2 Iterační metody pro řešení soustav lineárních rovnic s SPD maticí

2.1 Prostá iterační metoda

Metoda daná předpisem

$$u_{i+1} = u_i + (b - Au_i)\,.$$

Chyba po $(i+1)$. iteraci:

$$\begin{aligned} e_{i+1} &= u_{i+1} - u^* = u_i + b - A\cdot u_i - u^* = u_i + A\cdot u^* - A\cdot u_i - u^* = e_i - A\cdot e_i = \\ &= (I-A)\cdot e_i = (I-A)^{i+1}\cdot e_o \end{aligned}$$

Příklad 2.1.

$$\begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix} \cdot \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$\begin{vmatrix} \lambda - 2 & -1 \\ -1 & \lambda - 2 \end{vmatrix} = (\lambda-2)^2 - 1 \Rightarrow \lambda_1 = 3,\ \lambda_2 = 1$$

Vlastní čísla zadané matice jsou kladná, matice je tedy symetrická a pozitivně definitní. Její normované vlastní vektory označme v_1, v_2, ($\|v_i\| = 1$).

$$e_0 = \sum_{k=1}^{2} \alpha_k v_k$$

$$e_i = (I-A)^i \cdot e_0 = (I-A)^i \cdot \sum_{k=1}^{2} \alpha_k v_k = \sum_{k=1}^{2} \alpha_k \cdot (I-A)^i \cdot v_k = \sum_{k=1}^{2} \alpha_k \cdot (1-\lambda_k)^i \cdot v_k$$

$$\|e_i\| = \left[\sum_{k=1}^{2} \alpha_k^2 \cdot (1-\lambda_k)^{2i}\right]^{1/2} \to 0$$

a tedy dostáváme podmínku konvergence $(1-\lambda_k)^i \to 0$. Metoda je tedy konvergentní, pokud $\lambda_k \in (0,2)$, což nenastane ve výše uvedeném případě. Pomocí může být jednoduché škálování pomocí $\omega > 0$:

$$A\cdot u = b \ \to\ \omega\cdot A\cdot u = \omega\cdot b \ \text{(škálovaná soustava)},$$

$$A\cdot v = \lambda\cdot v \ \to\ \omega\cdot A\cdot v = \omega\cdot\lambda\cdot v$$

$$u_{i+1} = u_i + (\omega\cdot b - \omega\cdot A\cdot u_i) = u_i + \omega\,(b - A\cdot u_i)$$

Například volbou $\omega = \frac{1}{3}$ dostaneme matici ωA, jejíž vlastní čísla jsou menší než 2 a máme zaručenu konvergenci. Získáváme tak Richardsonovu metodu. △

2.2 Richardsonova metoda

Richardsonova metoda je dána iteračním předpisem

$$u_{i+1} = u_i + \omega\,(b - Au_i)\,,$$

kde ω je vhodná konstanta.

Algoritmus 1 Richardsonova metoda

```
u_0 ←
for i = 0, 1, ...
    r_i = b − A · u_i
    u_{i+i} = u_i + ω · r_i
end
```

2.2.1 Konvergence iterační metody

Richardsonova metoda patří mezi iterační metody minimalizující eukleidovskou normu chyby.

Věta 2.1. *Nechť A je SPD, ω je vhodná konstanta, přesněji $0 < \omega < 2/\lambda_{max}(A)$. Potom Richadsonova metoda daná předpisem $u_{i+1} = u_i + \omega\,(b - Au_i)$ konverguje, přesněji*

$$\|e_{i+1}\|_{(m)} = \|u_{i+1} - u^*\|_{(m)} \le q^i \cdot \|u_0 - u^*\|_{(m)}$$

$$kde\ \ q = \max_k |1 - \omega\lambda_k| < 1,\ \ m = 0,\,1,\,\ldots$$

Důkaz. Vyjádříme počáteční chybu e_0 v bázi vlastních vektorů matice A a dosadíme do vzorce pro výpočet chyby $(i-1)$. iterace.

$$e_0 = \sum \alpha_k v_k$$

$$e_i = (I-\omega A)^i\, e_0 = (I-\omega A)^i \cdot \sum \alpha_k v_k = \sum \alpha_k \cdot (1-\omega\lambda_k)^i \cdot v_k$$

$$\begin{aligned} \|e_i\|^2_{(m)} &= \left\|\sum \alpha_k \cdot (1-\omega\lambda_k)^i \cdot v_k\right\|^2_{(m)} = \\ &= \left(\sum \alpha_k \cdot \lambda_k^m \cdot (1-\omega\lambda_k)^i \cdot v_k,\ \sum \alpha_k \cdot (1-\omega\lambda_k)^i \cdot v_k\right) = \\ &= \sum_k \sum_l \alpha_k \cdot \lambda_k^m\,(1-\omega\lambda_k)^i \cdot \alpha_l \cdot (1-\omega\lambda_l)^i \cdot (v_k,\, v_l) = \\ &= \sum \alpha_k^2 \cdot \lambda_k^m\,(1-\omega\lambda_k)^{2i} \le \\ &\le \max_k |1-\omega\lambda_k|^{2i} \cdot \sum \lambda_k^m \alpha_k^2 = \max_k |1-\omega\lambda_k|^{2i} \cdot \|e_0\|^2_{(m)} = \\ &= \max_k \left|P_i^R(\lambda_k)\right|^2 \cdot \|e_0\|^2_{(m)} \le q^{2i} \cdot \|e_0\|^2_{(m)}\,, \quad P_i^R(\lambda) = (1-\omega\lambda)^i \ (1) \end{aligned}$$

□

Poznámka 2.1. Polynom

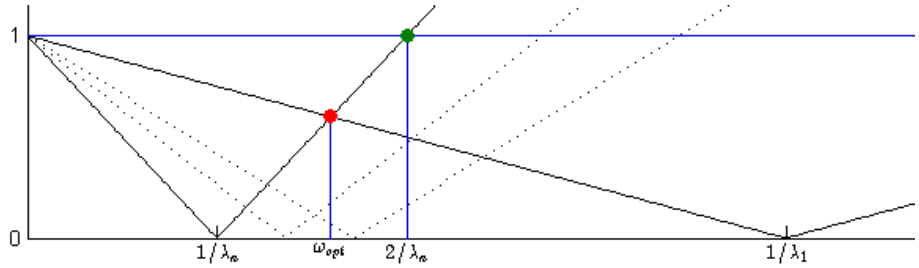
$$P_i^R(\lambda) = P_i^{R,\omega}(\lambda) = (I - \omega\lambda)^i, \tag{2}$$

který se objevuje v (1), nazveme iteračním polynomem Richardsonovy metody. Redukce chyby souvisí s hodnotami tohoto polynomu v prvcích $\lambda_k \in \sigma(A)$.

2.2.2 Volba optimálního faktoru

Konstantu ω se snažíme zvolit tak, aby norma rezidua klesala co nejrychleji. Nejlepší z takových konstant nazýváme optimálním faktorem a značíme ω_{opt}.

Konvergenční faktor $q = \max|1 - \omega\lambda_k|$ musí být nutně menší než 1 a ideálně co nejmenší. Hledáme tedy takovou hodnotu ω, pro kterou je $\max|1 - \omega\lambda_k|$ co nejmenší, viz. obrázek 1.



Obrázek 1: Optimální faktor

$$\omega_{\text{opt}}\lambda_n - 1 = 1 - \omega_{opt}\lambda_1 \Longrightarrow \omega_{\text{opt}} = \frac{2}{\lambda_n + \lambda_1} \tag{3}$$

$$q_{\text{opt}} = 1 - \frac{2}{\lambda_n + \lambda_1}\lambda_1 = \frac{\lambda_n - \lambda_1}{\lambda_n + \lambda_1} = \frac{\frac{\lambda_n}{\lambda_1} - 1}{\frac{\lambda_n}{\lambda_1} + 1} = \frac{\kappa(A) - 1}{\kappa(A) + 1} \tag{4}$$

Optimální faktor můžeme vyjádřit také pomocí iteračního polynomu. Minimalizací polynomu (2) získáme právě výše odvozený optimální faktor ω_{opt}, tedy

$$\omega_{\text{opt}} = \underset{\omega}{\operatorname{argmin}} \max_{i=1,\dots,n} \left|P_\omega^R(\lambda_i)\right| = \frac{2}{\lambda_1 + \lambda_n}.$$

Problém je ale v tom, že vlastní čísla matice A většinou neznáme a určit je by bylo složitější než řešit původní úlohu. Obvykle je tedy nahrazujeme jejich odhady, můžeme použít např. Geršgorinovu větu. Pokud $\underline{\lambda}$ je odhad nejmenšího vlastního čísla $(0 \le \underline{\lambda} \le \lambda_1)$ a $\overline{\lambda}$ je odhad největšího vlastního čísla $\left(\overline{\lambda} \ge \lambda_n\right)$, zvolíme kvazi optimální faktor

$$\omega_{kopt} = \frac{2}{\underline{\lambda} + \overline{\lambda}}.$$

Pro konvergenční faktor dostaneme odhad

$$q = 1 - \frac{2}{\underline{\lambda} + \overline{\lambda}} \cdot \lambda_1 = \frac{\underline{\lambda} + \overline{\lambda} - 2 \cdot \lambda_1}{\underline{\lambda} + \overline{\lambda}} \le \frac{\overline{\lambda} - \underline{\lambda}}{\overline{\lambda} + \underline{\lambda}} = \frac{c - 1}{c + 1},$$

kde $c = \frac{\overline{\lambda}}{\underline{\lambda}} \ge \kappa(A)$. Neznáme-li pro λ_1 lepší odhad než hodnotu 0, použijeme

$$\omega = \frac{2}{\overline{\lambda}}.$$

Pokud zvolíme ω mimo interval $(0,\, 2/\lambda_{max})$, budou některé složky chyby narůstat. Pokud zvolíme ω z uvedeného intervalu, budou všechny složky klesat, nejvíce ty, pro které je λ_k blízko $1/\omega$. Vhodnou volbou ω tak lze dosáhnout toho, že budou dobře redukovány složky odpovídající velkým vlastním číslům, opačně to nelze. Pomocí Richardsovy metody lze dosáhnout rychlé redukce oscilujících složek (složek odpovídajících oscilujícím vlastním vektorům), velmi efektivní metodu dostaneme, když zbývající složky (odpovídající méně oscilujícím, hladkým vlastní vektorům) redukujeme jinou technikou - např. korekcí vypočtenou pomocí hrubší diskretizace, viz [Multigrid].

2.2.3 Ukončení iterací

- sledovat rozdíl iterací $\delta_{i+1} = u_{i+1} - u_i = \omega r_i$
- sledovat reziduum $r_i = b - A \cdot u_i \longrightarrow$ ukončovací podmínka $\|r_i\| \le \epsilon \cdot \|r_0\|$

Kolik iterací je potřeba pro dosažení dané přesnosti ε**:**

$$\begin{aligned} q^i &\le \varepsilon \\ i \cdot \ln q &\le \ln\varepsilon \ \ (q, \varepsilon < 1) \\ i &\ge \frac{\ln\varepsilon}{\ln q} \end{aligned}$$

Výhody a nevýhody metody:

1. jednoduché operace, možná paralelizace
2. pomalá konvergence
3. nutnost odhadnout parametr ω

Zrychlení metody:

- volba různých parametrů $\omega = \omega_i$ pro jednotlivé iterace (vede na Čebyševovu metodu)
- pro opravu vylepšit směr rezidua (vede na metodu sdružených gradientů)