Cvičení 8 - zadání pro studenty

Základy numerické matematiky - NMNM201 Verze z 20. listopadu 2023

1 Lokalizace vlastních čísel pomocí Gerschgorinovy věty

Věta 1. *Nechť $A \in \mathbb{C}^{n \times n}$ a r_i značí součet mimodiagonálních prvků v i-tém řádku*

$$r_i = \sum_{i=1, j \neq i}^{n} |a_{ij}|.$$

Pak všechna vlastní čísla matice A leží ve sjednocení Gerschgorinovych kruhů $\cup_{i=1}^{n} D_i$, kde

$$D_i = \{z \in \mathbb{C} : |z - a_{ii}| \leq r_i\}.$$

Pokud m kruhů tvoří souvislou oblast, která je disjunktní od ostatních, pak právě m vlastních čísel matice A leží v této souvislé oblasti.

Úloha 1. *Pomocí Gerschgorinovy věty lokalizujte vlastní čísla matice*

$$A = \begin{bmatrix} 3 & 1 & 1 \\ 2 & 2 & 0 \\ 1 & -1 & -4 \end{bmatrix}.$$

Úloha 2. *(navíc) Občas jednoduchou podobnostní transformací můžeme matici A převést na $D^{-1}AD$, jejíž Gerschgorinovy kruhy nám o vlastních číslech původní matice prozradí víc. Uvažujte*

$$D = diag(1, 2, 4)$$

pro matici A z předchozí úlohy a znovu lokalizujte její vlastní čísla.

2 Stacionární iterační metody

Přímé metody (jako například LU rozklad) pro řešení soustav lineárních rovnic $Ax = b$ s regulární maticí, po nějaké době výpočtu, vydají jedno numerické řešení. Myšlenka iteračních metod je principálně odlišná, spočívá v konstrukci *posloupnosti* aproximací (přibližných řešení) $x^{(0)}, x^{(1)}, \ldots$, jež by se měla přibližovat skutečnému řešení x. Výhodou iteračních metod je, že (nějakou) aproximaci získáváme v každé iteraci, tj. kdykoli zastavíme výpočet.

2.1 Klasické iterační metody

Klasické iterační metody jsou založeny na štěpení matice soustavy $A = M - N$, kde matice M je regulární a snadno invertovatelná. Dosazením do vztahu $Ax = b$ postupně dostáváme

$$\begin{aligned} (M - N)x &= b \\ Mx &= Nx + b \\ x &= M^{-1}Nx + M^{-1}b. \end{aligned}$$

Je-li dána počáteční aproximace řešení x_0, můžeme definovat iterační proces

$$x_k = M^{-1}Nx_{k-1} + M^{-1}b.$$

Pro analýzu stacionárních iteračních metod je důležitý následující vztah mezi chybami dvou následujících přibližných řešení x_{k-1} a x_k:

$$x - x_k = M^{-1}N(x - x_{k-1}) = (I - M^{-1}A)(x - x_{k-1}).$$

2.2 Příklady klasických iteračních metod

Klasické iterační metody jsou založeny na štěpení ve tvaru $A = D - L - U$, kde D je hlavní diagonála, $-L$ je striktně dolní trojúhelník matice A a $-U$ je striktně horní trojúhelník matice A. Jednotlivé metody pak lze odvodit z rovnice

$$(D - L - U)x = b.$$

- Jacobiho metoda je definována iterací

$$Dx_k = Lx_{k-1} + Ux_{k-1} + b,$$

- Gauss–Seidelova metoda je zas definována jako

$$Dx_k = Lx_k + Ux_{k-1} + b.$$

2.3 Asymptotická konvergence

Z přednášky víme, že metoda je konvergentní právě tehdy, když $\rho(M^{-1}N) < 1$. Tím myslíme, že pro libovolný počáteční vektor chyba $x - x_k$ konverguje k nulovému vektoru.

Úloha 3. *Pro matici*

$$A = \begin{bmatrix} 1 & 0.5 & 0.5 \\ 0.5 & 1 & 0.5 \\ 0.5 & 0.5 & 1 \end{bmatrix} \qquad \textit{případně (navíc)} \quad A_2 = \begin{bmatrix} 1 & 0 & 1 \\ -1 & 1 & 0 \\ 1 & 2 & -3 \end{bmatrix}$$

odvoďte matici $M^{-1}N$ z Jacobiho a Gauss–Seidelovy metody a rozhodněte, zda metody budou konvergentní, nebo ne.
Pro výpočty inverzí matice a vlastních čísel můžete využít MATLAB.

2.4 Přechodový jev

Zatímco vlastnost $\rho(M^{-1}N) < 1$ zaručuje, že chyba $x - x_k$ konverguje k nulovému vektoru, což také zaručuje $\|x - x_k\|_* \to^{k\to\infty} 0$ pro libovolnou vektorovou normu, pro popis $\|x - x_k\|_*$ v úvodních iteracích (pro malé k) nemusí být $\rho(M^{-1}N)$ vypovídající.
Situaci, kdy chyba $\|x - x_k\|_*$ roste před tím, než dosáhne asymptotického chování odpovídajícímu $(\rho(M^{-1}N))^k$, říkáme *přechodový jev*.

Úloha 4. *Naprogramujte Jacobiho a Gauss-Seidelovu metodu (doplňte předpřipravené skripty) a vyzkoušejte na skriptu* `iteracni_metody_jac_gs.m`*.*
[Hint: Nastudujte si v nápovědě MATLABu funkce `diag` *(z matice „vyzobne“ diagonálu jako vektor, z vektoru vytvoří diagonální matici),* `tril` *a* `triu`*.]*
I na základě pozorování odpovězte na následující otázky, případně odkažte na konkrétní úlohu ze skriptu `iteracni_metody_jac_gs.m`*:*

- *Konverguje-li metoda například v Euklidovské normě, musí konvergovat i v jiných vektorových normách?*
- *Kdy máme zaručenu monotonní konvergenci (například v Euklidovské normě)?*
- *Souvisí přítomnost přechodového jevu (tj. jevu, kdy chyba na začátku výpočtu nejprve roste) s velikostí maticových norem či spektrálního poloměru iterační matice?*

- *Lze v plné obecnosti vzájemně porovnat Jacobiho a Gauss-Seidelovu metodu? (Porovnáním máme na mysli výpovědi typu: Gauss-Seidelova metoda má vždy/nikdy rychlejší konvergenci než Jacobiho metoda. Jacobiho metoda konverguje pouze když/právě když Gauss-Seidelova metoda, atp.)*

Úloha 5 (navíc). *Naprogramujte Jacobiho a Gauss-Seidelovu metodu s výpočtem nové aproximace po složkách.*

[Poznámka: tento způsob implementace bude v MATLABu pravděpodobně pomalejší, protože MATLAB je optimalizovaný pro práci s maticemi a vektory.]