{A})} |\lambda - \mu| \leq 0.2459 \times 10^0.$$

Věty 2.2 a 2.3 ukazují možnou citlivost výpočtu vlastního čísla, není-li matice \mathbf{A} normální. Speciálně, je-li $\kappa_p(\mathbf{X})$ nebo $\|\mathbf{N}\|_2^{p-1}$ velké, potom i malé změny v matici \mathbf{A} mohou vyvolat velké změny ve vlastních číslech, což je vidět i v příkladu 2.2.

Příklad 2.3 Nechť jsou dány matice

$$\mathbf{A} = \begin{pmatrix} \mathbf{0} & \mathbf{I}_7 \\ 0 & \mathbf{0}^T \end{pmatrix} \quad \text{a} \quad \mathbf{E} = \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ 10^{-10} & \mathbf{0}^T \end{pmatrix},$$

kde $\mathbf{A}, \mathbf{E} \in \mathbb{R}^{8,8}$.

Potom pro všechna $\lambda \in \lambda(\mathbf{A})$ a $\mu \in \lambda(\mathbf{A} + \mathbf{E})$ platí $|\lambda - \mu| = 10^{-3}$. V tomto případě změna řádu 10^{-10} v matici \mathbf{A} způsobí změnu řádu 10^{-3} v jejich vlastních číslech.

Je-li matice \mathbf{A} normální, není výpočet vlastního čísla příliš citlivý na změny. Na druhé straně, nenormalita nemusí nutně znamenat citlivost vlastního čísla. Jestliže matice \mathbf{A} není normální, může být pro některá její vlastní čísla výpočet dobře podmíněn a pro jiná špatně. Proto je vhodné vylepšit teorii citlivosti tak, aby byla použitelná pro konkrétní vlastní čísla a ne pro spektrum jako celek.

Za tímto účelem předpokládejme, že λ je jednoduché vlastní číslo matice $\mathbf{A} \in \mathbb{C}^{n,n}$ a že vektory \mathbf{x} a \mathbf{y} vyhovují rovnicím $\mathbf{A}\mathbf{x} = \lambda\mathbf{x}$ a $\mathbf{y}^H\mathbf{A} = \lambda\mathbf{y}^H$ a $\|\mathbf{x}\|_2 = \|\mathbf{y}\|_2 = 1$. Je-li $\mathbf{Y}^H\mathbf{A}\mathbf{X} = \mathbf{J}$ Jordanův rozklad matice \mathbf{A} , kde $\mathbf{Y}^H = \mathbf{X}^{-1}$, pak \mathbf{y} a \mathbf{x} jsou nenulové násobky vektorů \mathbf{x}_i a \mathbf{y}_i pro některé $i = 1, \dots, n$, kde \mathbf{x}_i resp. \mathbf{y}_i jsou sloupce matice \mathbf{X} resp. \mathbf{Y} . To plyne ze vztahu $1 = \mathbf{y}_i^H\mathbf{x}_i$, a tedy platí $\mathbf{y}^H\mathbf{x} \neq 0$. Tuto skutečnost budeme využívat i dále.

Užitím klasických výsledků z teorie funkcí se dá ukázat, že existují diferencovatelné funkce $\mathbf{x}(\varepsilon)$ a $\lambda(\varepsilon)$ takové, že

$$(\mathbf{A} + \varepsilon\mathbf{F})\mathbf{x}(\varepsilon) = \lambda(\varepsilon)\mathbf{x}(\varepsilon), \quad \|\mathbf{F}\|_2 = 1, \quad (9)$$

kde $\lambda(0) = \lambda$ a $\mathbf{x}(0) = \mathbf{x}$. Derivováním této rovnice podle ε a volbou $\varepsilon = 0$ obdržíme

$$\mathbf{A}\mathbf{x}'(0) + \mathbf{F}\mathbf{x} = \lambda'(0)\mathbf{x} + \lambda\mathbf{x}'(0).$$

Tuto rovnici vynásobíme zleva vektorem \mathbf{y}^H a vydělíme $\mathbf{y}^H \mathbf{x}$, tj.

$$\frac{\mathbf{y}^H \mathbf{A} \mathbf{x}'(0)}{\mathbf{y}^H \mathbf{x}} + \frac{\mathbf{y}^H \mathbf{F} \mathbf{x}}{\mathbf{y}^H \mathbf{x}} = \frac{\lambda'(0) \mathbf{y}^H \mathbf{x}}{\mathbf{y}^H \mathbf{x}} + \frac{\lambda \mathbf{y}^H \mathbf{x}'(0)}{\mathbf{y}^H \mathbf{x}}$$

$$\frac{\lambda \mathbf{y}^H \mathbf{x}'(0)}{\mathbf{y}^H \mathbf{x}} + \frac{\mathbf{y}^H \mathbf{F} \mathbf{x}}{\mathbf{y}^H \mathbf{x}} = \lambda'(0) + \frac{\lambda \mathbf{y}^H \mathbf{x}'(0)}{\mathbf{y}^H \mathbf{x}}.$$

Odtud, ze vztahu (9) a z vlastnosti vektorů \mathbf{x} , \mathbf{y} dostaneme v absolutní hodnotě

$$|\lambda'(0)| = \left| \frac{\mathbf{y}^H \mathbf{F} \mathbf{x}}{\mathbf{y}^H \mathbf{x}} \right| \leq \frac{1}{|\mathbf{y}^H \mathbf{x}|}.$$

Rovnosti je dosaženo, jestliže $\mathbf{F} = \mathbf{y} \mathbf{x}^H$. Z tohoto důvodu říkáme hodnotě

$$s(\lambda) = |\mathbf{y}^H \mathbf{x}|$$

podmínka pro vlastní číslo λ .

Z výše uvedené analýzy plyne, že jsou-li prvky matice \mathbf{A} pozměněny o hodnotu ε , potom vlastní číslo λ matice \mathbf{A} může být pozměněné hodnotou $\varepsilon/s(\lambda)$. Tedy jestliže $s(\lambda)$ je malé, potom λ je špatně podmíněné vlastní číslo, což znamená, že i malá změna v matici \mathbf{A} má za následek velkou změnu ve vlastním čísle.

Malé $s(\lambda)$ znamená, že matice \mathbf{A} je blízká matici, která má vícenásobné vlastní číslo. Obzvláště, jestliže λ je jednoduché a $s(\lambda) < 1$, potom existuje matice \mathbf{E} taková, že λ je vícenásobné vlastní číslo matice $\mathbf{A} + \mathbf{E}$ a platí

$$\frac{\|\mathbf{E}\|_2}{\|\mathbf{A}\|_2} \leq \frac{s(\lambda)}{\sqrt{1 - s(\lambda)^2}}.$$

Příklad 2.4 Nechť jsou dány matice

$$\mathbf{A} = \begin{pmatrix} 1 & 3 & 1 \\ 0 & 3 & 2 \\ 0 & 0 & 3.001 \end{pmatrix} \quad \text{a} \quad \mathbf{E} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0.001 & 0 & 0 \end{pmatrix}.$$

Víme, že $\lambda(\mathbf{A}) = \{1, 3, 3.001\}$. Výpočtem charakteristického polynomu matice $\mathbf{A} + \mathbf{E}$ zjistíme její spektrum $\lambda(\mathbf{A} + \mathbf{E}) = \{1.001, 3.0548, 2.9452\}$. Vlastní vektory

matice \mathbf{A} zapíšeme do matice

$$\mathbf{X} = \begin{pmatrix} 1 & 0.8321 & 0.8320 \\ 0 & 0.5547 & 0.5548 \\ 0 & 0 & 0.0003 \end{pmatrix}.$$

Potom

$$\mathbf{Y}^H = \mathbf{X}^{-1} = 10^3 \begin{pmatrix} 0.001 & -0.0015 & 0.0010 \\ 0 & 0.0018 & -3.6056 \\ 0 & 0 & 3.6047 \end{pmatrix},$$

kde sloupce matice \mathbf{Y}^H odpovídají levým vlastním vektorům matice \mathbf{A} . Nyní můžeme určit hodnoty $s(\lambda) = |\mathbf{y}^H \mathbf{x}|$ pro jednotlivá vlastní čísla λ matice \mathbf{A} , kde \mathbf{x} resp. \mathbf{y} jsou sloupce matice \mathbf{X} resp. \mathbf{Y}^H . Postupně dostáváme $s(1) = 10^0$, $s(3) = 0.2 \times 10^0$ a $s(3.001) = 2 \times 10^3$.

Jestliže je λ vícenásobné vlastní číslo, je otázka citlivosti vlastních čísel složitější.

Každá množina vlastních vektorů, které odpovídají příslušným vlastním číslům, definuje invariantní podprostor. Podobně, jak tomu bylo u výpočtu vlastních čísel, může být také výpočet invariantního podprostoru, tj. výpočet vlastních vektorů, citlivý na změny v matici \mathbf{A} . Jestliže množina vlastních čísel odpovídá množině blízkých vlastních vektorů, říkáme, že tyto vektory definují **necitlivý invariantní podprostor**.

2.1.2 Citlivost výpočtu invariantního podprostoru

Množina vlastních vektorů, odpovídajících špatně podmíněným vlastním číslům, může definovat necitlivý invariantní podprostor za předpokladu, že odpovídající shluk vlastních čísel je izolovaný. Předpokládejme, že

$$\mathbf{Q}^H \mathbf{A} \mathbf{Q} = \begin{pmatrix} \mathbf{T}_{11} & \mathbf{T}_{12} \\ \mathbf{0} & \mathbf{T}_{22} \end{pmatrix} \quad (10)$$

je Schurův rozklad matice \mathbf{A} , kde $\mathbf{T}_{11} \in \mathbb{C}^{r,r}$, $\mathbf{T}_{12} \in \mathbb{C}^{r,n-r}$ a $\mathbf{T}_{22} \in \mathbb{C}^{n-r,n-r}$, přičemž

$$\mathbf{Q} = \begin{pmatrix} \mathbf{Q}_1 & \mathbf{Q}_2 \end{pmatrix}, \quad (11)$$

kde $\mathbf{Q}_1 \in \mathbb{C}^{n,r}$ a $\mathbf{Q}_2 \in \mathbb{C}^{n,n-r}$.

Z rozkladu matice vlastních vektorů je vidět (viz vztah (11)), že citlivost výpočtu invariantního podprostoru $\text{ran}(\mathbf{Q}_1)$ závisí na vzdálenosti mezi $\lambda(\mathbf{T}_{11})$ a $\lambda(\mathbf{T}_{22})$. Správné měření této vzdálenosti ukáže, že jde o nejmenší singulární hodnotu lineární transformace $\mathbf{X} \rightarrow \mathbf{T}_{11}\mathbf{X} - \mathbf{X}\mathbf{T}_{22}$ (viz lemma 1.4). Zejména, jestliže vzdálenost mezi maticemi \mathbf{T}_{11} a \mathbf{T}_{22} definujeme vztahem

$$\text{sep}(\mathbf{T}_{11}, \mathbf{T}_{22}) = \min_{\mathbf{X} \neq 0} \frac{\|\mathbf{T}_{11}\mathbf{X} - \mathbf{X}\mathbf{T}_{22}\|_F}{\|\mathbf{X}\|_F} \geq 0, \quad (12)$$

přičemž rovnosti je dosaženo, když $\|\mathbf{T}_{11}\mathbf{X} - \mathbf{X}\mathbf{T}_{22}\|_F = 0$, dostaneme následující tvrzení.

Věta 2.4 *Předpokládejme, že vztahy (10) a (11) platí a že pro každou matici $\mathbf{E} \in \mathbb{C}^{n,n}$ máme rozklad matice $\mathbf{Q}^H \mathbf{E} \mathbf{Q}$ ve tvaru*

$$\mathbf{Q}^H \mathbf{E} \mathbf{Q} = \begin{pmatrix} \mathbf{E}_{11} & \mathbf{E}_{12} \\ \mathbf{E}_{21} & \mathbf{E}_{22} \end{pmatrix},$$

kde $\mathbf{E}_{11} \in \mathbb{C}^{r,r}$, $\mathbf{E}_{12} \in \mathbb{C}^{r,n-r}$ a $\mathbf{E}_{22} \in \mathbb{C}^{n-r,n-r}$. Jestliže $\text{sep}(\mathbf{T}_{11}, \mathbf{T}_{22}) > 0$ a

$$\|\mathbf{E}\|_2 \left(1 + \frac{5\|\mathbf{T}_{12}\|_2}{\text{sep}(\mathbf{T}_{11}, \mathbf{T}_{22})} \right) \leq \frac{\text{sep}(\mathbf{T}_{11}, \mathbf{T}_{22})}{5},$$

potom existuje $\mathbf{P} \in \mathbb{C}^{n-r,r}$ taková, že

$$\|\mathbf{P}\|_2 \leq 4 \frac{\|\mathbf{E}_{21}\|_2}{\text{sep}(\mathbf{T}_{11}, \mathbf{T}_{22})},$$

a sloupce matice $\hat{\mathbf{Q}}_1 = (\mathbf{Q}_1 + \mathbf{Q}_2 \mathbf{P})(\mathbf{I} + \mathbf{P}^T \mathbf{P})^{-1/2}$ tvoří ortonormální bázi invariantního podprostoru pro matici $\mathbf{A} + \mathbf{E}$.

Důkaz: viz [5] str. 325.

Důsledek 2.1 *Jestliže platí předpoklady ve větě 2.4, potom pro vzdálenost mezi podprostory platí*

$$\text{dist}(\text{ran}(\mathbf{Q}_1), \text{ran}(\hat{\mathbf{Q}}_1)) \leq 4 \frac{\|\mathbf{E}_{21}\|_2}{\text{sep}(\mathbf{T}_{11}, \mathbf{T}_{22})}.$$

Důkaz: viz [5] str. 325.

Jak je uvedeno v literatuře [5] str. 325, z důsledku 2.1 se ukazuje, že převrácenou hodnotu $\text{sep}(\mathbf{T}_{11}, \mathbf{T}_{22})$ můžeme brát jako číslo podmíněnosti matice \mathbf{A} , které udává míru citlivosti výpočtu invariantního podprostoru $\text{ran}(\mathbf{Q}_1)$.

Příklad 2.5 Nechť jsou dány matice

$$\mathbf{T}_{11} = \begin{pmatrix} 2 & 9 \\ 0 & 1 \end{pmatrix}, \quad \mathbf{T}_{22} = \begin{pmatrix} 0 & -18 \\ 0 & 2.01 \end{pmatrix} \quad \text{a} \quad \mathbf{T}_{12} = \begin{pmatrix} -1 & 1 \\ 1 & -1 \end{pmatrix},$$

pro které platí

$$\mathbf{A} = \mathbf{T} = \begin{pmatrix} 2 & 9 & -1 & 1 \\ 0 & 1 & 1 & -1 \\ 0 & 0 & 0 & -18 \\ 0 & 0 & 0 & 2.01 \end{pmatrix}.$$

Potom matice \mathbf{Q}_1 bude mít tvar $\mathbf{Q}_1 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}$ a bude platit $\mathbf{A}\mathbf{Q}_1 = \mathbf{Q}_1\mathbf{T}_{11}$.

Ze vztahu $\text{sep}(\mathbf{T}_{11}, \mathbf{T}_{22}) = \min_{\mathbf{X} \neq 0} \frac{\|\mathbf{T}_{11}\mathbf{X} - \mathbf{X}\mathbf{T}_{22}\|_F}{\|\mathbf{X}\|_F}$ vypočítáme vzdálenost mezi maticemi \mathbf{T}_{11} a \mathbf{T}_{22} , kde matici \mathbf{X} volíme tak, aby platilo $\mathbf{T}_{11}\mathbf{X} - \mathbf{X}\mathbf{T}_{22} = -\mathbf{T}_{12}$ (viz lemma 1.4), tj. $\mathbf{X} = 10^3 \begin{pmatrix} 0.005 & -7.8307 \\ -0.001 & -0.0188 \end{pmatrix}$. Potom

$$\text{sep}(\mathbf{T}_{11}, \mathbf{T}_{22}) = 2.554 \times 10^{-4}.$$

Jestliže $\mathbf{E}_{21} = 10^{-5} \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$ a zkoumáme Schurův rozklad matice

$$\mathbf{A} + \mathbf{E} = \begin{pmatrix} \mathbf{T}_{11} & \mathbf{T}_{12} \\ \mathbf{E}_{21} & \mathbf{T}_{22} \end{pmatrix},$$

zjistíme, že pro matici \mathbf{Q}_1 dostaneme dle vztahu (10) matici

$$\hat{\mathbf{Q}}_1 = \begin{pmatrix} -1 & 0.0009 \\ -0.0007 & -0.7377 \\ -0.0005 & -0.6709 \\ 0.0001 & 0.0749 \end{pmatrix}.$$

Tedy máme $dist(ran(\mathbf{Q}_1), ran(\hat{\mathbf{Q}}_1)) = 10^{-5}/sep(\mathbf{T}_{11}, \mathbf{T}_{22}) = 0.0392$. Porovnáním matic \mathbf{Q} a $\hat{\mathbf{Q}}$ zjistíme, že výpočet invariantního podprostoru $ran(\mathbf{Q}_1)$ je citlivý na změny v matici \mathbf{A} .

2.1.3 Citlivost výpočtu vlastního vektoru

Jestliže v předcházejícím odstavci položíme $r = 1$, potom se celá analýza citlivosti výpočtu invariantního podprostoru zredukuje na otázku citlivosti výpočtu vlastního vektoru. Toto je názorně ukázáno v následujícím důsledku.

Důsledek 2.2 *Nechť jsou dány matice $\mathbf{A}, \mathbf{E} \in \mathbb{C}^{n,n}$ a unitární matice*

$\mathbf{Q} = (\mathbf{q}_1 \ \mathbf{Q}_2) \in \mathbb{C}^{n,n}$, kde $\mathbf{q}_1 \in \mathbb{C}^n$. Dále předpokládejme, že

$$\mathbf{Q}^H \mathbf{A} \mathbf{Q} = \begin{pmatrix} \lambda & \mathbf{v}^T \\ \mathbf{o} & \mathbf{T}_{22} \end{pmatrix} \quad \mathbf{Q}^H \mathbf{E} \mathbf{Q} = \begin{pmatrix} \varepsilon & \gamma^T \\ \delta & \mathbf{E}_{22} \end{pmatrix}.$$

Jestliže $\sigma = \sigma_{min}(\mathbf{T}_{22} - \lambda \mathbf{I}) > 0$, kde σ_{min} je nejmenší singulární číslo matice $\mathbf{T}_{22} - \lambda \mathbf{I}$ a

$$\|\mathbf{E}\|_2 \left(1 + \frac{5\|\mathbf{v}\|_2}{\sigma}\right) \leq \frac{\sigma}{5},$$

potom existuje $\mathbf{p} \in \mathbb{C}^{n-1}$ takové, že

$$\|\mathbf{p}\|_2 \leq \frac{4}{\sigma} \|\delta\|_2$$

a $\hat{\mathbf{q}}_1 = (\mathbf{q}_1 + \mathbf{Q}_2 \mathbf{p})/\sqrt{1 + \mathbf{p}^T \mathbf{p}}$ je vlastní vektor matice $\mathbf{A} + \mathbf{E}$. Dále platí

$$dist(span(\mathbf{q}_1), span(\hat{\mathbf{q}}_1)) \leq \frac{4}{\sigma} \|\delta\|_2.$$

Důkaz: viz [5] str. 326.

Poznámka 2.2

Z věty 2.4 a důsledku 2.1 dostaneme $sep(\mathbf{T}_{11}, \mathbf{T}_{22}) = \sigma_{min}(\mathbf{T}_{22} - \lambda \mathbf{I})$, pro $\mathbf{T}_{11} = \lambda$. Vidíme tedy, že singulární číslo $\sigma_{min}(\mathbf{T}_{22} - \lambda \mathbf{I})$ přibližně měří vzdálenost λ od vlastních čísel matice \mathbf{T}_{22} . Musíme říct „přibližně,“ protože

$$sep(\lambda, \mathbf{T}_{22}) = \sigma_{min}(\mathbf{T}_{22} - \lambda \mathbf{I}) \leq \min_{\mu \in \lambda(\mathbf{T}_{22})} |\mu - \lambda|$$

a horní hranice může být dost nadhodnocená, jak jsme viděli např. v příkladě 2.2.

Separace vlastních čísel má vliv na citlivost výpočtu vlastních vektorů. Je-li λ nedefektní vícenásobné vlastní číslo, potom existuje neomezený počet odpovídajících vlastních vektorů, které tvoří příslušný invariantní podprostor. Je tedy zřejmé, že vlastní vektor, odpovídající blízkým vlastním číslům, je nestabilní.

Příklad 2.6 Jestliže

$$\mathbf{A} = \begin{pmatrix} 2.01 & 0.01 \\ 0 & 1.99 \end{pmatrix},$$

potom pro vlastní číslo $\lambda = 1.99$ matice \mathbf{A} platí $1/s(1.99) = 1.2880$ a odpovídající vlastní vektor má tvar $\mathbf{x} = (-0.4472, 0.8922)^T$. Zatímco vlastní číslo $\hat{\lambda} = 2$ blízké matice

$$\mathbf{A} + \mathbf{E} = \begin{pmatrix} 2.01 & 0.01 \\ 0 & 2 \end{pmatrix}$$

má vlastní vektor ve tvaru $\hat{\mathbf{x}} = (-0.7071, 0.7071)^T$. Tedy vlastní vektor, odpovídající vlastnímu číslu 1.99 matice \mathbf{A} , je nestabilní, neboť malá změna v matici \mathbf{A} vyvolala velkou změnu ve vlastním vektoru.

2.2 QR rozklad

Jedna skupina metod, umožňujících výpočet vlastních čísel, vychází z ortogonálních transformací matice \mathbf{A} . Základní myšlenka metody QR rozkladu je

založena na poznatku, že každou reálnou matici \mathbf{A} lze rozložit na součin ortogonální matice \mathbf{Q} a horní trojúhelníkové matice \mathbf{R} . Matice \mathbf{Q} , \mathbf{R} , tvořící QR rozklad matice \mathbf{A} , nejsou určeny jednoznačně. Tento rozklad lze provést několika způsoby, například pomocí Householderovy transformace, nebo Givensovy matice rotace. Nejprve si uvedeme QR rozklad pomocí Householderovy transformace.

Jak lze pomocí QR rozkladu vypočítat vlastní čísla matice \mathbf{A} bude popsáno v kapitolách 2.4 a 2.5.

2.2.1 Householderovy matice

V této podkapitole zadefinujeme pojmy Householderovy matice a Householderova vektoru, které využívá Householderova transformace k realizaci QR rozkladu.

V celé podkapitole budeme předpokládat reálné matice a vektory. Nechť $\mathbf{v} \in \mathbb{R}^n$ je nenulový vektor. Matice $\mathbf{P} \in \mathbb{R}^{n,n}$, definovaná vztahem

$$\mathbf{P} = \mathbf{I} - \frac{2}{\mathbf{v}^T \mathbf{v}} \mathbf{v} \mathbf{v}^T,$$

se nazývá **Householderova matice**. Vektor \mathbf{v} se nazývá **Householderův vektor**. Dále předpokládejme, že máme dán nenulový vektor $\mathbf{x} \in \mathbb{R}^n$ a hledáme matici \mathbf{P} tak, aby $\mathbf{P}\mathbf{x}$ bylo násobkem vektoru $\mathbf{e}_1 = \mathbf{I}(:, 1)$, kde $\mathbf{I}(:, 1)$ označuje první sloupec jednotkové matice, tj. chceme aby byly složky x_2, x_3, \dots, x_n vektoru \mathbf{x} nulové. Potom

$$\mathbf{P}\mathbf{x} = \left(\mathbf{I} - \frac{2\mathbf{v}\mathbf{v}^T}{\mathbf{v}^T \mathbf{v}} \right) \mathbf{x} = \mathbf{x} - \frac{2\mathbf{v}^T \mathbf{x}}{\mathbf{v}^T \mathbf{v}} \mathbf{v}.$$

Položíme-li $\mathbf{v} = \mathbf{x} + \alpha \mathbf{e}_1$, pro $\alpha \in \mathbb{R}$, dostaneme

$$\mathbf{v}^T \mathbf{x} = \mathbf{x}^T \mathbf{x} + \alpha x_1,$$

kde x_1 je první složka vektoru \mathbf{x} a

$$\mathbf{v}^T \mathbf{v} = \mathbf{x}^T \mathbf{x} + 2\alpha x_1 + \alpha^2.$$

Potom

$$\begin{aligned}\mathbf{P}\mathbf{x} &= \mathbf{x} - \frac{2\mathbf{v}^T\mathbf{x}}{\mathbf{v}^T\mathbf{v}}\mathbf{v} = \mathbf{x} - \frac{2\mathbf{v}^T\mathbf{x}}{\mathbf{v}^T\mathbf{v}}(\mathbf{x} + \alpha\mathbf{e}_1) = \\ &= \mathbf{x} - \frac{2\mathbf{v}^T\mathbf{x}}{\mathbf{v}^T\mathbf{v}}\mathbf{x} - 2\alpha\frac{\mathbf{v}^T\mathbf{x}}{\mathbf{v}^T\mathbf{v}}\mathbf{e}_1 = \left(1 - \frac{2\mathbf{v}^T\mathbf{x}}{\mathbf{v}^T\mathbf{v}}\right)\mathbf{x} - 2\alpha\frac{\mathbf{v}^T\mathbf{x}}{\mathbf{v}^T\mathbf{v}}\mathbf{e}_1.\end{aligned}$$

Tedy $\mathbf{P}\mathbf{x}$ lze psát ve tvaru

$$\mathbf{P}\mathbf{x} = \left(1 - 2\frac{\mathbf{x}^T\mathbf{x} + \alpha x_1}{\mathbf{x}^T\mathbf{x} + 2\alpha x_1 + \alpha^2}\right)\mathbf{x} - 2\alpha\frac{\mathbf{v}^T\mathbf{x}}{\mathbf{v}^T\mathbf{v}}\mathbf{e}_1. \quad (13)$$

Aby byly složky x_2, x_3, \dots, x_n vektoru \mathbf{x} nulové, zvolíme $\alpha = \pm\|\mathbf{x}\|_2$, potom

$$\mathbf{v} = \mathbf{x} \pm \|\mathbf{x}\|_2\mathbf{e}_1$$

a ze vztahu (13) plyne

$$\mathbf{P}\mathbf{x} = \mp\|\mathbf{x}\|_2\mathbf{e}_1. \quad (14)$$

Volba vektoru \mathbf{v} v tomto tvaru se ukáže při výpočtu QR rozkladu pomocí Householderovy transformace značně užitečná, což si ukážeme později.

Příklad 2.7 Nechť je dán vektor $\mathbf{x} = (1, 2, 4, 2)^T$. Najděme Householderův vektor \mathbf{v} a Householderovu matici \mathbf{P} tak, aby součin $\mathbf{P}\mathbf{x}$ byl násobkem $\mathbf{e}_1 = \mathbf{I}(:, 1)$.

Užitím vztahu $\mathbf{v} = \mathbf{x} + \|\mathbf{x}\|_2\mathbf{e}_1$ dostaneme Householderův vektor ve tvaru $\mathbf{v} = (6, 2, 4, 2)^T$. Potom Householderova matice

$$\mathbf{P} = \mathbf{I} - 2\frac{\mathbf{v}\mathbf{v}^T}{\mathbf{v}^T\mathbf{v}} = -\frac{1}{30} \begin{pmatrix} 6 & 12 & 24 & 12 \\ 12 & -26 & 8 & 4 \\ 24 & 8 & -14 & 8 \\ 12 & 4 & 8 & -26 \end{pmatrix}$$

má tu vlastnost, že $\mathbf{P}\mathbf{x} = (-5, 0, 0, 0)^T$.

Existuje celá řada praktických detailů spojených se stanovením Householderova vektoru. Jednou z možností je volba složky

$$v_1 = x_1 - \|\mathbf{x}\|_2$$

vektoru \mathbf{v} podle vztahu (14). Pak $\mathbf{P}\mathbf{x}$ je kladným násobkem \mathbf{e}_1 . Ale toto nastavení není vhodné v případě, kdy vektor \mathbf{x} je blízký vektoru \mathbf{e}_1 . Z literatury se ukazuje, že je užitečné volit $v_1 = 1$. Dále označíme $\beta = \frac{2}{\mathbf{v}^T \mathbf{v}}$ a výše popsaný výpočet Householderova vektoru shrneme v následujícím algoritmu.

ALGORITMUS 2.1

Nechť je dán vektor $\mathbf{x} \in \mathbb{R}^n$, potom algoritmus vypočítá vektor $\mathbf{v} \in \mathbb{R}^n$, kde $\mathbf{v}(1) = 1$ a $\beta \in \mathbb{R}$ tak, že matice $\mathbf{P} = \mathbf{I}_n - \beta \mathbf{v} \mathbf{v}^T$ je ortogonální a platí $\mathbf{P}\mathbf{x} = \|\mathbf{x}\|_2 \mathbf{e}_1$.

1. zvolíme $\mathbf{x} \in \mathbb{R}^n$

2. vypočteme

(a) $\sigma = \mathbf{x}(2:n)^T \mathbf{x}(2:n)$

(b) $\mathbf{v} = (1, \mathbf{x}(2:n))^T$

3. je-li $\sigma = 0$, pak položíme $\beta = 0$, jinak vypočteme $\mu = \sqrt{\mathbf{x}(1)^2 + \sigma}$

(a) jestliže $\mathbf{x}(1) \leq 0$, položíme $\mathbf{v}(1) = \mathbf{x}(1) - \mu$, jinak položíme

$$\mathbf{v}(1) = -\frac{\sigma}{\mathbf{x}(1) + \mu}$$

(b) vypočteme

i. $\beta = \frac{2\mathbf{v}(1)^2}{\sigma + \mathbf{v}(1)^2}$

ii. $\mathbf{v} = \frac{\mathbf{v}}{\mathbf{v}(1)}$

Příklad 2.8 Nechť je dán vektor $\mathbf{x} = (1, 2, 4, 2)^T$ z příkladu 2.7. Vypočítejme Householderův vektor \mathbf{v} pomocí algoritmu 2.1.

Dle bodu 2 vypočteme

$$\sigma = (2 \ 4 \ 2) \begin{pmatrix} 2 \\ 4 \\ 2 \end{pmatrix} = 24 \quad a \quad \mathbf{v} = \begin{pmatrix} 1 \\ 2 \\ 4 \\ 2 \end{pmatrix}.$$

Dále vypočteme $\mu = \sqrt{\mathbf{x}(1)^2 + \sigma} = \sqrt{1 + 24} = 5$ a položíme $\mathbf{v}(1) = -\frac{24}{1+5} = -4$.

Pak vektor \mathbf{v} bude mít tvar $\mathbf{v} = (-4 \ 2 \ 4 \ 2)^T$. Dle vztahu $\beta = \frac{2\mathbf{v}(1)^2}{\sigma + \mathbf{v}(1)^2}$ vypočteme $\beta = 0.8$ a položíme

$$\mathbf{v} = \frac{\mathbf{v}}{\mathbf{v}(1)} = -\frac{1}{4} \begin{pmatrix} -4 \\ 2 \\ 4 \\ 2 \end{pmatrix} = \begin{pmatrix} 1 \\ -0.5 \\ -1 \\ -0.5 \end{pmatrix}.$$

Potom snadno ověříme, že Householderova matice

$$\mathbf{P} = \frac{1}{5} \begin{pmatrix} 1 & 2 & 4 & 2 \\ 2 & 4 & -2 & -1 \\ 4 & -2 & 1 & -2 \\ 2 & -1 & -2 & 4 \end{pmatrix}$$

je ortogonální a platí $\mathbf{P}\mathbf{x} = (5, 0, 0, 0)^T$.

Dle algoritmu 2.1 jsem sestavila m-file (viz příloha procedura house.m), který využijeme dále v m-filu pro výpočet QR rozkladu matice \mathbf{A} .

2.2.2 Výpočet QR rozkladu pomocí Householderovy transformace

Definujme Householderovu transformaci matice $\mathbf{A} \in \mathbb{R}^{n,n}$ vztahem

$$\mathbf{A}_j = \mathbf{P}_j \mathbf{A}_{j-1}, \quad j = 1, \dots, n,$$

kde $\mathbf{A}_0 = \mathbf{A}$ a Householderovy matice $\mathbf{P}_j \in \mathbb{R}^{n,n}$ budou pro $j = 1, \dots, n$ ve tvaru

$$\mathbf{P}_j = \mathbf{I} - \beta_j \mathbf{v}^{(j)} \mathbf{v}^{(j)T},$$

kde $\mathbf{v}^{(j)} \in \mathbb{R}^n$ jsou Householderovy vektory a $\beta_j = 2/\mathbf{v}^{(j)T} \mathbf{v}^{(j)}$.

Podstatu Householderovy transformace lze vysvětlit na malém příkladu. Nechť $n = 5$, pak jsou Householderovy matice \mathbf{P}_1 a \mathbf{P}_2 zkonstruovány tak, aby platilo

$$\mathbf{P}_2 \mathbf{P}_1 \mathbf{A} = \begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & * & \times & \times \\ 0 & 0 & * & \times & \times \\ 0 & 0 & * & \times & \times \end{pmatrix}.$$

Pomocí zvýrazněných položek $*$ určíme matici $\tilde{\mathbf{P}}_3 \in \mathbb{R}^{3,3}$ tak, aby platilo

$$\tilde{\mathbf{P}}_3 \begin{pmatrix} * \\ * \\ * \end{pmatrix} = \begin{pmatrix} \times \\ 0 \\ 0 \end{pmatrix}.$$

Jestliže $\mathbf{P}_3 = \text{diag}(\mathbf{I}_2, \tilde{\mathbf{P}}_3)$, potom

$$\mathbf{P}_3 \mathbf{P}_2 \mathbf{P}_1 \mathbf{A} = \begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & \times & \times \end{pmatrix}.$$

Takto lze induktivně pokračovat dále. Celkem tak po n opakováních dostaneme horní trojúhelníkovou matici

$$\mathbf{A}_n = \mathbf{P}_n \mathbf{P}_{n-1} \dots \mathbf{P}_1 \mathbf{A}.$$

Označme $\mathbf{R} = \mathbf{A}_n$. Snadno lze ověřit, že pro $j = 1, \dots, n$ platí $\mathbf{P}_j = \mathbf{P}_j^{-1}$, neboť matice \mathbf{P}_j jsou ortogonální. Potom volbou $\mathbf{Q} = \mathbf{P}_1 \dots \mathbf{P}_n$ dostaneme ortogonální rozklad matice \mathbf{A} ve tvaru $\mathbf{A} = \mathbf{Q}\mathbf{R}$.

V průběhu výpočtu není nutné vytvářet matici \mathbf{Q} explicitně, tj. provádět násobení n matic. Tím se ušetří nejen čas výpočtu, ale zmenší se i počet zaokrouhlovacích chyb. Vychází se z poznatku, že v r -tém kroku, kde $r \leq n$, je

$$\mathbf{Q} = \mathbf{P}_1 \mathbf{P}_2 \dots \mathbf{P}_r, \quad \mathbf{P}_j = \mathbf{I} - \beta_j \mathbf{v}^{(j)} \mathbf{v}^{(j)T},$$

a každé $\mathbf{v}^{(j)}$ volíme tak, aby jeho prvních $j - 1$ složek bylo rovno nule a j -tá složka byla rovna 1, tj.

$$\mathbf{v}^{(j)} = (0, 0, \dots, 0, 1, v_{j+1}^{(j)}, \dots, v_n^{(j)})^T.$$

Householderovy vektory $\mathbf{v}^{(j)}$ vypočítáme ze zobecněného algoritmu 2.1.

ZOBECNĚNÝ ALGORITMUS 2.1

Algoritmus 2.1 na výpočet Householderova vektoru \mathbf{v} , kde $\mathbf{v}(1) = 1$, lze zobecnit pro výpočet vektoru $\mathbf{v}^{(j)}$, kde $j = 1, \dots, n$ následovně. V bodě 2 nahradíme vektor $\mathbf{x}(2 : n)$ vektorem $\mathbf{x}(j + 1 : n)$ a vektor $\mathbf{v}^{(j)}$ vypočítáme takto

1. $\mathbf{v}^{(j)}(1 : j - 1) = \mathbf{o}$
2. $\mathbf{v}^{(j)}(j) = 1$
3. $\mathbf{v}^{(j)}(j + 1 : n) = \mathbf{x}(j + 1 : n)$.

Potom také v bodě 3 algoritmu 2.1 nahradíme $\mathbf{x}(1)$, $\mathbf{v}(1)$ hodnotami $\mathbf{x}(j)$ a $\mathbf{v}^{(j)}(j)$.

Uvedený postup výpočtu QR rozkladu matice \mathbf{A} užitím Householderovy transformace shrneme v algoritmu.

ALGORITMUS 2.2

Nechť je dána matice $\mathbf{A} \in \mathbb{R}^{n,n}$. Algoritmus najde QR rozklad matice \mathbf{A} , tj. najde ortogonální matici \mathbf{Q} a horní trojúhelníkovou matici \mathbf{R} takové, že platí $\mathbf{A} = \mathbf{QR}$.

1. položíme $\mathbf{R} = \mathbf{A}$,
2. pro $j = 1, \dots, n$
 - (a) vybereme z matice \mathbf{R} vektor $\mathbf{x} = \mathbf{R}(j : n, j)$
 - (b) dle zobecněného algoritmu 2.1 vypočteme vektor $\mathbf{v}^{(j)}$ a položíme

$$\mathbf{R}(j : n, j : n) = (\mathbf{I}_{n-j+1} - \beta \mathbf{v}^{(j)} \mathbf{v}^{(j)T}) \mathbf{R}(j : n, j : n)$$
3. položíme $\mathbf{Q} = \mathbf{I}_n$,
4. pro $r = j : -1 : 1$ položíme
 - (a) $\mathbf{v}^{(j)}(r : n) = (1, \mathbf{R}^T(r + 1 : n, r))^T$
 - (b) $\mathbf{Q}(r : n, r : n) = (\mathbf{I}_{n-r+1} - \beta \mathbf{v}^{(j)}(r : n) \mathbf{v}^{(j)}(r : n)^T) \mathbf{Q}(r : n, r : n)$

Na základě výše popsaného algoritmu sestavíme m-file na QR rozklad matice \mathbf{A} (viz příloha procedura QRHouse.m).

Příklad 2.9 Rozložme matici

$$\mathbf{A} = \begin{pmatrix} 1 & 0 & 3 & -4 \\ 1 & 5 & 6 & -2 \\ -3 & 4 & 0 & 1 \\ 2 & 0 & 7 & -1 \end{pmatrix}$$

na součin ortogonální matice \mathbf{Q} a horní trojúhelníkové matice \mathbf{R} pomocí Householderových transformací.

Užitím procedury `QRHouse.m` dostaneme

$$\mathbf{Q} = \begin{pmatrix} 0.2582 & 0.0760 & 0.2506 & 0.9299 \\ 0.2582 & 0.8899 & -0.3734 & -0.0438 \\ -0.7746 & 0.4233 & 0.4668 & 0.0547 \\ 0.5164 & 0.1519 & 0.7615 & -0.3610 \end{pmatrix}$$

a

$$\mathbf{R} = \begin{pmatrix} 3.8730 & -1.8074 & 5.9386 & -2.8402 \\ 0 & 6.1427 & 6.6311 & -1.8124 \\ 0 & 0 & 3.8421 & -0.5504 \\ 0 & 0 & 0 & -3.2164 \end{pmatrix}.$$

Lze snadno ověřit, že platí $\mathbf{A} = \mathbf{QR}$.

Householderova transformace tvoří základ metod, které lze užít k výpočtu vlastních čísel libovolné reálné matice.

2.2.3 Givensovy matice rotace

Než si uvedeme, jak lze použít Givensovy matice rotace k provedení QR rozkladu, zdefinujme pojem Givensovy matice rotace. Givensovou maticí rotace nazýváme matici, která vznikne z jednotkové matice tak, že má pozice (i, i) , (k, k) obsazeny parametrem c a pozice (i, k) resp. (k, i) jsou obsazeny parametrem s

resp. $-s$, kde $c = \cos(\theta)$ a $s = \sin(\theta)$ pro nějaký úhel θ a $i, k \in \{1, \dots, n\}$, tj.

$$\mathbf{G}(i, k, \theta) = \begin{pmatrix} 1 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & & \vdots & & \vdots \\ 0 & \cdots & c & \cdots & s & \cdots & 0 \\ \vdots & & \vdots & \ddots & \vdots & & \vdots \\ 0 & \cdots & -s & \cdots & c & \cdots & 0 \\ \vdots & & \vdots & & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & 0 & \cdots & 1 \end{pmatrix}. \quad (15)$$

Snadno lze ověřit, že $\mathbf{G}(i, k, \theta)^T \mathbf{G}(i, k, \theta) = \mathbf{I}$, tedy Givensovy matice rotace jsou ortogonální.

Givensova matice $\mathbf{G}(i, k, \theta)$ představuje rotaci prvků o úhel θ proti směru hodinových ručiček na pozicích (i, k) . Skutečně, je-li $\mathbf{x} \in \mathbb{R}^n$ a $\mathbf{y} = \mathbf{G}(i, k, \theta)^T \mathbf{x}$, potom

$$y_j = \begin{cases} cx_i - sx_k & j = i \\ sx_i + cx_k & j = k \\ x_j & j \neq i, k \end{cases}.$$

Z tohoto vztahu je zřejmé, že položíme-li

$$c = \frac{x_i}{\sqrt{x_i^2 + x_k^2}}, \quad s = \frac{-x_k}{\sqrt{x_i^2 + x_k^2}} \quad (16)$$

dostaneme $y_k = 0$. Toto je jednoduchý způsob, jak pomocí Givensovy matice rotace vynulovat jistou složku daného vektoru.

Postup shrneme v algoritmu.

ALGORITMUS 2.3 Nechť je dán vektor $\mathbf{x} \in \mathbb{R}^n$. Algoritmus vypočítá parametry $c = \cos(\theta)$ a $s = \sin(\theta)$ tak, aby platilo

$$\begin{pmatrix} c & s \\ -s & c \end{pmatrix}^T \begin{pmatrix} x_k \\ x_i \end{pmatrix} = \begin{pmatrix} r \\ 0 \end{pmatrix},$$

tj. pomocí k -té složky vektoru \mathbf{x} se vynuluje jeho i -tá složka.

1. z vektoru \mathbf{x} vybereme složky x_k a x_i

2. jestliže $x_i = 0$ položíme

$$c = 1, s = 0,$$

jinak jestliže platí $|x_i| > |x_k|$ vypočteme

$$\tau = -x_k/x_i, s = 1/\sqrt{1 + \tau^2}, c = s\tau,$$

v opačném případě počítáme

$$\tau = -x_i/x_k, c = 1/\sqrt{1 + \tau^2}, s = c\tau$$

Příklad 2.10 Nechť je dán vektor $\mathbf{x} = (2, 7, 1, 3)^T$. Pomocí Givensovy matice rotace vynulujeme poslední prvek ve vektoru \mathbf{x} .

Vybereme např. prvek $x_2 = 7$ a prvek $x_4 = 3$.

Postupujeme dle algoritmu 2.3. Protože $|x_2| > |x_4|$ vypočteme $\tau = -\frac{x_4}{x_2} = -\frac{3}{7}$,

$$c = \frac{1}{\sqrt{1 + (-3/7)^2}} = \frac{7}{\sqrt{58}} = 0.9191 \text{ a } s = -\frac{3}{7} \frac{7}{\sqrt{58}} = -\frac{3}{\sqrt{58}} = -0.3939.$$

Givensova matice rotace bude ve tvaru

$$\mathbf{G}(2, 4, \theta) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & c & 0 & s \\ 0 & 0 & 1 & 0 \\ 0 & -s & 0 & c \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0.9191 & 0 & -0.3939 \\ 0 & 0 & 1 & 0 \\ 0 & 0.3939 & 0 & 0.9191 \end{pmatrix}.$$

Potom pro $\mathbf{y} = \mathbf{G}(2, 4, \theta)^T \mathbf{x}$ dostáváme

$$\mathbf{y} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0.9191 & 0 & 0.3939 \\ 0 & 0 & 1 & 0 \\ 0 & -0.3939 & 0 & 0.9191 \end{pmatrix} \begin{pmatrix} 2 \\ 7 \\ 1 \\ 3 \end{pmatrix} = \begin{pmatrix} 2 \\ 7.6158 \\ 1 \\ 0 \end{pmatrix}.$$

Na základě algoritmu 2.3 sestavíme m-file (viz příloha procedura givens.m), který využijeme dále v m-filu pro výpočet QR rozkladu pomocí Givensových matic rotace.

2.2.4 Výpočet QR rozkladu pomocí Givensových matic rotace

Jednoduchou strukturu Givensovy matice rotace lze využít pro výpočet QR rozkladu matice. Předpokládejme, že je dána reálná matice $\mathbf{A} \in \mathbb{R}^{n,n}$ a parametry $c = \cos(\theta)$ a $s = \sin(\theta)$. Definujme transformaci

$$\mathbf{A}_j = \mathbf{G}_j^T \mathbf{A}_{j-1}, \quad \mathbf{A}_0 = \mathbf{A},$$

kde $\mathbf{A}_{j-1} = (a_{lm}^{j-1})_{l,m=1}^n$, $a_{lm}^{j-1} \in \mathbb{R}$ a $\mathbf{G}_j = \mathbf{G}(i, k, \theta_j)$ je Givensova matice rotace, pro $j = 1, \dots, t$, kde $t = \frac{(n-1)n}{2}$ označuje počet poddiagonálních prvků. Za indexy i, k volíme postupně $k = 1, \dots, n$, $i = n, n-1, \dots, k+1$. Potom násobení matice \mathbf{A}_{j-1} zleva maticí $\mathbf{G}_j^T = \mathbf{G}^T(i, k, \theta_j)$ se projeví ve dvou řádcích matice \mathbf{A} , což lze stručně zapsat takto

$$\mathbf{A}((i, k), :) = \begin{pmatrix} c & s \\ -s & c \end{pmatrix}^T \mathbf{A}((i, k), :),$$

tj. vynuluje se prvek na pozici (i, k) . Tímto způsobem tedy lze vynulovat všechny nenulové prvky pod diagonálou a tím obdržíme

$$\mathbf{R} = \mathbf{G}_t^T \dots \mathbf{G}_1^T \mathbf{A} = \mathbf{Q}^T \mathbf{A}, \quad \mathbf{Q} = \mathbf{G}_1 \dots \mathbf{G}_t.$$

Potom

$$\mathbf{A} = \mathbf{Q}\mathbf{R},$$

kde \mathbf{Q} je ortogonální a \mathbf{R} je horní trojúhelníková matice.

Poznámka 2.3 Pokud prvek a_{ik}^{j-1} je již nulový, potom matice $\mathbf{G}_j = \mathbf{G}(i, k, \theta_j)$ je rovna jednotkové matici a provádíme tedy jen tolik kroků, kolik je nenulových poddiagonálních prvků. Proto je vhodné tuto metodu použít pro QR rozklad řídkých matic.

Následující algoritmus provede QR rozklad matice \mathbf{A} užitím Givensových matic.

ALGORITMUS 2.4

Nechť $\mathbf{A} \in \mathbb{R}^{n,n}$, potom algoritmus počítá horní trojúhelníkovou matici \mathbf{R} a ortogonální matici \mathbf{Q} , pro které platí $\mathbf{QR} = \mathbf{A}$.

1. položíme $\mathbf{R} = \mathbf{A}$ a sloupce matice \mathbf{R} označíme \mathbf{r}_i , pro $i = 1, \dots, n$
2. pro $k = 1, \dots, n$, $i = n, \dots, k + 1$
 - (a) z vektoru $\mathbf{r}_k = (r_{k1}, \dots, r_{kn})^T$ vybereme prvky $x_k = r_{kk}$ a $x_i = r_{ik}$
 - (b) podle algoritmu 2.3 vypočítáme parametry c, s
 - (c) položíme $\mathbf{R}([k, i], k : n) = \begin{pmatrix} c & -s \\ s & c \end{pmatrix} \mathbf{R}([k, i], k : n)$
 - (d) položíme $\mathbf{Q}(:, [k, i]) = \mathbf{Q}(:, [k, i]) \begin{pmatrix} c & s \\ -s & c \end{pmatrix}$

Na základě výše popsaného algoritmu jsem sestavila m-file na QR rozklad matice \mathbf{A} (viz příloha procedura QRGivens.m).

Poznámka 2.4 Protože počítač nezná absolutní nulu, v m-filech počítáme s jistou odchylkou ε , kde ε je malé kladné číslo, např. $\varepsilon = 10^{-5}$. Pokud v druhém kroku algoritmu 2.3 je $|x_i| \leq \varepsilon$, bereme $x_i = 0$.

Příklad 2.11 Užitím Givensových matic rotace rozložme matici

$$\mathbf{A} = \begin{pmatrix} 2 & 3 & 1 & 0 \\ 1 & 2 & 1 & 1 \\ 3 & 1 & 1 & -2 \\ -4 & -2 & 1 & 6 \end{pmatrix}$$

na součin ortogonální matice \mathbf{Q} a horní trojúhelníkové matice \mathbf{R} .

Užitím procedury QRGivens.m dostaneme

$$\mathbf{Q} = \begin{pmatrix} 0.3651 & -0.7096 & 0.0449 & -0.6009 \\ 0.1826 & -0.5595 & 0.1900 & 0.7858 \\ 0.5477 & 0.3684 & 0.7497 & -0.0462 \\ -0.7303 & -0.2183 & 0.6323 & -0.1387 \end{pmatrix}$$

a

$$\mathbf{R} = \begin{pmatrix} 5.4772 & 3.4689 & 0.3651 & -5.2947 \\ 0 & -2.4427 & -1.1190 & -2.6064 \\ 0 & 0 & 1.6169 & 2.4842 \\ 0 & 0 & 0 & 0.0462 \end{pmatrix}.$$

Snadno lze ověřit, že platí $\mathbf{A} = \mathbf{QR}$.

2.3 Mocninná metoda

Mocninná metoda se užívá k výpočtu vlastního čísla dané matice, které je v absolutní hodnotě maximální, pokud je takové vlastní číslo jediné a jednoduché. Myšlenku této metody vysvětlíme za předpokladu, že všichni elementární dělitelé matice \mathbf{A} jsou lineární, tj. vlastní čísla $\lambda_1, \lambda_2, \dots, \lambda_n$ matice $\mathbf{A} \in \mathbb{C}^{n,n}$ jsou navzájem různá. Pak můžeme říct, že existuje n lineárně nezávislých vlastních vektorů matice \mathbf{A} , které tvoří n -rozměrný lineární prostor \mathbb{C}^n .

Nechť je $\mathbf{A} \in \mathbb{C}^{n,n}$ diagonalizovatelná matice, tj. $\mathbf{X}^{-1}\mathbf{A}\mathbf{X} = \text{diag}(\lambda_1, \dots, \lambda_n)$, kde matice $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_n)$ je tvořena lineárně nezávislými vlastními vektory matice \mathbf{A} a vlastní čísla jsou uspořádána následovně $|\lambda_1| > |\lambda_2| \geq \dots \geq |\lambda_n|$. Zvolme libovolný počáteční vektor $\mathbf{q}_0 \in \mathbb{C}^n$, ten lze vyjádřit jako lineární kombinaci vlastních vektorů, tj.

$$\mathbf{q}_0 = \sum_{i=1}^n \alpha_i \mathbf{x}_i,$$

kde $\alpha_i \in \mathbb{C}$, pro $i = 1, \dots, n$ a navíc platí $\alpha_1 \neq 0$. Pomocí iteračního předpisu

$$\mathbf{q}_k = \mathbf{A}\mathbf{q}_{k-1} = \mathbf{A}\mathbf{A}\mathbf{q}_{k-2} = \dots = \mathbf{A}^k \mathbf{q}_0 \quad \text{pro } k = 1, 2, \dots \quad (17)$$

vytvoříme posloupnost vektorů $\{\mathbf{q}_k\}$. Z lemmatu 1.1 víme, že $\mathbf{A}^k \mathbf{x}_i = \lambda_i^k \mathbf{x}_i$, pro $i = 1, \dots, n$ a $k = 1, 2, \dots$. Potom lze vektor \mathbf{q}_k psát ve tvaru

$$\mathbf{q}_k = \mathbf{A}^k \mathbf{q}_0 = \mathbf{A}^k \sum_{i=1}^n \alpha_i \mathbf{x}_i = \sum_{i=1}^n \alpha_i \mathbf{A}^k \mathbf{x}_i = \sum_{i=1}^n \alpha_i \lambda_i^k \mathbf{x}_i. \quad (18)$$

Na základě těchto poznatků můžeme vyslovit větu.

Věta 2.5 (Von Mises) *Nechť matice $\mathbf{A} \in \mathbb{C}^{n,n}$ má n lineárně nezávislých vlastních vektorů \mathbf{x}_i , pro $i = 1, \dots, n$ a nechť λ_1 je vlastní číslo, které je v absolutní hodnotě největší ze všech vlastních čísel matice \mathbf{A} a ostatní vlastní čísla jsou seřazena následovně*

$$|\lambda_1| > |\lambda_2| \geq \dots \geq |\lambda_n|.$$

Potom pro vektory $\mathbf{q}_k \in \mathbb{C}^n$ ze vztahu (17) platí

$$\lim_{k \rightarrow \infty} \frac{1}{\lambda_1^k} \mathbf{q}_k = \alpha_1 \mathbf{x}_1 \quad (19)$$

$$\lim_{k \rightarrow \infty} \frac{\mathbf{z}^T \mathbf{q}_{k+1}}{\mathbf{z}^T \mathbf{q}_k} = \lambda_1, \quad (20)$$

kde $\mathbf{q}_0 = \sum_{i=1}^n \alpha_i \mathbf{x}_i$ je libovolný vektor s koeficientem $\alpha_1 \neq 0$ a \mathbf{z} je libovolný vektor s vlastností $\mathbf{z}^T \mathbf{x}_1 \neq 0$.

Důkaz: viz [2] str. 120.

Podívejme se na vlastnosti konvergence mocninné metody. Ze vztahu (18) plyne

$$\mathbf{q}_k = \mathbf{A}^k \mathbf{q}_0 = \alpha_1 \lambda_1^k \left(\mathbf{x}_1 + \sum_{i=2}^n \frac{\alpha_i}{\alpha_1} \left(\frac{\lambda_i}{\lambda_1} \right)^k \mathbf{x}_i \right). \quad (21)$$

Vzhledem k tomu, že $\mathbf{q}_k \in \text{span}\{\mathbf{A}^k \mathbf{q}_0\}$ je

$$\text{dist}(\text{span}\{\mathbf{q}_k\}, \text{span}\{\mathbf{x}_1\}) = O\left(\left|\frac{\lambda_2}{\lambda_1}\right|^k\right)$$

a navíc

$$|\lambda_1 - \lambda_1^{(k)}| = O\left(\left|\frac{\lambda_2}{\lambda_1}\right|^k\right),$$

kde $\lambda_1^{(k)}$ je aproximace vlastního čísla λ_1 . Jestliže platí $|\lambda_1| > |\lambda_2| \geq \dots \geq |\lambda_n|$, potom říkáme, že λ_1 je **dominantní vlastní číslo**. Tedy mocninná metoda kon-

verguje, jestliže je λ_1 dominantní vlastní číslo a \mathbf{q}_0 má složku ve směru odpovídajícího dominantního vlastního vektoru \mathbf{x}_1 , tj. vektor \mathbf{q}_0 je ortogonální k dominantnímu vlastnímu vektoru \mathbf{x}_1 .

Ze vztahu (21) vidíme, že v případě $|\lambda_1| > 1$ bude $\mathbf{q}_k \rightarrow \infty$ a naopak pro $|\lambda_1| < 1$ bude $\mathbf{q}_k \rightarrow 0$. Z hlediska numerického výpočtu je výhodné a doporučené vektor \mathbf{q}_k normovat v každém kroku. Tak získáme algoritmus vhodný k výpočtu vlastního čísla λ_1 .

ALGORITMUS 2.5

Nechť je dána čtvercová matice $\mathbf{A} \in \mathbb{C}^{n,n}$ a požadovaná přesnost výpočtu ε . Algoritmus vypočítá číslo $\lambda_1^{(k)}$, které je v absolutní hodnotě aproximací maximálního vlastního čísla λ_1 s chybou ε a vektor \mathbf{q}_k , který je aproximací normovaného vlastního vektoru \mathbf{x}_1 k příslušnému vlastnímu číslu $\lambda_1^{(k)}$.

1. zvolíme $\mathbf{q}_0 \in \mathbb{C}^n$ a položíme $k = 1$

2. vypočítáme

(a) $\mathbf{z}_k = \mathbf{A}\mathbf{q}_{k-1}$

(b) $\mathbf{q}_k = \mathbf{z}_k / \|\mathbf{z}_k\|_2$

(c) $\lambda_1^{(k)} = \mathbf{q}_k^H \mathbf{A} \mathbf{q}_k$

3. jestliže platí $\left\| \frac{\lambda_1^{(k)} - \lambda_1^{(k-1)}}{\lambda_1^{(k-1)}} \right\| > \varepsilon$, položíme $k = k + 1$ a vrátíme se do kroku 2, jinak výpočet ukončíme

Na výpočet vlastního čísla o největší absolutní hodnotě a příslušného normovaného vlastního vektoru jsem vytvořila m-file (viz příloha procedura MocMet.m), který využijeme v následujícím příkladě.

Příklad 2.12 Vypočtíme vlastní číslo matice

$$\mathbf{A} = \begin{pmatrix} 2 & 0 & -3 \\ 1 & 5 & 0 \\ 4 & 3 & 2 \end{pmatrix},$$

kteře je v absolutní hodnotě největší. Zvolme $\mathbf{q}_0 = (1, 1, 1)^T$. Výsledky výpočtů (pro různou požadovanou přesnost), jež jsme získali užitím procedury MocMet.m, jsou uvedeny v následující tabulce

ε	k	$\lambda_1^{(k)}$	\mathbf{q}_k
10^{-5}	66	4.50884	$(-0.41396, 0.84179, 0.34644)^T$
10^{-7}	105	4.5079247	$(-0.4142050, 0.8417443, 0.3462669)^T$
10^{-10}	160	4.5079185552	$(-0.4142066323, 0.8417440648, 0.3462654980)^T$

kde $\lambda_1^{(k)}$ je aproximace vlastního čísla λ_1 , \mathbf{q}_k je aproximace normovaného vlastního vektoru \mathbf{x}_1 k příslušnému vlastnímu číslu $\lambda_1^{(k)}$ a k udává počet iterací potřebných k dosažení tohoto výsledku s požadovanou přesností výpočtu ε .

Program MATLAB nabízí pro výpočet vlastních čísel matice \mathbf{A} vlastní příkaz $\text{eig}(\mathbf{A})$. Užitím tohoto příkazu bylo vypočteno vlastní číslo $\lambda_1 = 4.5079185611$, které je v absolutní hodnotě maximální. Tato hodnota je brána jako referenční. Hodnoty vlastního čísla, vypočtené pomocí mnou navrženého m-filu, se uspokojivě přiblíží k referenční hodnotě při přesnosti $\varepsilon = 10^{-10}$ (viz tabulka výše).

Ze vztahu (21) vidíme, že konvergence mocninné metody bude záviset od rychlosti, se kterou se bude podíl $|\lambda_2/\lambda_1|^k$ blížit k nule. Číslo $|\lambda_2/\lambda_1|$ tedy charakterizuje rychlost konvergence. Je zřejmé, že pro $|\lambda_2|$ blízké $|\lambda_1|$ může být konvergence neefektivně pomalá. O zrychlení konvergence se podrobněji hovoří např. v [2].

Chybu $|\lambda_1^{(k)} - \lambda_1|$ je možné odhadnout použitím teorie citlivosti výpočtu, popsané v předcházející části. Definujme vektor $\mathbf{r}_k = \mathbf{A}\mathbf{q}_k$ a všimněme si, že $(\mathbf{A} + \mathbf{E}_k)\mathbf{q}_k = \lambda_1^{(k)}\mathbf{q}_k$, kde $\mathbf{E}_k = -\mathbf{r}_k\mathbf{q}_k^H$. Potom $\lambda_1^{(k)}$ je vlastní číslo matice $\mathbf{A} + \mathbf{E}_k$ a

$$\left| \lambda_1^{(k)} - \lambda_1 \right| \approx \frac{\|\mathbf{E}_k\|_2}{s(\lambda_1)} = \frac{\|\mathbf{r}_k\|_2}{s(\lambda_1)}.$$

Jestliže použijeme mocninnou metodu na výpočet přibližných levých a pravých vlastních vektorů, potom je možné získat odhad $s(\lambda_1)$. Nechť $\mathbf{w}_k = (\mathbf{A}^k)^H \mathbf{w}_0$, potom můžeme použít aproximaci $s(\lambda_1) \approx |\mathbf{w}_k^H \mathbf{q}_k|$.

Ze znalosti vlastního čísla λ_1 a vlastního vektoru \mathbf{x}_1 lze dopočítat další vlastní

čísla matice \mathbf{A} . Tímto výpočtem se zabývá tzv. metoda maticové redukce. Tato metoda spočívá v postupné redukci matice \mathbf{A} řádu n na matici řádu $n - 1$. Podrobněji jsem se touto metodou zabývala ve své bakalářské práci (viz [9]).

Jestliže má matice \mathbf{A} víc blízkých vlastních čísel, nemusí se podařit určení λ_1 a \mathbf{x}_1 s dostatečnou přesností a případný výpočet dalších vlastních čísel a vlastních vektorů je potom málo hodnotný, až nemožný. V takových případech je výhodnější použít zobecněnou mocninnou metodu.

Zobecněná mocninná metoda může být použita pro výpočet r -tice největších vlastních čísel v absolutní hodnotě. Tato metoda je náročná na počet operací, proto se v praxi často nepoužívá a upřednostňují se metody jiné. Např. metoda QR transformace, která počítá všechna vlastní čísla matice \mathbf{A} .

2.4 Metoda QR transformace

Jak jsme již zmínili, podobné matice mají stejná vlastní čísla. Mnoho metod pro výpočet vlastních čísel matice je proto založeno na převodu matice na matici s ní podobnou, která má jednodušší tvar. Jelikož většina problémů výpočtu vlastních čísel obsahuje reálná data, budeme se dále zabývat pouze maticemi s reálnými prvky.

Jednou z neznámějších metod pro výpočet vlastních čísel matice, která je založena na převodu matice na matici s ní podobnou, je metoda QR transformace. Tato vychází z poznatku, že libovolnou matici \mathbf{A} lze rozložit na součin ortogonální matice \mathbf{Q} a horní trojúhelníkové matice \mathbf{R} , která má na diagonále vlastní čísla matice \mathbf{A} .

Nechť je dána matice $\mathbf{A} \in \mathbb{R}^{n,n}$, pro jejíž vlastní čísla platí

$$|\lambda_1| > |\lambda_2| > \dots > |\lambda_n|.$$

Označme $\mathbf{A} = \mathbf{A}_0 = \mathbf{Q}_0\mathbf{R}_0$. Dále položíme $\mathbf{A}_1 = \mathbf{R}_0\mathbf{Q}_0$ a definujme posloupnost matic

$$\mathbf{A}_{k-1} = \mathbf{Q}_{k-1}\mathbf{R}_{k-1}, \quad k = 1, 2, \dots,$$

kde \mathbf{Q}_{k-1} je ortogonální matice a \mathbf{R}_{k-1} horní trojúhelníková matice taková, že

$\mathbf{A}_k = \mathbf{R}_{k-1}\mathbf{Q}_{k-1}$, pro $k = 1, 2, \dots$. Všechny matice \mathbf{A}_k jsou podobné, protože platí

$$\mathbf{A}_k = \mathbf{R}_{k-1}\mathbf{Q}_{k-1} = \mathbf{Q}_{k-1}^T(\mathbf{Q}_{k-1}\mathbf{R}_{k-1})\mathbf{Q}_{k-1} = \mathbf{Q}_{k-1}^T\mathbf{A}_{k-1}\mathbf{Q}_{k-1}.$$

Indukcí dostaneme

$$\mathbf{A}_k = (\mathbf{Q}_0\mathbf{Q}_1 \dots \mathbf{Q}_{k-1})^T \mathbf{A} (\mathbf{Q}_0\mathbf{Q}_1 \dots \mathbf{Q}_{k-1}), \text{ pro } k = 1, 2, \dots \quad (22)$$

Lze dokázat (viz [5]), že posloupnost $\{\mathbf{A}_k\}$ ze vztahu (22) téměř vždy konverguje k Schurovu rozkladu matice \mathbf{A} . Pro tento případ označme $\mathbf{Q} = \lim_{k \rightarrow \infty} \mathbf{Q}_0\mathbf{Q}_1 \dots \mathbf{Q}_{k-1}$ a $\mathbf{R} = \lim_{k \rightarrow \infty} \mathbf{A}_k$, potom dostáváme QR rozklad matice \mathbf{A} , tj.

$$\mathbf{A} = \mathbf{Q}\mathbf{R},$$

kde \mathbf{Q} je ortogonální matice a \mathbf{R} je horní trojúhelníková matice s vlastními čísly matice \mathbf{A} na diagonále. Pro zastavení výpočtu QR rozkladu matice \mathbf{A} využijeme poznatek, že jestliže $\lim_{k \rightarrow \infty} \mathbf{A}_k$ konverguje, pak konverguje k Schurovu rozkladu matice \mathbf{A} .

Výše popsanou metodu QR transformace přehledně shrneme v následujícím algoritmu.

ALGORITMUS 2.6

Nechť je dána matice $\mathbf{A} \in \mathbb{R}^{n,n}$ a požadovaná přesnost výpočtu ε . Algoritmus počítá QR rozklad matice \mathbf{A} , kde \mathbf{Q} je ortogonální matice, jejíž sloupce jsou tvořeny vlastními vektory matice \mathbf{A} a \mathbf{R} je horní trojúhelníková matice s vlastními čísly matice \mathbf{A} na diagonále.

1. užitím Householderovy transformace rozložíme matici $\mathbf{A}_0 = \mathbf{A}$ na součin ortogonální matice \mathbf{Q}_0 a horní trojúhelníkové matice \mathbf{R}_0 , položíme $\mathbf{Q} = \mathbf{Q}_0$ a $k = 1$
2. položíme $\mathbf{A}_k = \mathbf{R}_{k-1}\mathbf{Q}_{k-1}$, kde $\mathbf{A}_k = (a_{ij}^k)_{i,j=1}^n$ a užitím Householderovy transformace provedeme rozklad $\mathbf{A}_k = \mathbf{Q}_k\mathbf{R}_k$

3. vypočítáme $\mathbf{Q} = \mathbf{Q}\mathbf{Q}_k$
4. jestliže platí $|a_{ij}^k| > \varepsilon$, kde $j = 1, \dots, n-1$, $i = j+1, \dots, n$, položíme $k = k+1$ a vrátíme se do kroku 2, v opačném případě položíme $\mathbf{R} = \mathbf{A}_k$ a výpočet ukončíme

Na výpočet vlastních čísel a vlastních vektorů matice \mathbf{A} , užitím QR transformace, jsem vytvořila m-file (viz příloha procedura QRrozklad.m), který využijeme také v následujícím příkladě.

Příklad 2.13 Pomocí QR transformace určíme vlastní čísla a vlastní vektory matice

$$\mathbf{A} = \begin{pmatrix} 3 & 4 & -1 \\ 2 & 0 & 5 \\ 1 & -2 & 6 \end{pmatrix}.$$

Užitím procedury QRrozklad.m dostaneme (s požadovanou přesností výpočtu $\varepsilon = 10^{-5}$) po 29 iteracích matici \mathbf{R} , která má na diagonále vlastní čísla matice \mathbf{A} . Můžeme tedy psát $\mathbf{R} = \text{diag}(5.1451, 3.5240, 0.3309)$ a matici

$$\mathbf{Q} = \begin{pmatrix} 0.8492 & 0.4516 & -0.2738 \\ 0.4988 & -0.5154 & 0.6968 \\ 0.1736 & -0.7283 & -0.6629 \end{pmatrix},$$

jejíž sloupce jsou tvořeny vlastními vektory matice \mathbf{A} .

Poznámka 2.5 Matici \mathbf{A} , jejíž vlastní čísla jsou reálná, výše zmíněný algoritmus převede bez problémů na matici v horním trojúhelníkovém tvaru s vlastními čísly matice \mathbf{A} na diagonále. Současně je k dispozici matice \mathbf{Q} , jejíž sloupce jsou tvořeny vlastními vektory matice \mathbf{A} . Jelikož ve výpočtu uvažujeme pouze ortogonální transformace, v případě komplexních vlastních čísel se může vyskytnout problém ten, že matice \mathbf{A}_k konverguje k blokové horní trojúhelníkové matici s diagonálními bloky typu 1x1 nebo 2x2. Z tohoto důvodu nebude splněno ukončovací kritérium, které požaduje, aby byly prvky matice \mathbf{A}_k pod diagonálou nulové.

Proto byl počet kroků omezen na hodnotu 100. Pokud výpočet v průběhu daného počtu iterací nedosáhne uspokojivého výsledku, je použito náhradní řešení, jehož výsledkem jsou vlastní čísla matice \mathbf{A} . Tento případ si názorně ukážeme v následujícím příkladu.

Příklad 2.14 Pomocí QR transformace určíme vlastní čísla matice

$$\mathbf{A} = \begin{pmatrix} 1 & 0 & 3 & -4 \\ 1 & 5 & 6 & -2 \\ -3 & 4 & 0 & 1 \\ 2 & 0 & 7 & -1 \end{pmatrix}$$

z příkladu 2.9.

Užitím procedury QRrozklad.m, pro požadovanou přesnost výpočtu $\varepsilon = 10^{-5}$, nezískáme ani po 100 iteracích horní trojúhelníkovou matici, ale pouze blokovou horní trojúhelníkovou matici

$$\mathbf{A}_k = \begin{pmatrix} 7.6018 & 2.8390 & -4.7549 & -1.5823 \\ 0 & -1.2680 & 7.3653 & 0.6786 \\ 0 & -1.5969 & -3.6868 & 1.7453 \\ 0 & 0 & 0 & -2.3530 \end{pmatrix}.$$

Proto je výpočet v této fázi zastaven a je aplikováno náhradní řešení integrované v proceduře QRrozklad.m. Ze získané blokové horní trojúhelníkové matice \mathbf{A}_k spočítáme, užitím zmíněného náhradního řešení, sloupcový vektor

$$\text{vlcA} = \begin{pmatrix} 7.6018 \\ -2.4774 + 3.2093i \\ -2.4774 - 3.2093i \\ 2.3530 \end{pmatrix},$$

jež je tvořen vlastními čísly matice \mathbf{A} .

Krok 2 v algoritmu 2.6, který se neustále opakuje, je náročný na počet operací. Proto metoda QR transformace představuje nákladný způsob, jak vypočítat Schurův rozklad. Naštěstí mohou být tyto problémy překonány, jak ukážeme v následující kapitole.

2.5 Hessenbergovy a reálné Schurovy tvary matice

V této kapitole ukážeme, jak lze metodu QR transformace z kapitoly 2.4 upravit na rychlou a efektivní metodu pro výpočet Schurova rozkladu.

Připomeňme, že $\mathbf{Q}^H \mathbf{A} \mathbf{Q} = \mathbf{T}$, kde $\mathbf{T} = \text{diag}(\lambda_1, \dots, \lambda_n) + \mathbf{N}$, je Schurův rozklad matice $\mathbf{A} \in \mathbb{C}^{n,n}$ ze vztahu (2). Tedy ze Schurova rozkladu lze snadno určit vlastní čísla a vlastní vektory matice \mathbf{A} .

Obdobně, jako v předchozí kapitole, budeme uvažovat pouze reálné matice.

V obecném případě Schurův rozklad teoreticky zabezpečuje existenci komplexní horní trojúhelníkové matice s vlastními čísly matice \mathbf{A} na diagonále. V případě reálné matice s komplexními vlastními čísly by to znamenalo hledat komplexní podobnostní transformace. Z tohoto důvodu se spokojíme s výpočtem alternativního rozkladu, známého jako reálný Schurův rozklad.

2.5.1 Reálný Schurův rozklad

Bloková horní trojúhelníková matice s diagonálními bloky typu 1x1, nebo 2x2 se nazývá **horní kvazi-trojúhelníková matice**. Reálným Schurovým rozkladem se myslí rozklad na horní kvazi-trojúhelníkový tvar.

Věta 2.6 (Reálný Schurův rozklad) *Nechť $\mathbf{A} \in \mathbb{R}^{n,n}$, potom existuje ortogonální matice $\mathbf{Q} \in \mathbb{R}^{n,n}$ taková, že platí*

$$\mathbf{Q}^T \mathbf{A} \mathbf{Q} = \begin{pmatrix} \mathbf{R}_{11} & \mathbf{R}_{12} & \cdots & \mathbf{R}_{1k} \\ \mathbf{0} & \mathbf{R}_{22} & \cdots & \mathbf{R}_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{R}_{kk} \end{pmatrix}, \quad (23)$$

kde každá matice \mathbf{R}_{jj} , $j = 1, \dots, k$, je prvního nebo druhého řádu, přičemž $\mathbf{R}_{jj} = r_{jj} = \lambda_j$ pro reálné vlastní číslo nebo

$$\mathbf{R}_{jj} = \begin{pmatrix} r_{jj} & r_{j,j+1} \\ r_{j+1,j} & r_{j+1,j+1} \end{pmatrix} = \begin{pmatrix} \lambda & \mu \\ -\mu & \lambda \end{pmatrix},$$

kde $\lambda_{j,j+1} = \lambda \pm i\mu$, pro komplexně sdružená vlastní čísla matice \mathbf{A} .

Důkaz: viz [5] str. 341.

Věta nám říká, že libovolná reálná matice je ortogonálně podobná horní kvazitrojúhelníkové matici. Je zřejmé, že reálnou a imaginární část komplexních vlastních čísel lze snadno získat z blokově diagonální matice typu 2×2 .

2.5.2 Hessenbergova redukce

Nejdřív převedeme matici \mathbf{A} na horní Hessenbergův tvar. Výpočet počátečního Hessenbergova tvaru je velmi důležitý a může být počítán jako součin Householderových matic $\mathbf{P}_1, \dots, \mathbf{P}_{n-2}$. Úlohou matice \mathbf{P}_k , kde $k = 1, \dots, n-2$, je nulovat k -tý sloupec matice \mathbf{A} pod subdiagonálou. Např. pro $n = 6$ máme

$$\begin{pmatrix} \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times \end{pmatrix} \xrightarrow{\mathbf{P}_1} \begin{pmatrix} \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times \\ \mathbf{0} & \times & \times & \times & \times & \times \\ \mathbf{0} & \times & \times & \times & \times & \times \\ \mathbf{0} & \times & \times & \times & \times & \times \\ \mathbf{0} & \times & \times & \times & \times & \times \end{pmatrix} \xrightarrow{\mathbf{P}_2} \begin{pmatrix} \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times \\ \mathbf{0} & \times & \times & \times & \times & \times \\ \mathbf{0} & \mathbf{0} & \times & \times & \times & \times \\ \mathbf{0} & \mathbf{0} & \times & \times & \times & \times \\ \mathbf{0} & \mathbf{0} & \times & \times & \times & \times \end{pmatrix} \xrightarrow{\mathbf{P}_3} \\ \begin{pmatrix} \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times \\ \mathbf{0} & \times & \times & \times & \times & \times \\ \mathbf{0} & \mathbf{0} & \times & \times & \times & \times \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \times & \times & \times \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \times & \times & \times \end{pmatrix} \xrightarrow{\mathbf{P}_4} \begin{pmatrix} \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times \\ \mathbf{0} & \times & \times & \times & \times & \times \\ \mathbf{0} & \mathbf{0} & \times & \times & \times & \times \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \times & \times & \times \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \times & \times \end{pmatrix}.$$

Nechť je matice $\mathbf{A} \in \mathbb{R}^{n,n}$ ve tvaru

$$\mathbf{A} = \begin{pmatrix} a_{11} & \hat{\mathbf{a}}_1^T \\ \mathbf{a}_1 & \mathbf{A}_{22} \end{pmatrix}, \quad \text{kde } \mathbf{a}_1, \hat{\mathbf{a}}_1 \in \mathbb{R}^{n-1}, \mathbf{A}_{22} \in \mathbb{R}^{n-1, n-1}.$$

V 1. kroku iterace zvolíme Householderovu matici

$$\mathbf{P}_1 = \begin{pmatrix} 1 & \mathbf{o}^T \\ \mathbf{o} & \mathbf{P}_{11} \end{pmatrix} \quad \text{tak, že } \mathbf{P}_{11}\mathbf{a}_1 \text{ je násobkem } \mathbf{e}_1,$$

kde $\mathbf{P}_{11} \in \mathbb{R}^{n-1, n-1}$, $\mathbf{e} \in \mathbb{R}^{n-1}$. Potom matice

$$\mathbf{A}_1 = \mathbf{P}_1 \mathbf{A} \mathbf{P}_1^T = \begin{pmatrix} a_{11} & \hat{\mathbf{a}}_1^T \mathbf{P}_{11}^T \\ \mathbf{P}_{11} \mathbf{a}_{11} & \mathbf{P}_{11} \mathbf{A}_{22} \mathbf{P}_{11}^T \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12}^{(1)} & \hat{\mathbf{a}}_1^{(1)T} \\ a_{21}^{(1)} & a_{22}^{(1)} & \hat{\mathbf{a}}_2^{(1)T} \\ \mathbf{o} & \mathbf{a}_2^{(1)} & \mathbf{A}_{33}^{(1)} \end{pmatrix},$$

kde $a_{12}^{(1)}, a_{21}^{(1)}, a_{22}^{(1)} \in \mathbb{R}$, $\mathbf{a}_2^{(1)}, \hat{\mathbf{a}}_1^{(1)}, \hat{\mathbf{a}}_2^{(1)} \in \mathbb{R}^{n-2}$, $\mathbf{A}_{33}^{(1)} \in \mathbb{R}^{n-1, n-1}$, má první sloupec v Hessenbergově tvaru. Ve 2. kroku zvolíme matici

$$\mathbf{P}_2 = \begin{pmatrix} 1 & 0 & \mathbf{o}^T \\ 0 & 1 & \mathbf{o}^T \\ \mathbf{o} & \mathbf{o} & \mathbf{P}_{22} \end{pmatrix} \quad \text{tak, že } \mathbf{P}_{22} \mathbf{a}_2^{(1)} \text{ je násobkem } \mathbf{e}_1^{(2)},$$

kde $\mathbf{P}_{22} \in \mathbb{R}^{n-2, n-2}$, $\mathbf{e}_1^{(2)} \in \mathbb{R}^{n-2}$. Potom má matice $\mathbf{A}_2 = \mathbf{P}_2 \mathbf{A}_1 \mathbf{P}_2^T$ druhý sloupec v Hessenbergově tvaru. Po $n-2$ transformacích s Householderovými maticemi \mathbf{P}_k tvaru

$$\mathbf{P}_k = \begin{pmatrix} \mathbf{I}_k & 0 \\ 0 & \mathbf{P}_{kk} \end{pmatrix}, \quad \text{kde } \mathbf{I}_k \in \mathbb{R}^{k, k}, \mathbf{P}_{kk} \in \mathbb{R}^{n-k, n-k}$$

a $\mathbf{P}_{kk} \mathbf{a}_k^{(k-1)}$ je násobkem $\mathbf{e}_1^{(n-k)}$, pro $\mathbf{a}_k^{(k-1)}, \mathbf{e}_1^{(n-k)} \in \mathbb{R}^2$, dostáváme matici

$$\mathbf{H} \equiv \mathbf{P}_{n-2} \dots \mathbf{P}_1 \mathbf{A} \mathbf{P}_1^T \dots \mathbf{P}_{n-2}^T,$$

kteřá má $n-2$ sloupců v Hessenbergově tvaru, tj. sama je v horním Hessenbergově tvaru.

Výše popsaný výpočet matice v horním Hessenbergově tvaru shrneme v algoritmu.

ALGORITMUS 2.7

Nechť je dána matice $\mathbf{A} \in \mathbb{R}^{n, n}$. Algoritmus převede tuto matici, užitím Householderových matic $\mathbf{P}_1, \dots, \mathbf{P}_{n-2}$, na matici v horním Hessenbergově tvaru.

1. položíme $\mathbf{H} = \mathbf{A}$,
2. pro $k = 1, \dots, n-2$ vybereme z matice \mathbf{H} vektor $\mathbf{x} = \mathbf{H}(k+1 : n, k)$

(a) dle algoritmu 2.1 vypočteme vektor \mathbf{v}

(b) položíme

$$\mathbf{H}(k+1:n, k:n) = (\mathbf{I}_{n-k} - \beta \mathbf{v} \mathbf{v}^T) \mathbf{H}(k+1:n, k:n)$$

$$\mathbf{H}(1:n, k+1:n) = \mathbf{H}(1:n, k+1:n) (\mathbf{I}_{n-k} - \beta \mathbf{v} \mathbf{v}^T)$$

Podle algoritmu 2.7 jsem vytvořila jednoduchý m-file (viz příloha procedura HessHous.n), pomocí kterého snadno převedeme matici \mathbf{A} na Hessenbergův tvar.

Příklad 2.15 Převedte matici

$$\mathbf{A} = \begin{pmatrix} 2 & 3 & 1 & 0 \\ 1 & 2 & 1 & 1 \\ 3 & 1 & 1 & -2 \\ -4 & -2 & 1 & 6 \end{pmatrix}$$

na horní Hessenbergovu matici.

Užitím procedury HessHous.m dostaneme horní Hessenbergovu matici

$$\mathbf{H} = \begin{pmatrix} 2 & 1.1767 & -2.5343 & -1.4807 \\ 5.0990 & 4.9615 & 1.3448 & -3.5243 \\ 0 & 1.1429 & 2.9637 & -1.4410 \\ 0 & 0 & 0.9124 & 1.0747 \end{pmatrix}.$$

2.5.3 Hessenbergův QR rozklad

Nyní se podívejme na výpočet QR rozkladu horní Hessenbergovy matice za použití Givensových matic rotace.

Nechť $\mathbf{H} \in \mathbb{R}^{n,n}$ je horní Hessenbergova matice. Definujme transformaci

$$\mathbf{H}_i = \mathbf{G}_i \mathbf{H}_{i-1}, i = 1, \dots, n-1,$$

kde $\mathbf{H}_0 = \mathbf{H}$ a $\mathbf{G}_i = \mathbf{G}(i, i+1, \theta_i)$ jsou Givensovy matice rotace. Podstatu Givensovy transformace lze vysvětlit na malém příkladu. Předpokládejme, že

$n = 6$ a po dvou krocích jsme vypočítali

$$\mathbf{G}(2, 3, \theta_2)^T \mathbf{G}(1, 2, \theta_1)^T \mathbf{A} = \begin{pmatrix} \times & \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & 0 & \times & \times \end{pmatrix},$$

tj. vynulovali jsme prvky na pozicích $(2, 1)$ a $(3, 2)$.

Pomocí Givensovy matice rotace $\mathbf{G}(3, 4, \theta_3)$ získáme nulu na pozici $(4, 3)$ a dostaneme

$$\mathbf{G}(3, 4, \theta_3)^T \mathbf{G}(2, 3, \theta_2)^T \mathbf{G}(1, 2, \theta_1)^T \mathbf{A} = \begin{pmatrix} \times & \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & 0 & \times & \times \end{pmatrix}.$$

Takto lze pokračovat induktivně a po $n - 1$ opakováních dostaneme horní trojúhelníkovou matici $\mathbf{R} = \mathbf{G}_{n-1}^T \dots \mathbf{G}_1^T \mathbf{H}$ a volbou $\mathbf{Q} = \mathbf{G}_1 \dots \mathbf{G}_{n-1}$ dostaneme ortogonální rozklad $\mathbf{H} = \mathbf{QR}$.

Výše popsaný postup výpočtu QR rozkladu Hessenbergovy matice shrneme v algoritmu.

ALGORITMUS 2.8

Nechť je dána horní Hessenbergova matice $\mathbf{H} \in \mathbb{R}^{n,n}$. Potom algoritmus počítá QR rozklad této matice, kde \mathbf{Q} je ortogonální matice a \mathbf{R} horní trojúhelníková matice takové, že $\mathbf{QR} = \mathbf{H}$.

1. položíme $\mathbf{R} = \mathbf{H}$ a sloupce matice \mathbf{R} označíme \mathbf{r}_i , pro $i = 1, \dots, n$
2. pro $j = 1, \dots, n - 1$
 - (a) z vektoru $\mathbf{r}_j = (r_{j1}, \dots, r_{jn})^T$ vybereme prvky $x_k = r_{jj}$ a $x_i = r_{j+1,j}$
 - (b) podle algoritmu 2.3 vypočítáme parametry c a s

$$(c) \text{ položíme } \mathbf{R}([j, j+1], j:n) = \begin{pmatrix} c & -s \\ s & c \end{pmatrix} \mathbf{R}([j, j+1], j:n)$$

$$(d) \text{ položíme } \mathbf{Q}(:, [j, j+1]) = \mathbf{Q}(:, [j, j+1]) \begin{pmatrix} c & s \\ -s & c \end{pmatrix}$$

Na základě výše popsaného algoritmu jsem sestavila m-file na QR rozklad horní Hessenbergovy matice \mathbf{H} (viz příloha procedura QRHess.m).

Příklad 2.16 Rozložme matici

$$\mathbf{H} = \begin{pmatrix} 2 & 1.1767 & -2.5343 & -1.4807 \\ 5.0990 & 4.9615 & 1.3448 & -3.5243 \\ 0 & 1.1429 & 2.9637 & -1.4410 \\ 0 & 0 & 0.9124 & 1.0747 \end{pmatrix},$$

jež jsme vypočetli v příkladu 2.14, na součin ortogonální matice \mathbf{Q} a horní trojúhelníkové matice \mathbf{R} .

Užitím procedury QRHess.m dostaneme

$$\mathbf{Q} = \begin{pmatrix} -0.3651 & 0.4944 & 0.5348 & 0.5799 \\ -0.9309 & -0.1939 & -0.2098 & -0.2274 \\ 0 & -0.8474 & 0.3600 & 0.3904 \\ 0 & 0 & -0.7351 & 0.6779 \end{pmatrix}$$

a

$$\mathbf{R} = \begin{pmatrix} -5.4772 & -5.0486 & -0.3265 & 3.8216 \\ 0 & -1.3488 & -4.0250 & 1.1724 \\ 0 & 0 & -1.2412 & -1.3614 \\ 0 & 0 & 0 & 0.1091 \end{pmatrix}.$$

Snadno lze ověřit, že platí $\mathbf{H} = \mathbf{QR}$.

2.5.4 QR transformace z matice v horním Hessenbergově tvaru

Nyní si ukážeme, jak lze metodu QR transformace z kapitoly 2.4 upravit na rychlou a efektivní metodu pro výpočet Schurova rozkladu.

V podkapitole 2.5.2 jsme ukázali, jak převedeme matici $\mathbf{A} \in \mathbb{R}^{n,n}$ na matici $\mathbf{H} \in \mathbb{R}^{n,n}$, která je v horním Hessenbergově tvaru. Pro tuto matici sestrojíme posloupnost matic $\{\mathbf{H}_k\}$ následujícím způsobem.

Označíme $\mathbf{H} = \mathbf{H}_0 = \mathbf{Q}_0\mathbf{R}_0$. Dále položíme

$$\mathbf{H}_1 = \mathbf{R}_0\mathbf{Q}_0.$$

Tuto matici rozložíme na součin ortogonální a horní trojúhelníkové matice, tj. $\mathbf{H}_1 = \mathbf{Q}_1\mathbf{R}_1$. Položíme

$$\mathbf{H}_2 = \mathbf{R}_1\mathbf{Q}_1$$

a rozložíme $\mathbf{H}_2 = \mathbf{Q}_2\mathbf{R}_2$. Postupně analogicky dostáváme posloupnost matic

$$\mathbf{H}_k = \mathbf{R}_{k-1}\mathbf{Q}_{k-1}, \quad k = 1, 2, \dots,$$

kde $\mathbf{H}_{k-1} = \mathbf{Q}_{k-1}\mathbf{R}_{k-1}$.

Aby byla metoda QR transformace z matice v horním Hessenbergově tvaru opravdu efektivní, musí být také matice, která vznikne součinem matic $\mathbf{R}_k\mathbf{Q}_k$, pro $k = 0, 1, 2, \dots$, v horním Hessenbergově tvaru.

Označme $\mathbf{R} = \mathbf{R}_k$, a $\mathbf{Q} = \mathbf{Q}_k$. Potom nechť pro matici $\mathbf{H}_{nova} \in \mathbb{R}^{n,n}$ platí

$$\mathbf{H}_{nova} = \mathbf{R}\mathbf{Q},$$

kde $\mathbf{H} = \mathbf{Q}\mathbf{R}$ je QR rozklad matice \mathbf{H} . Dále nechť $\mathbf{G}_j = \mathbf{G}(j, j+1, \theta_j)$, $j = 1, \dots, n-1$, je j -tá Givensova matice rotace. Lze dokázat (viz. [5] str. 343), že matice $\mathbf{Q} = \mathbf{G}_1 \dots \mathbf{G}_{n-1}$ je horní Hessenbergova a tedy matice $\mathbf{R}\mathbf{Q} = \mathbf{H}_{nova}$ je taky horní Hessenbergova. Např. pro $n = 4$ použijeme tři Givensovy matice rotace a provedeme výpočet matice $\mathbf{H}_{nova} = \mathbf{R}\mathbf{Q}$ tak, že matici \mathbf{R} postupně násobíme zprava maticemi \mathbf{G}_1 , \mathbf{G}_2 a \mathbf{G}_3 , tj. postupně dostaneme

$$\begin{pmatrix} \times & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & 0 & \times & \times \\ 0 & 0 & 0 & \times \end{pmatrix} \rightarrow \begin{pmatrix} \times & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & 0 & \times & \times \\ 0 & 0 & \times & \times \end{pmatrix} \rightarrow \begin{pmatrix} \times & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & 0 & \times & \times \end{pmatrix} \rightarrow \begin{pmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & 0 & \times & \times \end{pmatrix}.$$

Výpočet matice \mathbf{H}_{nova} shrneme v algoritmu.

ALGORITMUS 2.9

Nechť $\mathbf{H} \in \mathbb{R}^{n,n}$ je horní Hessenbergova matice. Potom tento algoritmus přepíše matici \mathbf{H} na matici $\mathbf{H}_{nova} = \mathbf{RQ}$, kde $\mathbf{H} = \mathbf{QR}$ je QR rozklad matice \mathbf{H} .

1. pro $j = 1, \dots, n - 1$

(a) vybereme z matice \mathbf{H} prvky $x_i = \mathbf{H}(j, j)$ a $x_l = \mathbf{H}(j + 1, j)$

(b) podle algoritmu 2.3 vypočítáme parametry c a s

(c) položíme

$$\mathbf{H}([j, j + 1], j : n) = \begin{pmatrix} c_j & -s_j \\ s_j & c_j \end{pmatrix} \mathbf{H}([j, j + 1], j : n)$$

$$\mathbf{H}(1 : j + 1, [j, j + 1]) = \mathbf{H}(1 : j + 1, [j, j + 1]) \begin{pmatrix} c_j & s_j \\ -s_j & c_j \end{pmatrix}$$

Podle algoritmu 2.9 jsem vytvořila jednoduchý m-file (viz příloha procedura RQHess.m), pomocí kterého vypočítáme matici \mathbf{H}_{nova} , pro kterou platí $\mathbf{H}_{nova} = \mathbf{RQ}$.

Příklad 2.17 Přepišme matici

$$\mathbf{H} = \begin{pmatrix} 2 & 1.1767 & -2.5343 & -1.4807 \\ 5.0990 & 4.9615 & 1.3448 & -3.5243 \\ 0 & 1.1429 & 2.9637 & -1.4410 \\ 0 & 0 & 0.9124 & 1.0747 \end{pmatrix}$$

z příkladu 2.15 na matici \mathbf{H}_{nova} pro kterou bude platit $\mathbf{H}_{nova} = \mathbf{RQ}$, kde matice \mathbf{R} a \mathbf{Q} jsme vypočítali v příkladě 2.15.

Užitím procedury RQHess.m dostaneme

$$\mathbf{H}_{nova} = \begin{pmatrix} 6.7000 & -1.4521 & -4.7971 & 0.4355 \\ 1.2557 & 3.6721 & -2.0280 & -0.4696 \\ 0 & 1.0517 & 0.5540 & -1.4075 \\ 0 & 0 & -0.0802 & 0.0739 \end{pmatrix}.$$

Snadno lze ověřit, že platí $\mathbf{H}_{nova} = \mathbf{RQ}$.

Nyní si shrneme všechny poznatky o QR transformaci z matice v horním Hessenbergově tvaru do algoritmu.

ALGORITMUS 2.10

Nechť je dána matice $\mathbf{A} \in \mathbb{R}^{n,n}$ a požadovaná přesnost výpočtu ε . Algoritmus převede matici \mathbf{A} na matici v horním Hessenbergově tvaru \mathbf{H} . Z této matice pak počítá QR rozklad, kde matice \mathbf{Q} je ortogonální a \mathbf{R} je horní trojúhelníková matice s vlastními čísly matice \mathbf{A} na diagonále.

1. užitím algoritmu 2.7 převedeme matici \mathbf{A} na matici v horním Hessenbergově tvaru \mathbf{H}
2. položíme $\mathbf{H}_0 = \mathbf{H}$
3. užitím algoritmu 2.8 provedeme rozklad matice $\mathbf{H}_0 = \mathbf{Q}_0\mathbf{R}_0$, položíme $\mathbf{Q} = \mathbf{Q}_0$ a $k = 1$
4. pomocí algoritmu 2.9 položíme $\mathbf{H}_k = \mathbf{R}_{k-1}\mathbf{Q}_{k-1}$, kde $\mathbf{H}_k = (h_{ij}^k)_{i,j=1}^n$ a užitím algoritmu 2.8 provedeme rozklad $\mathbf{H}_k = \mathbf{Q}_k\mathbf{R}_k$
5. vypočítáme $\mathbf{Q} = \mathbf{Q}\mathbf{Q}_k$
6. jestliže platí $|h_{i+1,i}^k| > \varepsilon$, $i = 1, \dots, n$, položíme $k = k + 1$ a vrátíme se do kroku 4, v opačném případě položíme $\mathbf{R} = \mathbf{H}_k$ a výpočet ukončíme

Podle algoritmu 2.10 jsem vytvořila jednoduchý m-file (viz příloha procedura QRHrozklad.m), pomocí kterého vypočítáme vlastní čísla a vlastní vektory matice \mathbf{A} .

Příklad 2.18 Pomocí QR transformace určíme vlastní čísla a vlastní vektory matice

$$\mathbf{A} = \begin{pmatrix} 2 & 3 & 1 \\ 0 & -1 & 5 \\ 6 & 8 & 9 \end{pmatrix}.$$

Užitím procedury QRHrozklad.m nejprve převedeme matici \mathbf{A} na matici v horním Hessenbergově tvaru, tj.

$$\mathbf{H} = \begin{pmatrix} 2 & 1 & 3 \\ 6 & 9 & 8 \\ 0 & 5 & -1 \end{pmatrix}$$

a dále vypočítáme (s požadovanou přesností výpočtu $\varepsilon = 10^{-5}$) po 12 iteracích matici \mathbf{R} , která má na diagonále vlastní čísla matice \mathbf{A} , tedy můžeme psát $\mathbf{R} = \text{diag}(12.9904, -3.0410, 0.0506)$ a matici

$$\mathbf{Q} = \begin{pmatrix} -0.1748 & 0.4125 & 0.8940 \\ -0.9272 & 0.2355 & -0.2905 \\ -0.3314 & -0.1797 & 0.3411 \end{pmatrix},$$

jejíž sloupce jsou tvořeny vlastními vektory matice \mathbf{A} .

Pro srovnání vypočítejme matice \mathbf{R} a \mathbf{Q} užitím procedury QRrozklad.m z kapitoly 2.4. S požadovanou přesností výpočtu $\varepsilon = 10^{-5}$ (po 5 iteracích) dostaneme

$$\mathbf{R} = \begin{pmatrix} 12.9991 & 1.7686 & -4.9067 \\ -0.0794 & -3.0498 & 3.9382 \\ 0 & 0 & 0.0506 \end{pmatrix} \quad \text{a} \quad \mathbf{Q} = \begin{pmatrix} 0.1728 & -0.4134 & -0.8940 \\ 0.3357 & 0.8780 & -0.3411 \\ 0.9260 & -0.2412 & 0.2905 \end{pmatrix}.$$

Poznámka 2.6 Jelikož ve výpočtu uvažujeme pouze ortogonální transformace, v případě komplexních vlastních čísel se může vyskytnout, podobně jako v předchozí kapitole, problém ten, že matice \mathbf{H}_k konverguje k horní kvazi-trojúhelníkové matici. Z tohoto důvodu nebude splněno ukončovací kritérium, které požaduje, aby byly prvky matice \mathbf{H}_k pod diagonálou nulové. Proto byl počet kroků omezen na hodnotu 100. Pokud výpočet v průběhu daného počtu iterací nedosáhne uspokojivého výsledku, je použito náhradní řešení, jehož výsledkem jsou vlastní čísla matice \mathbf{A} .

3 Vlastní čísla symetrické matice

Cílem této kapitoly je představit problém vlastních čísel symetrické matice. Stejně jako u nesymetrických matic si nejdříve ujasníme, jak jsou vlastní čísla citlivá na změny v matici. V druhé části si probereme Jacobiho metodu, která se v literatuře objevila jako jedna z prvních.

3.1 Vlastnosti a rozklady

V této kapitole uvedeme matematické postupy, které jsou nezbytné pro rozvoj a analýzu algoritmů pro problém vlastních čísel symetrické matice.

Symetrie zaručuje, že všechna vlastní čísla matice \mathbf{A} jsou reálná a že existuje ortogonální báze vlastních vektorů, tj. vlastní vektory jsou lineárně nezávislé.

Věta 3.1 (Symetrický Schurův rozklad) *Nechť $\mathbf{A} \in \mathbb{R}^{n,n}$ je symetrická matice. Potom existuje ortogonální matice \mathbf{Q} taková, že platí*

$$\mathbf{Q}^T \mathbf{A} \mathbf{Q} = \bar{\mathbf{A}},$$

kde $\bar{\mathbf{A}} = \text{diag}(\lambda_1, \dots, \lambda_n)$.

Důkaz: viz [5] str. 393.

Příklad 3.1 Najděme Schurův rozklad symetrické matice \mathbf{A} , jestliže

$$\mathbf{A} = \begin{pmatrix} 3 & 2 & 4 \\ 2 & 0 & 2 \\ 4 & 2 & 3 \end{pmatrix} \quad \text{a} \quad \mathbf{Q} = \begin{pmatrix} -0.4941 & -0.5580 & 0.6667 \\ -0.4720 & 0.8161 & 0.3333 \\ 0.7301 & 0.1500 & 0.6667 \end{pmatrix}.$$

Potom matice \mathbf{Q} je ortogonální a $\mathbf{Q}^T \mathbf{A} \mathbf{Q} = \text{diag}(-1, -1, 8)$.

Symbolem $\lambda_k(\mathbf{A})$ označme k -té největší vlastní číslo symetrické matice \mathbf{A} . Dále necht' platí

$$\lambda_1(\mathbf{A}) \geq \lambda_2(\mathbf{A}) \geq \dots \geq \lambda_n(\mathbf{A}).$$

Potom pro spektrální normu $\|\mathbf{A}\|_2$ matice \mathbf{A} , která má vlastní čísla $\lambda_k(\mathbf{A})$, $k = 1, \dots, n$, platí (viz [5] str. 394)

$$\|\mathbf{A}\|_2 = \max\{|\lambda_1(\mathbf{A})|, |\lambda_n(\mathbf{A})|\}.$$

Vlastní čísla symetrické matice lze odhadnout z podílu $\mathbf{x}^T \mathbf{A} \mathbf{x} / \mathbf{x}^T \mathbf{x}$, jak ukazuje následující věta.

Věta 3.2 (Courant-Fischerova věta) *Nechť $\mathbf{A} \in \mathbb{R}^{n,n}$ je symetrická matice, potom*

$$\lambda_k(\mathbf{A}) = \max_{\dim(S)=k} \min_{0 \neq \mathbf{y} \in S} \frac{\mathbf{y}^T \mathbf{A} \mathbf{y}}{\mathbf{y}^T \mathbf{y}},$$

pro $k = 1, \dots, n$, kde S je invariantní podprostor pro matici \mathbf{A} .

Důkaz: viz [5] str. 394.

3.1.1 Citlivost výpočtu vlastního čísla

Řešení problému vlastních čísel symetrické matice zahrnuje vytvoření posloupností ortogonálních transformací \mathbf{Q}_k s vlastností, že matice $\mathbf{Q}_k^T \mathbf{A} \mathbf{Q}_k$ jsou s rostoucím k postupně „více diagonální“. Přirozeně vzniká otázka, jak dobře se diagonální prvky matice $\mathbf{Q}_k^T \mathbf{A} \mathbf{Q}_k$ blíží vlastním číslům matice \mathbf{A} .

Věta 3.3 (Gershgorinova věta)

Nechť $\mathbf{A} \in \mathbb{R}^{n,n}$ je symetrická matice a $\mathbf{Q} \in \mathbb{R}^{n,n}$ matice ortogonální. Jestliže platí $\mathbf{Q}^T \mathbf{A} \mathbf{Q} = \mathbf{D} + \mathbf{F}$, kde matice $\mathbf{D} = \text{diag}(d_1, \dots, d_n)$ a matice $\mathbf{F} = (f_{ij})_{i,j=1}^n$ má nulové diagonální prvky, potom pro spektrum matice \mathbf{A} platí

$$\lambda(\mathbf{A}) \subseteq \bigcup_{i=1}^n (d_i - r_i, d_i + r_i),$$

kde $r_i = \sum_{j=1}^n |f_{ij}|$, pro $i = 1, \dots, n$.

Důkaz: viz [5] str. 395.

Poznámka 3.1 Obdobně jako ve větě 2.1, zabývající se obecnými maticemi, je i ve větě 3.3, jež se zabývá symetrickými maticemi, více možností, jak zvolit matice \mathbf{D} a \mathbf{F} . V případě obecné matice (dle věty 2.1) platí, že vlastní čísla náležejí do kruhu daných vlastností. U symetrické matice (dle věty 3.3) se vlastní čísla zobrazí na reálnou osu, protože symetrie zaručuje, že všechna vlastní čísla matice \mathbf{A} jsou reálná.

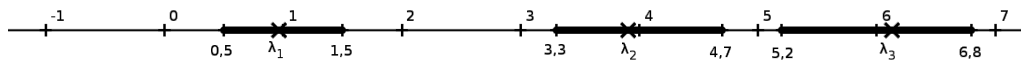
Příklad 3.2 Užitím věty 3.3 odhadněme polohu vlastních čísel symetrické matice

$$\mathbf{A} = \begin{pmatrix} 1 & 0.2 & 0.3 \\ 0.2 & 4 & -0.5 \\ 0.3 & -0.5 & 6 \end{pmatrix}.$$

Lze zvolit např.

$$\mathbf{D} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 6 \end{pmatrix} \quad \text{a} \quad \mathbf{F} = \begin{pmatrix} 0 & 0.2 & 0.3 \\ 0.2 & 0 & -0.5 \\ 0.3 & -0.5 & 0 \end{pmatrix}.$$

Dle vztahu $r_i = \sum_{j=1}^3 |f_{ij}|$ určíme $r_1 = |0.2| + |0.3| = 0.5$, $r_2 = |0.2| + |-0.5| = 0.7$ a $r_3 = |0.3| + |-0.5| = 0.8$. Potom, dle věty 3.3, budou Gershgorinovy intervaly ve tvaru $(0.5, 1.5)$, $(3.3, 4.7)$ a $(5.2, 6.8)$. Výpočtem kořenů charakteristického polynomu matice \mathbf{A} lze zjistit, že vlastní čísla jsou $\lambda_1 = 0.9644$, $\lambda_2 = 3.9057$ a $\lambda_3 = 6.1299$ (na obrázku jsou vyznačena znakem \times).



Další výsledky ukazují, že pokud je symetrická matice \mathbf{A} pozměněna o hodnoty symetrické matice \mathbf{E} , pak je součet druhých mocnin rozdílů vlastních čísel matic $\mathbf{A} + \mathbf{E}$ a \mathbf{A} maximálně roven hodnotě $\|\mathbf{E}\|_F^2$.

Věta 3.4 (Wielandt-Hoffman) *Nechť $\mathbf{A}, \mathbf{E} \in \mathbb{R}^{n,n}$ jsou symetrické matice.*

Potom platí

$$\sum_{i=1}^n (\lambda_i(\mathbf{A} + \mathbf{E}) - \lambda_i(\mathbf{A}))^2 \leq \|\mathbf{E}\|_F^2.$$

Důkaz: viz [5] str. 396.

Příklad 3.3 Nechť jsou dány matice

$$\mathbf{A} = \begin{pmatrix} 3 & 2 & 4 \\ 2 & 0 & 2 \\ 4 & 2 & 3 \end{pmatrix} \quad \text{a} \quad \mathbf{E} = \begin{pmatrix} 0.001 & 0.002 & 0.001 \\ 0.002 & 0.003 & 0.001 \\ 0.001 & 0.001 & 0 \end{pmatrix}.$$

Výpočtem kořenů charakteristického polynomu matice \mathbf{A} a matice $\mathbf{A} + \mathbf{E}$ zjistíme jejich spektra $\lambda(\mathbf{A}) = \{8, -1, -1\}$ a $\lambda(\mathbf{A} + \mathbf{E}) = \{8.003, -0.9984, -1.0006\}$.

Potom

$$\begin{aligned} \sum_{i=1}^3 (\lambda_i(\mathbf{A} + \mathbf{E}) - \lambda_i(\mathbf{A}))^2 &= (8.003 - 8)^2 + (-0.9984 - (-1))^2 + (-1.0006 - (-1))^2 \\ &= 1.192 \times 10^{-5} \end{aligned}$$

a

$$\|\mathbf{E}\|_F^2 = \sum_{i=1}^3 \sum_{j=1}^3 |a_{ij}|^2 = 2.2 \times 10^{-5},$$

tedy platí

$$1.192 \times 10^{-5} \leq 2.2 \times 10^{-5}.$$

Dle věty 3.2 lze odvodit další užitečné vlastnosti vlastních čísel.

Věta 3.5 *Nechť $\mathbf{A}, \mathbf{E} \in \mathbb{R}^{n,n}$ jsou symetrické matice. Potom*

$$\lambda_k(\mathbf{A}) + \lambda_n(\mathbf{E}) \leq \lambda_k(\mathbf{A} + \mathbf{E}) \leq \lambda_k(\mathbf{A}) + \lambda_1(\mathbf{E}),$$

pro $k = 1, \dots, n$.

Důkaz: viz [5] str. 396.

Příklad 3.4 Nechť jsou dány matice

$$\mathbf{A} = \begin{pmatrix} 3 & 2 & 4 \\ 2 & 0 & 2 \\ 4 & 2 & 3 \end{pmatrix} \quad \text{a} \quad \mathbf{E} = \begin{pmatrix} 0.001 & 0.002 & 0.001 \\ 0.002 & 0.003 & 0.001 \\ 0.001 & 0.001 & 0 \end{pmatrix}.$$

Ručním výpočtem vlastních čísel zjistíme, že $\lambda(\mathbf{A}) = \{8, -1, -1\}$, $\lambda(\mathbf{E}) = \{0.0046, 0, -0.0006\}$ a $\lambda(\mathbf{A} + \mathbf{E}) = \{8.003, -0.9984, -1.0006\}$. Potom

$$8 - 0.0006 \leq 8.003 \leq 8 + 0.0046$$

$$-1 - 0.0006 \leq -0.9984 \leq -1 + 0.0046$$

$$-1 - 0.0006 \leq -1.0006 \leq -1 + 0.0046.$$

Věta 3.6 (Vlastnost provázání) Nechť $\mathbf{A} \in \mathbb{R}^{n,n}$ je symetrická matice. Dále označme $\mathbf{A}_r = \mathbf{A}(1:r, 1:r)$, pro $r = 1, \dots, n-1$. Potom platí

$$\lambda_{r+1}(\mathbf{A}_{r+1}) \leq \lambda_r(\mathbf{A}_r) \leq \lambda_r(\mathbf{A}_{r+1}) \leq \dots \leq \lambda_2(\mathbf{A}_{r+1}) \leq \lambda_1(\mathbf{A}_r) \leq \lambda_1(\mathbf{A}_{r+1}).$$

Důkaz: viz [5] str. 397.

Příklad 3.5 Nechť je dána symetrická matice

$$\mathbf{A} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 3 & 5 & 7 \\ 1 & 5 & 9 & 11 \\ 1 & 7 & 11 & 13 \end{pmatrix},$$

kde

$$\mathbf{A}_1 = (1), \quad \mathbf{A}_2 = \begin{pmatrix} 1 & 1 \\ 1 & 3 \end{pmatrix}, \quad \mathbf{A}_3 = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 3 & 5 \\ 1 & 5 & 9 \end{pmatrix} \quad \text{a} \quad \mathbf{A}_4 = \mathbf{A}.$$

Ručním výpočtem vlastních čísel těchto matic zjistíme, že

$$\lambda(\mathbf{A}_1) = \{1\}, \quad \lambda(\mathbf{A}_2) = \{3.4142, 0.5858\}, \quad \lambda(\mathbf{A}_3) = \{12, 1, 0\}$$

a

$$\lambda(\mathbf{A}_4) = \{25.5727, 1.0313, 0.3261, -0.9301\}.$$

Tedy vidíme, že pro $r = 1$ platí

$$0.5858 < 1 < 3.4142,$$

pro $r = 2$

$$0 < 0.5858 < 1 < 3.4142 < 12.$$

Obdobně to můžeme ověřit pro $r = 3$ a $r = 4$.

3.1.2 Citlivost výpočtu invariantního podprostoru

Mnoho metod umožňujících výpočet vlastních čísel pracuje tak, že původní problém rozdělí na menší podproblémy. Tato myšlenka je základem pro tuto kapitolu.

Věta 3.7 *Nechť $\mathbf{A} \in \mathbb{R}^{n,n}$ je symetrická matice a*

$$\mathbf{Q} = \begin{pmatrix} \mathbf{Q}_1 & \mathbf{Q}_2 \end{pmatrix},$$

je ortogonální matice, kde $\mathbf{Q}_1 \in \mathbb{R}^{n,r}$ a $\mathbf{Q}_2 \in \mathbb{R}^{n,n-r}$, pro $1 \leq r < n$. Dále nechť $\text{ran}(\mathbf{Q}_1)$ je invariantní podprostor, potom platí

$$\mathbf{Q}^T \mathbf{A} \mathbf{Q} = \mathbf{D} = \begin{pmatrix} \mathbf{D}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_2 \end{pmatrix}, \quad (24)$$

kde $\mathbf{D}_1 \in \mathbb{R}^{r,r}$, $\mathbf{D}_2 \in \mathbb{R}^{n-r,n-r}$ a $\lambda(\mathbf{A}) = \lambda(\mathbf{D}_1) \cup \lambda(\mathbf{D}_2)$.

Důkaz: viz [5] str. 398.

Citlivost výpočtu invariantního podprostoru závisí na oddělení vlastních čísel, odpovídajících příslušné množině vlastních vektorů, od zbytku spektra. Definujme vzdálenost mezi symetrickými maticemi \mathbf{D}_1 a \mathbf{D}_2 vztahem

$$\text{sep}(\mathbf{D}_1, \mathbf{D}_2) = \min_{\lambda \in \lambda(\mathbf{D}_1), \mu \in \lambda(\mathbf{D}_2)} |\lambda - \mu| \geq 0. \quad (25)$$

Rovnosti je dosaženo, jestliže $\lambda = \mu$. Nyní můžeme vyslovit větu.

Věta 3.8 *Nechť $\mathbf{A}, \mathbf{E} \in \mathbb{R}^{n,n}$ jsou symetrické matice a necht'*

$$\mathbf{Q} = (\mathbf{Q}_1 \mathbf{Q}_2),$$

kde $\mathbf{Q}_1 \in \mathbb{R}^{n,r}$ a $\mathbf{Q}_2 \in \mathbb{R}^{n,n-r}$, pro $1 \leq r < n$, je ortogonální matice taková, že $\text{ran}(\mathbf{Q}_1)$ je invariantní podprostor matice \mathbf{A} . Dále rozložme matice $\mathbf{Q}^T \mathbf{A} \mathbf{Q}$ a $\mathbf{Q}^T \mathbf{E} \mathbf{Q}$ takto

$$\mathbf{Q}^T \mathbf{A} \mathbf{Q} = \begin{pmatrix} \mathbf{D}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_2 \end{pmatrix} \quad \text{a} \quad \mathbf{Q}^T \mathbf{E} \mathbf{Q} = \begin{pmatrix} \mathbf{E}_{11} & \mathbf{E}_{21}^T \\ \mathbf{E}_{21} & \mathbf{E}_{22} \end{pmatrix},$$

kde $\mathbf{D}_1, \mathbf{E}_{11} \in \mathbb{R}^{r,r}$, $\mathbf{D}_2, \mathbf{E}_{22} \in \mathbb{R}^{n-r,n-r}$ a $\mathbf{E}_{21} \in \mathbb{R}^{n-r,r}$. Jestliže $\text{sep}(\mathbf{D}_1, \mathbf{D}_2) > 0$
a

$$\|\mathbf{E}\|_2 \leq \frac{\text{sep}(\mathbf{D}_1, \mathbf{D}_2)}{5},$$

potom existuje matice $\mathbf{P} \in \mathbb{R}^{n-r,r}$ taková, že platí

$$\|\mathbf{P}\|_2 \leq 4 \frac{\|\mathbf{E}_{21}\|_2}{\text{sep}(\mathbf{D}_1, \mathbf{D}_2)}$$

a sloupce matice $\hat{\mathbf{Q}}_1 = (\mathbf{Q}_1 + \mathbf{Q}_2 \mathbf{P})(\mathbf{I} + \mathbf{P}^T \mathbf{P})^{-1/2}$ tvoří ortogonální bázi invariantního podprostoru pro matici $\mathbf{A} + \mathbf{E}$.

Důkaz: viz [5] str. 399.

Důsledek 3.1 *Jestliže platí předpoklady ve větě 3.8, potom pro vzdálenost mezi podprostory platí*

$$\text{dist}(\text{ran}(\mathbf{Q}_1), \text{ran}(\hat{\mathbf{Q}}_1)) \leq 4 \frac{\|\mathbf{E}_{21}\|_2}{\text{sep}(\mathbf{D}_1, \mathbf{D}_2)}.$$

Důkaz: viz [5] str. 399.

Jak je uvedeno v literatuře [5] str. 399, z důsledku 3.1 se ukazuje, že převrácená hodnota $\text{sep}(\mathbf{D}_1, \mathbf{D}_2)$ může být považovaná za číslo podmíněnosti matice \mathbf{A} , které měří citlivost výpočtu invariantního podprostoru $\text{ran}(\mathbf{Q}_1)$.

Nyní se podívejme na citlivost výpočtu jednoho vlastního vektoru. Položíme $r = 1$ a na výše uvedené výsledky aplikujeme následující větu.

Věta 3.9 *Nechť jsou dány symetrické matice $\mathbf{A}, \mathbf{E} \in \mathbb{R}^{n,n}$ a ortogonální matice*

$$\mathbf{Q} = (\mathbf{q}_1 \ \mathbf{Q}_2),$$

kde $\mathbf{q}_1 \in \mathbb{R}^n$ je vlastní vektor matice \mathbf{A} a $\mathbf{Q}_2 \in \mathbb{R}^{n,n-1}$. Rozložme matice $\mathbf{Q}^T \mathbf{A} \mathbf{Q}$ a $\mathbf{Q}^T \mathbf{E} \mathbf{Q}$ následovně

$$\mathbf{Q}^T \mathbf{A} \mathbf{Q} = \begin{pmatrix} \lambda & \mathbf{o}^T \\ \mathbf{o} & \mathbf{D}_2 \end{pmatrix} \quad \text{a} \quad \mathbf{Q}^T \mathbf{E} \mathbf{Q} = \begin{pmatrix} \varepsilon & \mathbf{e}^T \\ \mathbf{e} & \mathbf{E}_{22} \end{pmatrix},$$

kde $\lambda, \varepsilon \in \mathbb{R}$, $\mathbf{D}_2, \mathbf{E}_{22} \in \mathbb{R}^{n-1,n-1}$ a $\mathbf{e} \in \mathbb{R}^{n-1}$.

Jestliže $d = \min_{\mu \in \lambda(\mathbf{D}_2)} |\lambda - \mu| > 0$ a

$$\|\mathbf{E}\|_2 \leq \frac{d}{4},$$

potom existuje vektor $\mathbf{p} \in \mathbb{R}^{n-1}$ takový, že platí

$$\|\mathbf{p}\|_2 \leq \frac{4}{d} \|\mathbf{e}\|_2$$

a $\hat{\mathbf{q}}_1 = (\mathbf{q}_1 + \mathbf{Q}_2 \mathbf{p}) / \sqrt{1 + \mathbf{p}^T \mathbf{p}}$ je vlastní vektor matice $\mathbf{A} + \mathbf{E}$. Navíc platí

$$\text{dist}(\text{span}(\mathbf{q}_1), \text{span}(\hat{\mathbf{q}}_1)) = \sqrt{1 - (\mathbf{q}_1^T \hat{\mathbf{q}}_1)^2} \leq \frac{4}{d} \|\mathbf{e}\|_2.$$

Důkaz: viz [5] str. 400.

Následující příklad jsem převzala z literatury [5] str. 400, abych zpřehlednila početní postup, vedoucí k výsledku, z důvodu jeho absence v literatuře.

Příklad 3.6 Nechť jsou dány matice

$$\mathbf{A} = \text{diag}(0.999, 1.001, 2) \quad \text{a} \quad \mathbf{E} = \begin{pmatrix} 0 & 0.01 & 0.01 \\ 0.01 & 0 & 0.01 \\ 0.01 & 0.01 & 0 \end{pmatrix}.$$

Je zřejmé, že $\lambda(\mathbf{A}) = \{0.999, 1.001, 2\}$. Výpočtem kořenů charakteristického polynomu matice $\mathbf{A} + \mathbf{E}$ zjistíme její spektrum $\lambda(\mathbf{A} + \mathbf{E}) = \{0.9899, 1.0098, 2.0002\}$. Dále, dle věty 3.9, rozložme matice $\mathbf{Q}^T \mathbf{A} \mathbf{Q}$ a $\mathbf{Q}^T \mathbf{E} \mathbf{Q}$, tj.

$$\mathbf{Q}^T \mathbf{A} \mathbf{Q} = \begin{pmatrix} \lambda & \mathbf{o}^T \\ \mathbf{o} & \mathbf{D}_2 \end{pmatrix} \quad \text{a} \quad \mathbf{Q}^T \mathbf{E} \mathbf{Q} = \begin{pmatrix} \varepsilon & \mathbf{e}^T \\ \mathbf{e} & \mathbf{E}_{22} \end{pmatrix},$$

kde $\lambda = 0.999$, $\mathbf{D}_2 = \begin{pmatrix} 1.001 & 0 \\ 0 & 2 \end{pmatrix}$, $\varepsilon = 0$, $\mathbf{e} = \begin{pmatrix} 0.01 \\ 0.01 \end{pmatrix}$ a $\mathbf{E}_{22} = \begin{pmatrix} 0 & 0.01 \\ 0.01 & 0 \end{pmatrix}$.

Je zřejmé, že \mathbf{Q} je jednotková matice. Užitím Schurova rozkladu pro matici $\mathbf{A} + \mathbf{E}$ dostaneme

$$\hat{\mathbf{Q}}^T (\mathbf{A} + \mathbf{E}) \hat{\mathbf{Q}} = \text{diag}(0.9899, 1.0098, 2.002),$$

kde matice $\hat{\mathbf{Q}} = \begin{pmatrix} 0.7488 & 0.6706 & 0.0101 \\ -0.6706 & 0.7414 & 0.0101 \\ -0.0007 & -0.0143 & 0.9999 \end{pmatrix}$ je ortogonální. Výpočet uka-

zuje, že

$$\text{dist}(\text{span}(\mathbf{q}_1), \text{span}(\hat{\mathbf{q}}_1)) = \sqrt{1 - (\mathbf{q}_1^T \hat{\mathbf{q}}_1)^2} = 0.6706$$

a

$$\text{dist}(\text{span}(\mathbf{q}_2), \text{span}(\hat{\mathbf{q}}_2)) = \sqrt{1 - (\mathbf{q}_2^T \hat{\mathbf{q}}_2)^2} = 0.6707,$$

tj.

$$\text{dist}(\text{span}(\mathbf{q}_1), \text{span}(\hat{\mathbf{q}}_1)) = \text{dist}(\text{span}(\mathbf{q}_2), \text{span}(\hat{\mathbf{q}}_2)) = 0.67.$$

Tedy protože jsou vlastní vektory \mathbf{q}_1 a \mathbf{q}_2 spojené s blízkými vlastními čísly $\lambda_1(\mathbf{A}) = 0.999$ a $\lambda_2(\mathbf{A}) = 1.001$, nemohou být tyto vektory vypočítány přesně.

Na druhé straně vlastní čísla $\lambda_1(\mathbf{A})$, $\lambda_2(\mathbf{A})$ jsou dobře oddělená od vlastního čísla $\lambda_3(\mathbf{A})$, proto k nim odpovídající vlastní vektory \mathbf{q}_1 a \mathbf{q}_2 definují invariantní podprostor, který není obzvlášť citlivý na výpočet, jak ukážeme dále.

Rozložme matice $\mathbf{Q}^T \mathbf{A} \mathbf{Q}$ a $\mathbf{Q}^T \mathbf{E} \mathbf{Q}$ dle věty 3.8 takto

$$\mathbf{Q}^T \mathbf{A} \mathbf{Q} = \begin{pmatrix} \mathbf{D}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_2 \end{pmatrix} \quad \text{a} \quad \mathbf{Q}^T \mathbf{E} \mathbf{Q} = \begin{pmatrix} \mathbf{E}_{11} & \mathbf{E}_{21}^T \\ \mathbf{E}_{21} & \mathbf{E}_{22} \end{pmatrix},$$

kde $\mathbf{D}_1 = \begin{pmatrix} 0.999 & 0 \\ 0 & 1.001 \end{pmatrix}$, $\mathbf{D}_2 = 2$, $\mathbf{E}_{11} = \begin{pmatrix} 0 & 0.01 \\ 0.01 & 0 \end{pmatrix}$, $\mathbf{E}_{21} = (0.01, 0.01)$

a $\mathbf{E}_{22} = 0$. Ze vztahu $\text{sep}(\mathbf{D}_1, \mathbf{D}_2) = \min_{\lambda \in \lambda(\mathbf{D}_1), \mu \in \lambda(\mathbf{D}_2)} |\lambda - \mu|$ vypočítáme vzdálenost

mezi maticemi \mathbf{D}_1 a \mathbf{D}_2 , tj.

$$\text{sep}(\mathbf{D}_1, \mathbf{D}_2) = \min\{|0.999 - 2|, |1.001 - 2|\} = 0.999.$$

Tedy máme

$$\text{dist}(\text{span}(\mathbf{q}_1, \mathbf{q}_2), \text{span}(\hat{\mathbf{q}}_1, \hat{\mathbf{q}}_2)) \leq \frac{4}{\text{sep}(\mathbf{D}_1, \mathbf{D}_2)} \|\mathbf{E}_{21}\|_2 = 0.0566.$$

3.2 Jacobiho metoda

Jacobiho metoda vytváří posloupnost ortogonálních podobnostních transformací $\{\mathbf{A}_k\}$ s vlastností, že s rostoucím k jsou matice $\mathbf{A}_k = (a_{ij}^k)_{i,j=1}^n$ „více diagonální“. Nakonec prvky mimo diagonálu jsou dost malé (tj. $|a_{ij}^k| \leq \varepsilon$ pro $i \neq j$, kde např. $\varepsilon = 10^{-5}$), aby byly prohlášeny za nulové.

3.2.1 Teoretická podstata Jacobiho metody

Z věty 3.1 víme, že existuje ortogonální matice \mathbf{Q} , pro kterou platí

$$\mathbf{Q}^T \mathbf{A} \mathbf{Q} = \bar{\mathbf{A}},$$

kde $\bar{\mathbf{A}}$ je spektrální matice matice \mathbf{A} a matice \mathbf{Q} je tvořena vlastními vektory matice \mathbf{A} . Dále pro libovolnou matici \mathbf{A} platí

$$\sum_{i=1}^n |a_{ii}|^2 \leq \sum_{i,j=1}^n |a_{ij}|^2 = \|\mathbf{A}\|_F^2.$$

Jak je uvedeno v literatuře [2], ortogonální transformace zachovává velikost normy a platí

$$\|\mathbf{Q}^T \mathbf{A} \mathbf{Q}\|_F = \|\mathbf{A}\|_F,$$

potom ortogonální transformace \mathbf{Q} minimalizuje součet čtverců nediagonálních prvků matice $\mathbf{Q}^T \mathbf{A} \mathbf{Q}$. Podstatou Jacobiho metody je tedy systematicky snižovat hodnotu

$$\text{off}(\mathbf{A}) = \sqrt{\sum_{i=1}^n \sum_{j=1, j \neq i}^n a_{ij}^2}, \quad (26)$$

tj. normu nediagonálních prvků matice \mathbf{A} . Matici \mathbf{Q} hledáme jako limitu nekonečného součinu ortogonálních matic \mathbf{Q}_k , pro $k = 1, 2, \dots$, (tj. $\lim_{k \rightarrow \infty} \mathbf{Q}_1 \dots \mathbf{Q}_k = \mathbf{Q}$).

Označme $\{\mathbf{A}_k\}$ posloupnost matic

$$\mathbf{A}_k = \mathbf{Q}_k^T \mathbf{A}_{k-1} \mathbf{Q}_k \equiv \mathbf{Q}_k^T \dots \mathbf{Q}_1^T \mathbf{A} \mathbf{Q}_1 \dots \mathbf{Q}_k, \quad k = 1, 2, \dots,$$

kde $\mathbf{A}_0 = \mathbf{A}$ a matice \mathbf{Q}_k vybíráme tak, abychom v každém kroku maximálně zmenšili součet čtverců nediagonálních prvků matice \mathbf{A}_k . Za \mathbf{Q}_k volíme matici rovinné rotace

$$\mathbf{J}(p, q, \theta) = \begin{pmatrix} 1 & \dots & 0 & \dots & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & & \vdots & & \vdots \\ 0 & \dots & c & \dots & s & \dots & 0 \\ \vdots & & \vdots & \ddots & \vdots & & \vdots \\ 0 & \dots & -s & \dots & c & \dots & 0 \\ \vdots & & \vdots & & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & \dots & 0 & \dots & 1 \end{pmatrix},$$

kde parametr c je na pozicích (p, p) a (q, q) a parametr s resp. $-s$ je na pozicích (p, q) resp. (q, p) .

Poznámka 3.2 Matice $\mathbf{J}(p, q, \theta)$ je totožná s Givensovou maticí $\mathbf{G}(p, q, \theta)$, která byla popsána u nesymetrických matic. Pro větší přehlednost nazýváme matici $\mathbf{J}(p, q, \theta)$ **Jacobiho matice rotace**.

Dále označme $\mathbf{B} = \mathbf{A}_k$, pro $k = 1, 2, \dots$. Matice $\mathbf{B} = \mathbf{J}^T(p, q, \theta) \mathbf{A} \mathbf{J}(p, q, \theta)$ se od matice \mathbf{A} liší jen v p -tém a q -tém řádku a sloupci a platí

$$\begin{aligned} b_{pp} &= c^2 a_{pp} + s^2 a_{qq} - 2csa_{pq} \\ b_{qq} &= s^2 a_{pp} + c^2 a_{qq} + 2csa_{pq} \\ b_{pq} &= b_{qp} = (c^2 - s^2)a_{pq} + cs(a_{pp} - a_{qq}) \\ b_{ip} &= b_{pi} = ca_{pi} - sa_{qi}, \quad i \neq p, q \\ b_{iq} &= b_{qi} = sa_{pi} + ca_{qi} \quad i \neq p, q, \end{aligned} \tag{27}$$

pro $i, p, q = \{1, \dots, n\}$, kde $(b_{pq})_{p,q=1}^n$ resp. $(a_{pq})_{p,q=1}^n$ jsou prvky matice \mathbf{B} resp. \mathbf{A} . Snadno lze ověřit, že $b_{pi}^2 + b_{qi}^2 = a_{pi}^2 + a_{qi}^2$ a tedy

$$b_{pp}^2 + b_{qq}^2 + 2b_{pq}^2 = a_{pp}^2 + a_{qq}^2 + 2a_{pq}^2.$$

Jestliže zvolíme úhel θ tak, aby $b_{pq} = 0$, tj.

$$\tan(2\theta) = \frac{2a_{pq}}{a_{qq} - a_{pp}} \quad (28)$$

(viz rovnice ze vztahu (27)), potom $b_{pp}^2 + b_{qq}^2 = a_{pp}^2 + a_{qq}^2 + 2a_{pq}^2$, tj. součet čtverců nediagonálních prvků se zmenší o veličinu $2a_{pq}^2$ a součet čtverců diagonálních prvků se o touž veličinu zvětší. V každém kroku Jacobiho metody tedy

1. zvolíme nenulový nediagonální prvek (tj. vybereme dvojici indexů (p, q) , pro které platí $1 \leq p < q \leq n$),
2. vypočítáme $c = \cos(\theta)$ a $s = \sin(\theta)$ tak, aby platilo, že matice

$$\begin{pmatrix} b_{pp} & b_{pq} \\ b_{qp} & b_{qq} \end{pmatrix} = \begin{pmatrix} c & s \\ -s & c \end{pmatrix}^T \begin{pmatrix} a_{pp} & a_{pq} \\ a_{qp} & a_{qq} \end{pmatrix} \begin{pmatrix} c & s \\ -s & c \end{pmatrix} \quad (29)$$

je diagonální, tj. chceme aby $b_{pq} = b_{qp} = (c^2 - s^2)a_{pq} + cs(a_{pp} - a_{qq}) = 0$,

3. vypočítáme $\mathbf{B} = \mathbf{J}^T \mathbf{A} \mathbf{J}$, kde $\mathbf{J} = \mathbf{J}(p, q, \theta)$, tj. vypočítáme prvky matice \mathbf{B} , které se liší od prvků matice \mathbf{A} .

Toto ukazuje možnost nulování všech nediagonálních prvků matice \mathbf{A} pomocí postupných podobnostních transformací Jacobiho maticemi rotace. Upozorníme, že nediagonální prvek, který jsme v některé fázi anulovali se může v pozdějších fázích stát opět nenulovým. Proto bude diagonální matice $\bar{\mathbf{A}}$ dosaženo jen po nekonečně mnoha transformacích.

Konvergence této metody se dá dokázat pro různé možnosti výběru nediagonálního prvku v každém kroku. Při některé volbě je důkaz dosti nesnadný, ale v případě, kdy se v každém kroku anuluje nediagonální prvek o největší absolutní hodnotě, je důkaz velmi názorný (viz např. [2] str. 116).

Než budeme diskutovat o tom, jak je třeba zvolit indexy dvojice (p, q) , podívejme se na skutečné výpočty související s (p, q) podproblémem.

3.2.2 Symetrický Schurův rozklad druhého řádu

Předpokládejme, že jsme ze vztahu (29) vynulovali prvek b_{pq} . To znamená, že

$$0 = b_{pq} = (c^2 - s^2)a_{pq} + cs(a_{pp} - a_{qq}). \quad (30)$$

Jestliže $a_{pq} = 0$, potom položíme $c = 1$ a $s = 0$. Jinak položíme

$$\frac{2a_{pq}}{a_{qq} - a_{pp}} = \tan(2\theta) = \frac{2 \tan(\theta)}{1 - \tan^2(\theta)} = \frac{2t}{1 - t^2}, \quad \text{kde } t = \tan(\theta).$$

Dále označme

$$\tau = \frac{a_{qq} - a_{pp}}{2a_{pq}} = \cotan(2\theta).$$

Řešením rovnice $t^2 + 2\tau t - 1 = 0$ dostaneme kořeny

$$t_{1,2} = -\tau \pm \sqrt{\tau^2 + 1} = \frac{1}{\tau \pm \sqrt{\tau^2 + 1}}.$$

Aby byl výpočet stabilní je vhodné zabránit odčítání dvou možných blízkých čísel, proto volíme menší z kořenů $t_{1,2}$, tj.

$$t = \frac{\text{sign}(\tau)}{|\tau| + \sqrt{\tau^2 + 1}}.$$

Potom

$$c = \cos(\theta) = \frac{1}{\sqrt{1 + t^2}}, \quad s = \sin(\theta) = tc.$$

Volba $c > 0$, tj. $|\theta| \leq \frac{\pi}{4}$ zaručuje konvergenci posloupnosti matic $\{\mathbf{A}_k\}$, pro $k = 1, 2, \dots$, ke konkrétní matici $\bar{\mathbf{A}}$ (viz [2] str. 118). Výše popsany postup výpočtu parametrů $c = \cos(\theta)$ a $s = \sin(\theta)$ shrneme v algoritmu.

ALGORITMUS 3.1

Nechť je dána symetrická matice $\mathbf{A} \in \mathbb{R}^{n,n}$ a celá čísla p, q , pro která platí $1 \leq p < q \leq n$.

1. jestliže $a_{pq} = 0$ položíme

$$c = 1, \quad s = 0,$$

2. jinak počítáme

$$\tau = \frac{a_{qq} - a_{pp}}{2a_{pq}},$$

(a) je-li $\tau \geq 0$ vypočítáme t ze vztahu $t = 1/(\tau + \sqrt{1 + \tau^2})$,

(b) v opačném případě počítáme t jako $t = -1/(-\tau + \sqrt{1 + \tau^2})$,

3. potom

$$c = 1/\sqrt{1 + t^2}, \quad s = tc$$

Poznámka 3.3 Protože počítač nezná absolutní nulu, v m-filech počítáme s jistou odchylkou ε , kde ε je malé kladné číslo, např. $\varepsilon = 10^{-5}$. Tedy pokud v algoritmu 3.1 je $|a_{pq}| \leq \varepsilon$, bereme $a_{pq} = 0$.

Příklad 3.7 Nechť je dána matice

$$\mathbf{A} = \begin{pmatrix} 2 & 1 \\ 1 & 3 \end{pmatrix}.$$

Pomocí Jacobiho matice rotace vynulujeme prvky mimo diagonálu.

Vybereme dvojici indexů $(1, 2)$. Tedy $|a_{12}| \neq 0$ a dle algoritmu 3.1 počítáme

$$\tau = \frac{a_{22} - a_{11}}{2a_{12}} = \frac{3 - 2}{2} = 0.5.$$

Z toho vidíme, že $\tau > 0$ tedy t vypočítáme jako

$$t = \frac{1}{\tau + \sqrt{1 + \tau^2}} = \frac{1}{0.5 + \sqrt{1 + 0.5^2}} = 0.618$$

a parametry c a s jsou

$$c = \frac{1}{\sqrt{1 + t^2}} = 0.8507, \quad s = tc = 0.5257.$$

Jacobiho matice rotace je ve tvaru

$$\mathbf{J}(1, 2, \theta) = \begin{pmatrix} c & s \\ -s & c \end{pmatrix} = \begin{pmatrix} 0.8507 & 0.5257 \\ -0.5257 & 0.8507 \end{pmatrix}.$$

Potom pro $\mathbf{B} = \mathbf{J}(1, 2, \theta)^T \mathbf{A} \mathbf{J}(1, 2, \theta)$ platí

$$\mathbf{B} = \begin{pmatrix} 0.8507 & -0.5257 \\ 0.5257 & 0.8507 \end{pmatrix} \begin{pmatrix} 2 & 1 \\ 1 & 3 \end{pmatrix} \begin{pmatrix} 0.8507 & 0.5257 \\ -0.5257 & 0.8507 \end{pmatrix} = \begin{pmatrix} 1.382 & 0 \\ 0 & 3.618 \end{pmatrix}.$$

Na základě algoritmu 3.1 jsem sestavila m-file (viz příloha procedura jacobi.m), který využijeme dále v m-filu pro výpočet vlastních čísel a vlastních vektorů užitím Jacobiho metody.

3.2.3 Klasická Jacobiho metoda

Jak jsme se zmínili výše, pokud je řešen (p, q) podproblém, jsou změněny pouze řádky a sloupce p a q symetrické matice $\mathbf{A} \in \mathbb{R}^{n,n}$. Jakmile použijeme algoritmus 3.1, který určuje rotace druhého řádu, pak transformace

$$\mathbf{A} \rightarrow \mathbf{J}(p, q, \theta)^T \mathbf{A} \mathbf{J}(p, q, \theta)$$

mohou být realizovány v 6n krocích, pokud je využívána symetrie.

Otázkou je, jak zvolit indexy p a q . Z hlediska maximalizace redukce $\text{off}(\mathbf{A})$ je rozumné volit dvojici indexů (p, q) tak, aby a_{pq}^2 bylo maximální. To je základ klasické Jacobiho metody.

ALGORITMUS 3.2

1. položíme $\mathbf{Q} = \mathbf{I}_n$, $\text{tol} = \varepsilon \|\mathbf{A}\|_F$, kde ε je požadovaná přesnost výpočtu, a $k = 0$
2. ze vztahu (26) vypočítáme $\text{off}(\mathbf{A})$
3. zvolíme dvojici indexů (p, q) tak, aby platilo $|a_{pq}| = \max_{i,j=1,\dots,n, i \neq j} |a_{ij}|$
4. pomocí algoritmu 3.1 vypočítáme parametry c a s
5. položíme

$$(a) \mathbf{A} = \mathbf{J}(p, q, \theta)^T \mathbf{A} \mathbf{J}(p, q, \theta)$$

$$(b) \mathbf{Q} = \mathbf{Q} \mathbf{J}(p, q, \theta)$$

6. jestliže platí $\text{off}(\mathbf{A}) < \text{tol}$, položíme $k = k + 1$ a vrátíme se do kroku 2, v opačném případě výpočet ukončíme

Na výpočet vlastních čísel a vlastních vektorů matice \mathbf{A} , užitím Jacobiho metody, jsem vytvořila m-file (viz příloha procedura JacMet.m), který využijeme také v následujícím příkladě.

Příklad 3.8 Pomocí klasické Jacobiho metody určíme vlastní čísla symetrické matice

$$\mathbf{A} = \begin{pmatrix} 1 & 2 & 3 \\ 2 & -2 & -1 \\ 3 & -1 & 3 \end{pmatrix}.$$

Užitím procedury JacMet.m dostaneme (s požadovanou přesností výpočtu $\varepsilon = 10^{-5}$) po 7 iteracích spektrální matici $\bar{\mathbf{A}} = \text{diag}(0.6539, -3.8362, 5.1823)$ a matici

$$\mathbf{Q} = \begin{pmatrix} 0.5962 & -0.5341 & 0.5994 \\ 0.6346 & 0.7708 & 0.0557 \\ -0.4918 & 0.3472 & 0.7985 \end{pmatrix},$$

jejíž sloupce jsou tvořeny vlastními vektory matice \mathbf{A} . Lze snadno ověřit, že $\bar{\mathbf{A}} = \mathbf{Q}^T \mathbf{A} \mathbf{Q}$.

Problém klasické Jacobiho metody je, že hledání maximálního nediagonálního prvku vyžaduje pro matice vysokého řádu řádově vyšší počet operací, než provedení jedné transformace $\mathbf{J}(p, q, \theta)^T \mathbf{A} \mathbf{J}(p, q, \theta)$. Abychom se tohoto hledání zbavili, je možné užití metody zvané *Jacobiho cyklická metoda*. Její podstata je v tom, že cyklicky nulujeme prvky na místech $(1, 2), (1, 3), \dots, (1, n), (2, 3), \dots, (2, n), \dots, \dots, (n-1, n)$. V tomto případě můžeme nulovat i velmi malé prvky, i když v matici ještě existují podstatně větší prvky. Tím se nadměrně snižuje rychlost konvergence. Kompromisem mezi klasickou Jacobiho metodou a cyklickou Jacobiho metodou je tzv. *Jacobiho metoda se závoramí*. U této metody stanovíme

hodnotu ε_i , tzv. hodnota závory. Pak cyklicky, stejně jako u cyklické Jacobiho metody, anulujeme jen ty prvky, které jsou větší než ε_i . Jestliže už neexistuje větší nediagonální prvek, hodnotu závory zmenšíme a postup opakujeme. Rychlost konvergence závisí od vhodného výběru závory. Závory mohou být dané na začátku výpočtu, jako posloupnost $\{\varepsilon_i\}$ klesající k nule, nebo je můžeme měnit v každém kroku. Vzhledem k rozsahu diplomové práce se výše zmíněnými metodami, pro volbu dvojice indexů (p, q) , podrobněji nezabývám.

Závěr

V této diplomové práci jsou popsány některé metody výpočtu vlastních čísel a vlastních vektorů matic. K jednotlivým metodám jsou v matematickém softwaru MATLAB sestaveny programy, které jsou demonstrovány při řešení příkladů.

Je zde zpracována problematika citlivosti výpočtu vlastních čísel. Tato je testována na maticích, pro které lze snadno spočítat vlastní čísla, aby byl výpočet přehledný a názorný. Také zde řešíme problematiku Householderovy redukce a Givensovy rotace, které jsou základem pro výpočet vlastních čísel nesymetrické matice.

Dále jsou zde popsány konkrétní metody k výpočtu vlastních čísel a vlastních vektorů matic. Jednotlivé metody jsou přehledně rozřazeny do samostatných kapitol. Jako první je zpracována mocninná metoda, která se zabývá výpočtem vlastního čísla matice \mathbf{A} , jež je největší v absolutní hodnotě mezi všemi vlastními čísly matice \mathbf{A} . Druhá v pořadí je metoda QR transformace. Tato metoda počítá všechna vlastní čísla matice \mathbf{A} . Poslední metodou, která se zabývá problematikou výpočtu vlastních čísel nesymetrické matice, je vylepšená metoda QR transformace, jež využívá k výpočtu vlastních čísel a vlastních vektorů matice \mathbf{A} matici v horním Hessenbergově tvaru.

Najdeme zde také Jacobiho metodu, která se zabývá výpočtem vlastních čísel symetrické matice. Tato metoda využívá k výpočtu vlastních čísel symetrické matice \mathbf{A} tzv. Jacobiho rotace.

V celé práci jsou veškeré mezivýpočty při řešení jednotlivých příkladů ponechány bez zaokrouhlení. Tímto je docíleno vyšší přesnosti výsledku. Zaokrouhlení je až konečný výsledek a to na 4 desetinná místa.

Přílohy

Procedura house.m

```
function [v,B]=house(x)
% vstupní proměnná: x...nenulový sloupcový vektor
% výstupní proměnné: v...Householderův vektor, B...koeficient, který uchovává
% hodnotu 2/(vT * v)
m=size(x);
n=length(x);
if n~=m(1)
    error('vstupni vektor x musi byt sloupcovy')
    return;
end
if x==0
    error('vstupni vektor x musi byt nenulovy')
    return;
end
s=(x(2:n)')*x(2:n);
v=[1;x(2:n)];
if s==0
    B=0;
else
    M=sqrt(x(1)^2+s);
    if x(1)<=10^(-5)
        v(1)=x(1)-M;
    else
        v(1)=-s/(x(1)+M);
    end
    B=(2*v(1)^2)/(s+v(1)^2);
    v=v/v(1);
end
```


Procedura QRHouse.m

```
function [Q,R]=QRHouse(A)
% vstupní proměnná: A...reálná matice
% výstupní proměnné: Q...ortogonální matice, R...horní trojúhelníková matice
n=size(A);
if length(n)~=2 | n(1)~=n(2)
    error('matice A musi byt ctvercova');
    return;
end
n=n(1);
R=A;
b(1:(n-1))=0;
for j=1:(n-1)
    [v,B]=house(R(j:n,j));
    I=eye(n-j+1);
    R(j:n,j:n)=(I-B*v*(v'))*(R(j:n,j:n));
    b(j)=B;
    if j<n
        A(j+1:n,j)=v(2:n-j+1);
    end
end
end
Q=eye(n);
for r=j:-1:1
    v(r:n)=[1;A(r+1:n,r)];
    C=eye(n-r+1);
    Q(r:n,r:n)=(C-b(r)*v(r:n)*(v(r:n)'))*Q(r:n,r:n);
end
end
```

Procedura givens.m

```
function [c,s]=givens(xk,xi)
% vstupní proměnné: xk,xi... prvky zadaného vektoru x
% výstupní proměnné: c,s... parametry c=cos(úhlu), s=sin(úhlu)
if abs(xi)<=10(-5)
    c=1; s=0;
else
    if abs(xi)>abs(xk)
        t=-xk/xi; s=1/((1+t2)(1/2)); c=s*t;
    else
        t=-xi/xk; c=1/((1+t2)(1/2)); s=c*t;
    end
end
end
```

Procedura QRGivens.m

```
function [Q,R]=QRGivens(A)
% vstupní proměnná: A...reálná matice
% výstupní proměnné: Q...ortogonální matice, R...horní trojúhelníková matice
n=size(A);
if length(n)~=2 | n(1)~=n(2)
    error('matice A musi byt ctvercova');
    return;
end
n=n(1);
R=A;
Q=eye(n);
for k=1:n-1
    for i=n:-1:k+1
        [c,s]=givens(R(k,k),R(i,k));
        R([k,i],k:n)=[c -s;s c]*R([k,i],k:n);
        Q(:, [k,i])=Q(:, [k,i])*[c s;-s c];
    end
end
end
```

Procedura MocMet.m

```
function [lambda1,x1,k]=MocMet(A,q0,e)
% vstupní proměnné: A...čtvercová matice, q0...libovolný nenulový vektor,
% e...požadovaná přesnost výpočtu
% výstupní proměnné: lambda1...aproximace vlastního čísla matice A, x1...vlastní
% vektor matice A příslušný vlastnímu číslu lambda1, k...počet iterací potřebných
% k dosažení požadované aproximace vlastního čísla lambda1
n=size(A);
m=size(q0);
if length(n)~=2 | n(1)~=n(2) | n(1)~=length(q0) |n(1)~=m(1)
    error('A musi byt ctvercova matice a q0 sloupcovy vektor se stejnym
        poctem prvku jako ma A radku')
    return;
end
lambda0=q0'*A*q0;
z1=A*q0;
if norm(z1,2)==0
    error('musime zvolit jiny nenulovy vstupni vektor q0')
    return;
end
qk=z1./norm(z1,2);
lambda1=qk'*A*qk;
k=2;
while abs((lambda1-lambda0)./lambda0)>e
    zk=A*qk;
    qk=zk./norm(zk,2);
    lambda0=lambda1;
    lambda1=qk'*A*qk;
    k=k+1;
end
x1=qk;
```

Procedura QRrozklad.m

```
function [Ak,Q,k,vlcA]=QRrozklad(A,e)
% vstupní proměnné: A...čtvercová matice, e...požadovaná přesnost výpočtu
% výstupní proměnné: Ak...horní trojúhelníková matice s vlastními čísly matice A
% na diagonále, Q...ortogonální matice, jejíž sloupce jsou vlastní vektory matice A,
% k...počet iterací potřebných k dosažení požadované aproximace vlastních čísel
% vlcA...sloupcový vektor, jehož prvky jsou vlastní čísla matice A (vypise se
% pouze v případě, že matice A má komplexní vlastní čísla)
n=size(A);
if length(n)~=2 | n(1)~=n(2)
    error('matice A musí být čtvercová')
    return;
end
n=n(1);
Ak=A;
Q=eye(n);
k=1;
vlcA=[];
for i=1:n-1
    for s=n:-1:i+1
        while abs(Ak(s,i))>e
            if k<100
                [Qk,R]=QRHouse(Ak);
                Ak=R*Qk;
                Q=Q*Qk;
                k=k+1;
            else
                warning('po 100 iteracích nedosažen uspokojivý
vysledek, použito náhradní řešení, jehož výsledkem jsou
vlastní čísla matice A zobrazena v proměnné vlcA')
```

```

vlcA(1:n)=0;
vlcA=vlcA';
m=1;
while m<n
    Bk=Ak(m:m+1,m:m+1);
    if abs(Bk(2,1))>10^-3
        vlcA(m)=(Bk(1,1)+Bk(2,2))./2+sqrt((Bk(1,1)+
        +Bk(2,2))^2-4*(Bk(1,1)*Bk(2,2)-Bk(1,2)*Bk(2,1)))./2;
        vlcA(m+1)=(Bk(1,1)+Bk(2,2))./2-sqrt((Bk(1,1)+
        +Bk(2,2))^2-4*(Bk(1,1)*Bk(2,2)-Bk(1,2)*Bk(2,1)))./2;
        m=m+2;
    else
        vlcA(m)=Bk(1,1);
        m=m+1;
    end
end
if m==n
    vlcA(m)=Ak(n,n);
end
Q=[]; k=[];
return;
end
end
end
end

```

Procedura HessHous.m

```
function [H]=HessHous(A)
% vstupní proměnná: A...reálná matice
% výstupní proměnná: H...horní Hessenbergova matice
n=size(A);
if length(n)~=2 | n(1)~=n(2)
    error('matice A musi byt ctvercova')
    return;
end
n=n(1);
P=ones(n);
for i=1:n
    for j=1:n
        if imag(A(i,j))~=0
            error('matice A musi byt realna')
            return;
        end
        if i>j
            if abs(A(i,j))==0
                P(i,j)=0;
                P(i,i)=0;
                P(j,j)=0;
                P(j,i)=0;
            end
        end
    end
end
end
if P==zeros(n)
    error('matice A je diagonalni resp. v hornim trojuhelnikovem tvaru')
    return;
```

```
end
H=A;
for k=1:n-2
    [v,B]=house(H(k+1:n,k));
    I=eye(n-k);
    H(k+1:n,k:n)=(I-B*v*v')*H(k+1:n,k:n);
    H(1:n,k+1:n)=H(1:n,k+1:n)*(I-B*v*v');
end
```


Procedura QRHess.m

```
function [Q,R]=QRHess(H)
% vstupní proměnná: H...horní Hessenbergova matice
% výstupní proměnné: Q...ortogonální matice, R...horní trojúhelníková matice
n=size(H);
if length(n)~=2 | n(1)~=n(2)
    error('matice H musi byt ctvercova')
    return;
end
n=n(1);
for k=1:n-2
    for i=k+2:n
        if abs(H(i,k))>10^(-5)
            error('matice H musi byt v hornim Hessenbergove tvaru')
            return;
        end
    end
end
R=H;
Q=eye(n);
for j=1:n-1
    [c,s]=givens(R(j,j),R(j+1,j));
    R([j,j+1],j:n)=[c -s;s c]*R([j,j+1],j:n);
    Q(:,[j,j+1])=Q(:,[j,j+1])*[c s;-s c];
end
```

Procedura RQHess.m

```
function [H]=RQHess(H)
% vstupní proměnná: H...horní Hessenbergova matice
% výstupní proměnná: H...horní Hessenbergova matice
n=size(H);
if length(n)~=2 | n(1)~=n(2)
    error('matice H musi byt ctvercova')
    return;
end
n=n(1);
for j=1:n-2
    for i=j+2:n
        if abs(H(i,j))>10^(-5)
            error('matice H musi byt v hornim Hessenbergove tvaru')
            return;
        end
    end
end
C(1:n-1)=0;
S(1:n-1)=0;
for k=1:n-1
    [c,s]=givens(H(k,k),H(k+1,k));
    C(k)=c;
    S(k)=s;
    H([k,k+1],k:n)=[C(k) -S(k);S(k) C(k)]*H([k,k+1],k:n);
end
for k=1:n-1
    H(1:k+1,[k,k+1])=H(1:k+1,[k,k+1])*[C(k) S(k);-S(k) C(k)];
end
```

Procedura QRHrozklad.m

```
function [Hk,Q,k,vlcA]=QRHrozklad(A,e)
% vstupní proměnné: A...čtvercová matice, e...požadovaná přesnost výpočtu
% výstupní proměnné: Hk...horní trojúhelníková matice s vlastními čísly matice A
% na diagonále, Q...ortogonální matice, jejíž sloupce jsou vlastní vektory matice A,
% k...počet iterací potřebných k dosažení požadované aproximace vlastních čísel
% vlcA...sloupcový vektor, jehož prvky jsou vlastní čísla matice A (vypise se
% pouze v případě, že matice A má komplexní vlastní čísla)
n=size(A);
if length(n)~=2 | n(1)~=n(2)
    error('matice A musí být čtvercová')
    return;
end
n=n(1);
P=ones(n);
for i=1:n
    for j=1:n
        if i>j
            if abs(A(i,j))==0
                P(i,j)=0;
                P(i,i)=0;
                P(j,j)=0;
                P(j,i)=0;
            end
        end
    end
end
end
if P==zeros(n)
    Hk=A; Q=eye(n); k=1; vlcA=[];
    return;
end
```

```

end
[Hk]=HessHous(A);
[Qk,R]=QRHess(Hk);
Q=Qk;
k=1;
vlcA=[];
for i=1:n-1
    while abs(Hk(i+1,i))>e
        if k<100
            [Hk]=RQHess(Hk);
            [Qk,R]=QRHess(Hk);
            Q=Q*Qk;
            k=k+1;
        else
            warning('po 100 iteracich nedosazen uspokojivy vysledek,
            pouzito nahradni reseni, jehoz vysledkem jsou vlastni
            cisla matice A zobrazena v promenne vlcA')
            vlcA(1:n)=0;
            vlcA=vlcA';
            m=1;
            while m<n
                Bk=Hk(m:m+1,m:m+1);
                if abs(Bk(2,1))>10^-3
                    vlcA(m)=(Bk(1,1)+Bk(2,2))./2+sqrt((Bk(1,1)+
                    +Bk(2,2))^2-4*(Bk(1,1)*Bk(2,2)-Bk(1,2)*Bk(2,1)))./2;
                    vlcA(m+1)=(Bk(1,1)+Bk(2,2))./2-sqrt((Bk(1,1)+
                    +Bk(2,2))^2-4*(Bk(1,1)*Bk(2,2)-Bk(1,2)*Bk(2,1)))./2;
                    m=m+2;
                else
                    vlcA(m)=Bk(1,1);
                end
            end
        end
    end
end

```

```
        m=m+1;
    end
end
if m==n
    vlcA(m)=Hk(n,n);
end
Q=[]; k=[];
return;
end
end
end
```

Procedura jacobi.m

```
function [c,s]=jacobi(A,p,q)
% vstupní proměnné: A...symetrická matice, p,q...zvolené indexy
% výstupní proměnné: c,s...parametry c=cos(úhlu), s=sin(úhlu)
n=size(A);
if length(n)~=2 | n(1)~=n(2)
    error('matice A musi byt ctvercova');
    return;
end
if p<q
    if abs(A(p,q))<=10^(-5)
        c=1; s=0;
    else
        tau=(A(q,q)-A(p,p))/(2*A(p,q));
        if tau>=0
            t=1/(tau+sqrt(1+tau^2));
        else
            t=-1/(-tau+sqrt(1+tau^2));
        end
        c=1/sqrt(1+t^2);
        s=t*c;
    end
else
    error('p musi byt mensi nez q')
    return;
end
```

Procedura JacMet.m

```
function [A,Q,k]=JacMet(A,e)
% vstupní proměnné: A...symetrická reálná matice, e...požadovaná přesnost
% výpočtu
% výstupní proměnné: A...diagonální matice, Q...ortogonální matice, k... počet
% iterací potřebných k dosažení požadované aproximace vlastních čísel
n=size(A);
if length(n)~=2 | n(1)~=n(2)
    error('matice A musi byt ctvercova')
    return;
end
n=n(1);
if A==A'
    Q=eye(n);
    tol=e*norm(A);
    [offA]=NedNorm(A);
    k=1;
    while offA>tol
        [p,q]=maximum(A);
        [c,s]=jacobi(A,p,q);
        J=eye(n);
        J(p,p)=c; J(q,q)=c; J(p,q)=s; J(q,p)=-s;
        A=J'*A*J;
        Q=Q*J;
        [offA]=NedNorm(A);
        k=k+1;
    end
else
    error('matice A neni symetricka')
    return;
end
```

```

function [offA]=NedNorm(A)
% vypočítá normu nediagonálních prvků matice A
n=size(A);
n=n(1);
suma=0;
for i=1:n
    for j=1:n
        if i~=j
            suma=suma+A(i,j)^2;
        end
    end
end
offA=sqrt(suma);

function [p,q]=maximum(A)
% najde maximální prvek z nediagonálních prvků matice A
n=size(A);
a=0;
for i=1:n(1)
    for j=1:n(2)
        if i<j
            if abs(a)<abs(A(i,j))
                a=A(i,j);
                p=i; q=j;
            end
        end
    end
end
end
end

```


Literatura

- [1] Bartko, R., Miller, M.: *MATLAB I., algoritmizácia a riešenie úloh*, Digital Graphic, Trenčín, 2004
- [2] Bušinská, T.: *Numerické metódy lineárnej algebry*, Univerzita Komenského, Bratislava, 2005
- [3] Dušek, F., Honc, D.: *Matlab a Simulink: úvod do používání*, Univerzita Pardubice, 2005
- [4] Fiedler, M.: *Speciální matice a jejich použití v numerické matematice*, Praha, SNTL, 1981
- [5] Golub, G.H.: *Matrix computations*, The Johns Hopkins Press Ltd., London, 1996
- [6] Holenda, J.: *O maticích*, Vydavatelský servis, Plzeň, 2007
- [7] Horová, I., Zelinka, J.: *Numerické metody*, MU Brno, 2004
- [8] Kobza, J.: *Numerické metody*, UP Olomouc, 1993
- [9] Lepišová, M.: *Vlastní čísla a vlastní vektory matic*, Bakalářská práce, PřF, UP Olomouc, 2007
- [10] Míka, S., Brandner, M.: *Numerické metody I.*, ZČU Plzeň, 2000
- [11] Novák, J.: *Základy informatiky: počítačové modelování v Matlabu*, ČVUT Praha, 2005
- [12] Ralston, A.: *Základy numerické matematiky*, Academia Praha, nakladatelství Československé akademie věd, 1978
- [13] Rybička, J.: *LaTeX pro začátečníky*, KONVOJ Brno, 2003
- [14] Saad, Y.: *Numerical methods for large eigenvalue problems*, John Wiley and Sons, New York, 1992