1 Iterační metody pro řešení soustav lineárních rovnic

Přímé metody řešení soustav lineárních rovnic (LU, LDLT, Choleského faktorizace atd.) vyžadují $\mathcal{O}(n^3)$ operací a velikost soustavy, kterou jimi dokážeme vyřešit je značně omezená. V případě husté matice $\mathsf{A} \in \mathbb{R}^{n\times n}$ se přibližná velikost řešitelné soustavy historicky vyvíjela takto:

- 1950: $n = 20$ (Wilkinson)
- 1965: $n = 200$ (Forsythe a Moler)
- 1980: $n = 2000$ (LINPACK)
- 1995: $n = 20000$ (LAPACK)
- 2010: $n = 200000$ (HDSS)

Pracujeme-li s řídkými maticemi, jsme schopni řešit i systémy s mnohem větší dimenzí (miliony, desítky milionů neznámých) – zejména, pokud je řešič schopen pracovat paralelně. V případě použití přímého řešiče na řídkou matici však může dojít k jejímu zaplnění. Např. využitím prvního řádku následující matice k vynulování prvního sloupce dojde k zaplnění všech ostatních prvků v matici:

$$\begin{bmatrix} \times & \times & \times & \times & \times \\ \times & \times & 0 & 0 & 0 \\ \times & 0 & \times & 0 & 0 \\ \times & 0 & 0 & \times & 0 \\ \times & 0 & 0 & 0 & \times \end{bmatrix} \rightarrow \begin{bmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \end{bmatrix}$$

Tento problém lze částečně řešit vhodnou pivotizací.

Mezi další nevýhody přímých řešičů patří:

- Známe-li již přibližné řešení soustavy, nedokážeme tuto znalost využít ke snížení celkového počtu operací a zkrácení doby výpočtu.
- Naopak, pokud nám postačuje znalost pouze přibližného řešení, nemůžeme výpočet pomocí přímého řešiče ukončit předčasně.

Alternativou k přímým řešičům jsou iterační řešiče, které generují posloupnost přibližných řešení $\{\boldsymbol{x}^k\}$ a pracují téměř výhradně s násobením matice-vektor, které má náročnost $\mathcal{O}(n^2)$. Důležitou vlastností každé iterační metody je rychlost konvergence posloupnosti $\{\boldsymbol{x}^k\}$ k řešení. Může se totiž stát, že pro některé matice A iterační metoda konverguje velmi pomalu nebo vůbec.

1.1 Lineární iterační metody

Prvním typem iteračních metod, kterým se budeme zabývat, jsou tzv. lineární iterační metody. Ty hledají posloupnost řešení soustavy $\mathsf{A}\boldsymbol{x} = \boldsymbol{b}$ ve tvaru

$$\boldsymbol{x}^{k+1} := \mathsf{M}\boldsymbol{x}^k + \mathsf{N}\boldsymbol{b}, \tag{1.1}$$

kde M a N jsou nějaké matice odpovídajících rozměrů[1].

Definice Lineární iterační metodu nazveme konzistentní, řeší-li rovnici $\mathsf{M}\boldsymbol{x} + \mathsf{N}\boldsymbol{b} = \boldsymbol{x}$ právě jeden vektor $\boldsymbol{x} = \mathsf{A}^{-1}\boldsymbol{b}$. Je možné ukázat, že metoda je konzistentní právě tehdy, je-li splněno $\mathsf{M} = \mathsf{I} - \mathsf{NA}$.

Definice Iterační metodu nazveme konvergentní platí-li $\boldsymbol{x}^k \to \boldsymbol{x} = \mathsf{A}^{-1}\boldsymbol{b}$ pro $k \to \infty$. Je možné ukázat, že metoda je konvergentní, právě tehdy, je-li splněno $\|\mathsf{M}\| < 1$, kde $\|\mathsf{M}\| = \max_{\boldsymbol{v}\in\mathbb{R}^n} \frac{\|\mathsf{M}\boldsymbol{v}\|_v}{\|\boldsymbol{v}\|_v}$ je maticová norma indukovaná vektorovou normou.

Při odvozování následujících iteračních metod budeme využívat rozkladu matice A na součet dolní trojúhelníkové, diagonální a horní trojúhelníhové matice, tedy $\mathsf{A} = \mathsf{L} + \mathsf{D} + \mathsf{U}$ (pozor, nepleťte si matice $\mathsf{L}, \mathsf{D}, \mathsf{U}$ se stejně nazvanými maticemi, které se vyskytovaly u přímých řešičů). Např. matici

$$\mathsf{A} = \begin{bmatrix} 2 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 1 & 2 \end{bmatrix}$$

rozložíme na

$$\mathsf{L} = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \mathsf{D} = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix}, \mathsf{U} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}.$$

1.1.1 Jacobiho metoda

Vyjděme z rovnice $\mathsf{A}\boldsymbol{x} = \boldsymbol{b}$. Dosazením $\mathsf{A} = \mathsf{L} + \mathsf{D} + \mathsf{U}$ dostaneme

$$(\mathsf{L} + \mathsf{D} + \mathsf{U})\boldsymbol{x} = \boldsymbol{b}.$$

Roznásobme a přezávorkujme výraz na levé straně

$$\mathsf{D}\boldsymbol{x} + (\mathsf{L} + \mathsf{U})\boldsymbol{x} = \boldsymbol{b}$$

Jacobiho metodu odvodíme tak, že přidáme indexy $k+1$ a k k příslušným vektorům $\boldsymbol{x}$

$$\mathsf{D}\boldsymbol{x}^{k+1} + (\mathsf{L} + \mathsf{U})\boldsymbol{x}^k = \boldsymbol{b}.$$

Osamostatněním $\boldsymbol{x}^{k+1}$ dostaneme předpis pro $k+1$ aproximaci vektoru $\boldsymbol{x}$

$$\boldsymbol{x}^{k+1} := \mathsf{D}^{-1}(\boldsymbol{b} - (\mathsf{L} + \mathsf{U})\boldsymbol{x}^k) = \underbrace{-\mathsf{D}^{-1}(\mathsf{L} + \mathsf{U})}_{=\mathsf{M}}\boldsymbol{x}^k + \underbrace{\mathsf{D}^{-1}}_{=\mathsf{N}}\boldsymbol{x}^k.$$

[1] Index k označuje číslo aktuální iterace.

Jednotlivé složky vektoru $\boldsymbol{x}^{k+1}$ můžeme vyjádřit jako

$$(\boldsymbol{x}^{k+1})_i := \frac{1}{a_{i,i}} \left(b_i - \sum_{j=1, j\neq i}^{n} a_{i,j} (\boldsymbol{x}^k)_j \right) = \tag{1.2}$$

$$= \frac{1}{a_{i,i}} \left(b_i - \sum_{j=1}^{i-1} a_{i,j} (\boldsymbol{x}^k)_j - \sum_{j=i+1}^{n} a_{i,j} (\boldsymbol{x}^k)_j \right) \tag{1.3}$$

Pro snadnější a přehlednější zápis budeme i-tý prvek vektoru $\boldsymbol{x}^{k+1}$ také značit jako x_i^{k+1} (horní index tedy označuje číslo iterace, dolní index značí pořadí prvku ve vektoru).

Konzistence metody vyplývá z jejího odvození, můžeme však ještě ověřit, že $\mathsf{M} = \mathsf{I} - \mathsf{NA}$. V případě Jacobiho metody je $\mathsf{M} = -\mathsf{D}^{-1}(\mathsf{L}+\mathsf{U})$ a $\mathsf{N} = \mathsf{D}^{-1}$ (viz výše). Platí tedy

$$\begin{aligned} \mathsf{I} - \mathsf{NA} &= \mathsf{I} - \mathsf{D}^{-1}\mathsf{A} = \mathsf{I} - \mathsf{D}^{-1}(\mathsf{L}+\mathsf{D}+\mathsf{U}) = \\ &= \mathsf{I} - \mathsf{D}^{-1}(\mathsf{L}+\mathsf{U}) - \mathsf{D}^{-1}\mathsf{D} = -\mathsf{D}^{-1}(\mathsf{L}+\mathsf{U}) = \mathsf{M}. \end{aligned}$$

Metoda je tedy konzistentní.

Lze dokázat, že metoda je konvergentní právě tehdy, když $\|\mathsf{M}\| = \|\mathsf{D}^{-1}(\mathsf{L}+\mathsf{U})\| < 1$. To splňují např. striktně diagonálně dominantní matice (tedy matice, pro které platí $\forall i : |a_{i,i}| > \sum_{j=1, j\neq i}^{n} |a_{i,j}|$). Konvergenci metody pro diagonálně dominantní matice se dá poměrně snadno dokázat. Vyjděme ze vztahu pro i-tý prvek aproximovaného vektoru v iteraci $k+1$:

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

Jelikož je metoda konzistentní, musí tuto rovnost splňovat i prvky vektoru přesného řešení $\boldsymbol{x}$:

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

Odečteme-li od první rovnosti druhou, dostaneme

$$\underbrace{x^{k+1} - x_i}_{=e_i^{k+1}} = -\frac{1}{a_{i,i}} \sum_{j=1, j\neq i}^{n} a_{i,j} (\underbrace{x_j^k - x_j}_{=e_i^k}),$$

kde $\boldsymbol{e}^k$ je vektor chyby v k-tém kroku. Chybu v kroku $k+1$ můžeme tedy odhadnout pomocí vlastností striktně diagonálně dominantní matice:

$$|e_i^{k+1}| \leq \frac{1}{|a_{i,i}|} \sum_{j=1, j\neq i}^{n} |a_{i,j}||e_j^k| \leq \frac{1}{|a_{i,i}|} \sum_{j=1, j\neq i}^{n} |a_{i,j}| \max_{j=1,\ldots,n, j\neq i} |e_j^k| =$$

$$= \max_{j=1,\ldots,n, j\neq i} |e_j^k| \underbrace{\frac{1}{|a_{i,i}|} \sum_{j=1, j\neq i}^{n} |a_{i,j}|}_{<1} < \max_{j=1,\ldots,n, j\neq i} |e_j^k|$$

Každý prvek vektoru chyby v kroku $k+1$ je tedy v absolutní hodnotě menší než maximální prvek vektoru chyby v předchozím kroku. Vektor chyby tedy konverguje k nulovému vektoru.

1.1.2 Gaussova-Seidelova metoda

Všimněme si, že při výpočtu x_i^{k+1} využíváme v sumě $\sum_{j=1}^{i-1} a_{i,j} x_j^k$ ve výrazu (1.3) pouze prvky $x_1^k, \ldots, x_{i-1}^k$. Tyto prvky tedy můžeme nahradit již vypočtenými prvky aktuálními iterace $x_1^{k+1}, \ldots, x_{i-1}^{k+1}$. Dostaneme tak předpis Gaussovy-Seidelovy metody:

$$x_i^{k+1} := \frac{1}{a_{i,i}} \left(b_i - \sum_{j=1}^{i-1} a_{i,j} x_j^{k+1} - \sum_{j=i+1}^{n} a_{i,j} x_j^k \right) \tag{1.4}$$

Podobně jako v předchozím případě můžeme metodu odvodit, nahradíme-li v soustavě $\mathsf{A}\boldsymbol{x} = \boldsymbol{b}$ matici A součtem $\mathsf{L} + \mathsf{D} + \mathsf{U}$. Tentokrát ovšem vektor s indexem $k+1$ ponecháme u součtu $\mathsf{L} + \mathsf{D}$

$$(\mathsf{L} + \mathsf{D})\boldsymbol{x}^{k+1} = \boldsymbol{b} - \mathsf{U}\boldsymbol{x}^k, \tag{1.5}$$

tedy

$$\boldsymbol{x}^{k+1} = \underbrace{-(\mathsf{L} + \mathsf{D})^{-1}\mathsf{U}}_{=\mathsf{M}} \boldsymbol{x}^k + \underbrace{(\mathsf{L} + \mathsf{D})^{-1}}_{=\mathsf{N}} \boldsymbol{b}.$$

Vztah mezi maticovým zápisem a zápisem po prvcích (1.4) je nejlépe vidět na rovnosti (1.5). Jedná se o soustavu rovnic s dolní trojúhelníkovou maticí $\mathsf{L} + \mathsf{D}$, vektorem pravé strany $\boldsymbol{b} - \mathsf{U}\boldsymbol{x}^k$ a neznámým vektorem $\boldsymbol{x}^{k+1}$. Všimněte si, že výraz (1.4) pak přesně odpovídá algoritmu pro dopřednou substituci pro řešení takovéto soustavy.

Podobně jako v případě Jacobiho metody můžeme ověřit, zda je metoda konzistentní porovnáním $\mathsf{I} - \mathsf{NA}$ a M.

$$\mathsf{I} - \mathsf{NA} = \mathsf{I} - (\mathsf{L} + \mathsf{D})^{-1}\mathsf{A} = \mathsf{I} - (\mathsf{L} + \mathsf{D})^{-1}(\mathsf{L} + \mathsf{D} + \mathsf{U}) =$$
$$= \mathsf{I} - (\mathsf{L} + \mathsf{D})^{-1}(\mathsf{L} + \mathsf{D}) - (\mathsf{L} + \mathsf{D})^{-1}\mathsf{U} = -(\mathsf{L} + \mathsf{D})^{-1}\mathsf{U} = \mathsf{M}.$$

Metoda je tedy konzistentní.

Metoda je konvergentní, právě když $\|(\mathsf{L} + \mathsf{D})^{-1}\mathsf{U}\| < 1$, což opět platí pro diagonálně dominantní matice.

1.1.3 Richardsonova metoda

Iterace Richardsonovy metody je dána předpisem

$$\boldsymbol{x}^{k+1} := \boldsymbol{x}^k + \omega \boldsymbol{r}^k,$$

kde $\omega \in \mathbb{R}_+$ a $\boldsymbol{r}^k = \boldsymbol{b} - \mathsf{A}\boldsymbol{x}^k$ je reziduum, které určuje, jak dobře je splněna původní rovnice. Vztah mezi reziduem a chybou $\boldsymbol{e}^k = \boldsymbol{x}^k - \boldsymbol{x}$ lze odvodit přenásobením definice chyby maticí A

$$\mathsf{A}\boldsymbol{e}^k = \mathsf{A}\boldsymbol{x}^k - \mathsf{A}\boldsymbol{x} = \mathsf{A}\boldsymbol{x}^k - \boldsymbol{b} = -\boldsymbol{r}^k.$$

Studujme konvergenci metody pro symetrickou pozitivně definitní matici A. V takovém případě vlastní čísla λ_i a vlastní vektory $\boldsymbol{v}_i$ (tedy skaláry a vektory, pro které platí $\mathsf{A}\boldsymbol{v}_i = \lambda_i \boldsymbol{v}_i, \|\boldsymbol{v}\| = 1$) splňují

$$0 < \lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_n$$

a

$$\operatorname{span}\{\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_n\} = \mathbb{R}^n.$$

Zde span značí lineární obal. vlastní vektory tvoří ortonormální bázi $\mathbb{R}^n$.

Díky předchozímu poznatku můžeme reziduum vyjádřit jako lineární kombinaci prvků báze tvořené vlastními vektory, tedy $\boldsymbol{r}^{k+1} = \sum_{i=1}^n \alpha_i^{k+1} \boldsymbol{v}_i$. Studujme nyní, jak se chová reziduum (a tedy i chyba) v jednotlivých iteracích. Na základě toho se později pokusíme odvodit optimální hodnotu ω pro co nejrychlejší konvergenci.

$$\begin{aligned}
\sum_{i=1}^n \underline{\alpha_i^{k+1}} \boldsymbol{v}_i = \boldsymbol{r}^{k+1} = \boldsymbol{b} - \mathsf{A}\boldsymbol{x}^{k+1} = \underbrace{\boldsymbol{b} - \mathsf{A}(\boldsymbol{x}^k}_{=\boldsymbol{r}^k} + \omega \boldsymbol{r}^k) = \\
= \boldsymbol{r}^k - \mathsf{A}\omega\boldsymbol{r}^k = (\mathsf{I} - \omega\mathsf{A}) \underbrace{\boldsymbol{r}^k}_{=\sum_{i=1}^n \alpha_i^k \boldsymbol{v}_i} = (\mathsf{I} - \omega\mathsf{A}) \sum_{i=1}^n \alpha_i^k \boldsymbol{v}_i = \\
= \sum_{i=1}^n \alpha_i^k \boldsymbol{v}_i - \sum_{i=1}^n \alpha_i^k \omega \underbrace{\mathsf{A}\boldsymbol{v}_i}_{=\lambda_i \boldsymbol{v}_i} = \sum_{i=1}^n \alpha_i^k \boldsymbol{v}_i - \sum_{i=1}^n \alpha_i^k \omega \lambda_i \boldsymbol{v}_i = \\
= \sum_{i=1}^n \underline{(1 - \omega\lambda_i)\alpha_i^k} \boldsymbol{v}_i.
\end{aligned}$$

Všimněme si, že jsme vyjádřili koeficienty rozvoje rezidua $\boldsymbol{r}^{k+1}$ v bázi $\{\boldsymbol{v}_i\}_{i=1}^n$ pomocí násobků koeficientů v předchozím kroku (viz podtržené části předchozího výrazu). Koeficienty se tedy budou zmenšovat (a jednotlivé složky vektoru rezidua budou konvergovat k nule) právě tehdy, když $|1 - \omega\lambda_i| < 1$ pro všechna $i = 1, 2, \ldots, n$. Rychlost konvergence bude záviset na největší hodnotě $|1 - \omega\lambda_i|$. Shrněme tento poznatek do následující věty.

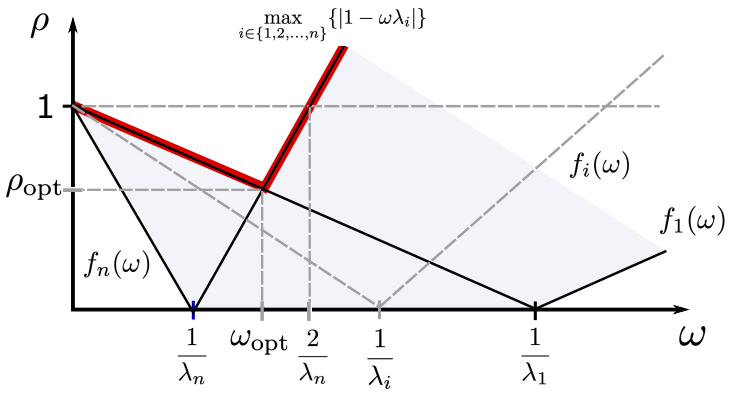

Figure 1.1: Konvergenční faktor Richardsonovy metody v závislosti na ω.

Věta Richardsonova metoda konverguje, právě když $\forall i \in \{1, 2, \dots, n\} : |1 - \omega\lambda_i| < 1$. Konvergenční faktor $\rho = \max_{i\in\{1,2,\dots,n\}}\{|1 - \omega\lambda_i|\}$ určuje rychlost konvergence: $\|\boldsymbol{r}^{k+1}\| \leq \rho\|\boldsymbol{r}^k\|$.

Čím menší bude konvergenční faktor $\rho = \max_{i\in\{1,2,\dots,n\}}\{|1-\omega\lambda_i|\}$, tím rychleji bude metoda konvergovat. Vzhledem k tomu, že vlastní čísla matice A jsou daná, můžeme konvergenční faktor ovlivnit pouze vhodnou volbou parametru ω. Odvození ideální hodnoty ω ilustrujme na Obrázku 1.1. Jsou na něm znázorněny funkce $f_1(\omega) = |1 - \omega\lambda_1|$ a $f_n(\omega) = |1 - \omega\lambda_n|$. Protože směrnice funkcí $f_i(\omega) = |1 - \omega\lambda_i|$ jsou určeny vlastními čísly matice A a ta jsou seřazena od nejmenšího po největší, budou grafy všech funkcí $f_i, i = 2, \dots, n-1$, ležet "mezi" grafy f_1 a f_n (v šedě vyznačené oblasti). Funkci

$$\max_{i\in\{1,2,\dots,n\}}\{|1 - \omega\lambda_i|\}$$

tedy můžeme vykreslit jako červeně zvýrazněnou lomenou čáru tvořenou částí funkce f_1 a částí funkce f_n. Z grafu této funkce tedy rovnou můžeme odvodit:

1. Interval, ve kterém musí ω ležet. Aby metoda konvergovala, musí platit $\rho = \max_{i\in\{1,2,\dots,n\}}\{|1 - \omega\lambda_i| < 1$. Červená funkce tedy musí ležet pod zakreslenou konstantní funkcí $\rho = 1$. Levý krajní bod intervalu je 0, pravý určíme jako průsečík příslušné části funkce f_n s konstantní funkcí 1:

   $$-(1 - \omega\lambda_n) = 1 \quad \Rightarrow \quad \omega = \frac{2}{\lambda_n}.$$

   Metoda tedy konverguje pro $\omega \in (0, 2/\lambda_n)$.

2. Optimální ω je bod, ve kterém červeně vyznačená funkce dosahuje minima. Tento bod dostaneme jako průsečík funkcí f_1 a f_n:

   $$1 - \omega_{\text{opt}}\lambda_1 = -(1 - \omega_{\text{opt}}\lambda_n) \quad \Rightarrow \quad \omega_{\text{opt}} = \frac{2}{\lambda_1 + \lambda_n}.$$

Zjistili jsme tedy, že nejlepší konvergence dosáhneme, zvolíme-li $\omega_{\text{opt}} = \frac{2}{\lambda_1+\lambda_n}$. Konvergenční faktor bude v tomto případě

$$\rho_{\text{opt}} = 1 - \omega_{\text{opt}}\lambda_1 = 1 - \frac{2\lambda_1}{\lambda_1+\lambda_n} = \frac{\lambda_1+\lambda_n-2\lambda_1}{\lambda_1+\lambda_n} = \\ = \frac{\lambda_n-\lambda_1}{\lambda_n+\lambda_1}\frac{\frac{1}{\lambda_1}}{\frac{1}{\lambda 1}} = \frac{\frac{\lambda_n}{\lambda_1}-1}{\frac{\lambda_n}{\lambda_1}+1} = \frac{\kappa(\mathsf{A})-1}{\kappa(\mathsf{A})+1},$$

kde $\kappa(\mathsf{A}) = \lambda_n/\lambda_1$ je číslo podmíněnosti matice A.

Můžeme také odvodit, kolik iterací je třeba, abychom dosáhli požadované relativní změny normy rezidua. Hledáme tedy k, pro které platí

$$\frac{\|\boldsymbol{r}^k\|}{\|\boldsymbol{r}^0\|} \le \varepsilon, \qquad \text{tedy} \qquad \|\boldsymbol{r}^k\| \le \varepsilon\|\boldsymbol{r}^0\|$$

Využijme toho, že $\|\boldsymbol{r}^k\| \le \rho_{\text{opt}}^k\|\boldsymbol{r}^0\|$ a přepišme nerovnici na

$$\rho_{\text{opt}}^k\|\boldsymbol{r}^0\| \le \varepsilon\|\boldsymbol{r}^0\|.$$

Vykrácením normy a zlogaritmováním obou stran nerovnice dostanem řešení $k \ge \frac{\log\varepsilon}{\log\rho_{\text{opt}}}$ (nezapomeňte, že protože $\rho_{\text{opt}} \in (0,1)$, je třeba otočit znaménko nerovnosti).

1.1.4 Ukončovací podmínky

Při použití iteračního řešiče většinou nemáme předem zadaný počet iterací, které mají proběhnout. Chceme výpočet ukončit ve chvíli, kdy se s odhadem řešení dostaneme dostatečně blízko přesnému řešení. Vzhledem k tomu, že přesné řešení (tedy ani přesnou chybu v dané iteraci) neznáme, musíme si pomoci jinak.

Jednou z možností je ukončit cyklus ve chvíli, kdy se s novým odhadem řešení příliš nepohneme od předchozího odhadu (tzn. $\|\boldsymbol{x}^{k+1} - \boldsymbol{x}^k\| < \varepsilon$). Tato podmínka ale nijak nebere v potaz velikost prvků v matici soustavy a vektoru pravé strany (jiná situace nastane, pokud jsou prvky matice a vektoru v řádech tisíců, jiná pokud jsou v řádech tisícin). Proto je vhodné tuto podmínku zvolit relativně např. vzhledem k normě vektoru pravé strany (tzn. $\|\boldsymbol{x}^{k+1} - \boldsymbol{x}^k\| < \|\boldsymbol{b}\|\varepsilon$, tedy $\|\boldsymbol{x}^{k+1} - \boldsymbol{x}^k\|/\|\boldsymbol{b}\| < \varepsilon$).

Nejčastěji se ovšem k výpočtu ukončovací podmínky používá normy vektoru rezidua $\boldsymbol{r}^{k+1} = \boldsymbol{b} - \mathsf{A}\boldsymbol{x}^{k+1}$. To nám poskytuje přirozený odhad toho, jak dobře je splněna původní rovnice. Ukončovací podmínku lze tedy volit ve tvaru $\|\boldsymbol{b} - \mathsf{A}\boldsymbol{x}^{k+1}\| < \varepsilon$. Podobně jako v předchozím případě je i zde vhodnější použít relativní změnu rezidua oproti vektoru pravé strany ($\|\boldsymbol{b} - \mathsf{A}\boldsymbol{x}^{k+1}\|/\|\boldsymbol{b}\| < \varepsilon$) nebo počátečnímu reziduu ($\|\boldsymbol{b} - \mathsf{A}\boldsymbol{x}^{k+1}\|/\|\boldsymbol{b} - \mathsf{A}\boldsymbol{x}^0\| < \varepsilon$).

1.2 Gradientní iterační metody

Věta Řešení soustavy $\mathsf{A}\boldsymbol{x} = \boldsymbol{b}$ se symetrickou pozitivně definitní maticí A je ekvivalentní s minimalizací kvadratické formy

$$f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^T\mathsf{A}\boldsymbol{x} - \boldsymbol{b}^t\boldsymbol{x}.$$

Důkaz Dokažme nejdříve implikaci $\mathsf{A}\boldsymbol{x} = \boldsymbol{b} \Rightarrow \boldsymbol{x} = \arg\min_{\boldsymbol{v}\in\mathbb{R}^n} f(\boldsymbol{v})$. Podívejme se, jak se změní funkční hodnota f, posuneme-li se z bodu $\boldsymbol{x}$ o nějaký nenulový vektor $\boldsymbol{c}$:

$$\begin{aligned} f(\boldsymbol{p}) = f(\boldsymbol{x}+\boldsymbol{c}) &= \frac{1}{2}(\boldsymbol{x}+\boldsymbol{c})^T\mathsf{A}(\boldsymbol{x}+\boldsymbol{c}) - \boldsymbol{b}(\boldsymbol{x}+\boldsymbol{c}) = \\ &= \frac{1}{2}\boldsymbol{x}^T\mathsf{A}\boldsymbol{x} + \boldsymbol{c}^T\underbrace{\mathsf{A}\boldsymbol{x}}_{=\boldsymbol{b}} + \frac{1}{2}\boldsymbol{c}^T\mathsf{A}\boldsymbol{c} - \boldsymbol{b}^T\boldsymbol{x} - \boldsymbol{b}^T\boldsymbol{c} = \\ &= \underbrace{\frac{1}{2}\boldsymbol{x}^T\mathsf{A}\boldsymbol{x} - \boldsymbol{b}^T\boldsymbol{x}}_{=f(\boldsymbol{x})} + \underbrace{\boldsymbol{c}^T\boldsymbol{b} - \boldsymbol{b}^T\boldsymbol{c}}_{=0} + \frac{1}{2}\boldsymbol{c}^T\mathsf{A}\boldsymbol{c} = f(\boldsymbol{x}) + \underbrace{\frac{1}{2}\boldsymbol{c}^T\mathsf{A}\boldsymbol{c}}_{>0}. \end{aligned}$$

Díky pozitivní definitnosti A je výraz $\boldsymbol{c}^T\mathsf{A}\boldsymbol{c}$ kladný. Posuneme-li se tedy z bodu $\boldsymbol{x}$ v libovolném směru, hodnota funkce f se zvětší. V bodě $\boldsymbol{x}$ tedy nastává minimum.

K důkazu opačné implikace $\boldsymbol{x} = \arg\min_{\boldsymbol{v}\in\mathbb{R}^n} f(\boldsymbol{v}) \Rightarrow \mathsf{A}\boldsymbol{x} = \boldsymbol{b}$ je třeba si uvědomit nutnou podmínku minima funkce $f : \mathbb{R}^n \to \mathbb{R}$, tedy nulovost gradientu:

$$\boldsymbol{x} = \arg\min_{\boldsymbol{v}\in\mathbb{R}^n} f(\boldsymbol{v}) \Rightarrow \nabla f(\boldsymbol{x}) = \boldsymbol{o}. \tag{1.6}$$

Lze ukázat, že pro gradient funkce f platí

$$\nabla f(\boldsymbol{x}) = \left[\frac{\partial f}{\partial x_1}(\boldsymbol{x}), \ldots, \frac{\partial f}{\partial x_n}(\boldsymbol{x})\right]^T = \frac{1}{2}\mathsf{A}^T\boldsymbol{x} + \frac{1}{2}\mathsf{A}\boldsymbol{x} - b = \mathsf{A}\boldsymbol{x} - \boldsymbol{b}.$$

Z podmínky (1.6) tedy vyplývá $\mathsf{A}\boldsymbol{x} - \boldsymbol{b} = \boldsymbol{o}$. □

V případě symetrické pozitivně definitní matice $\mathsf{A} \in \mathbb{R}^{n\times n}$ máme tedy dvě možnosti, jak geometricky nahlížet na řešení soustavy lineárních rovnic. První přístup je chápat každou rovnici jako předpis nadroviny v n-rozměrném prostoru. Řešení soustavy pak odpovídá hledání průsečítu těchto rovin. Druhý přístup, který využijeme při odvozování následujících algoritmů, odpovídá minimalizaci příslušné pozitivně definitní kvadratické formy. Grafem pozitivně definitní kvadratické formy $f : \mathbb{R}^n \to \mathbb{R}$ je n-dimenzionální paraboloid, který má minimum (viz Obrázek 1.2 pro $n = 2$).

1.2.1 Metoda největšího spádu

Metoda největšího spádu je iterační metoda s předpisem

$$\boldsymbol{x}^{k+1} := \boldsymbol{x}^k + \alpha^k\boldsymbol{v}^k, \tag{1.7}$$

kde $\boldsymbol{v}^k$ volíme jako směr největšího poklesu funkce f. Všimněme si, že pro gradient platí $\nabla f(\boldsymbol{x}^k) = \mathsf{A}\boldsymbol{x}^k - \boldsymbol{b}$ a pro reziduum k-tém kroku $\boldsymbol{r}^k = \boldsymbol{b} - \mathsf{A}\boldsymbol{x}^k$. Tedy $\boldsymbol{r}^k = -\nabla f(\boldsymbol{x}^k)$. Protože gradient odpovídá směru největšího růstu funkce v daném bodě, reziduum je směr největšího spádu. Logicky, protože chceme dosáhnout minima dané funkce, vydáváme se v každém kroku ve směru rezidua, tedy $\boldsymbol{v}^k = \boldsymbol{r}^k$.

Otázkou je, jak daleko se v každém kroku v tomto směru vydat, tedy jak zvolit koeficient α^k. Metoda největšího spádu volí tento koeficient tak, aby v každém kroku dosáhla minima funkce f ve směru rezidua. Definujme si tedy pomocnou funkci $F : \mathbb{R} \to \mathbb{R}$:

$$\begin{aligned} F(\alpha) = f(\boldsymbol{x}^k + \alpha\boldsymbol{r}^k) &= \frac{1}{2}(\boldsymbol{x}^k + \alpha\boldsymbol{r}^k)^T\mathsf{A}(\boldsymbol{x}^k + \alpha\boldsymbol{r}^k) - \boldsymbol{b}^T(\boldsymbol{x}^k + \alpha\boldsymbol{r}^k) = \\ &= \frac{1}{2}(\boldsymbol{x}^k)^T\mathsf{A}\boldsymbol{x}^k + (\boldsymbol{x}^k)^T\mathsf{A}\boldsymbol{x}^k + \frac{1}{2}\alpha^2(\boldsymbol{r}^k)^T\mathsf{A}\boldsymbol{r}^k - \boldsymbol{b}^T\boldsymbol{x}^k - \alpha\boldsymbol{b}^T\boldsymbol{r}^k \end{aligned}$$

Hledáme α, ve kterém tato funkce dosahuje minima, její derivace se tedy musí rovnat nule:

$$F'(\alpha) = \alpha(\boldsymbol{r}^k)^T\mathsf{A}\boldsymbol{r}^k + (\boldsymbol{r}^k)^T \underbrace{\mathsf{A}\boldsymbol{x}^k}_{=\boldsymbol{r}^k(\boldsymbol{b}-\boldsymbol{r}^k)} - \boldsymbol{b}^T\boldsymbol{r}^k = \alpha(\boldsymbol{r}^k)^T\mathsf{A}\boldsymbol{r}^k - (\boldsymbol{r}^k)^T\boldsymbol{r} = 0$$

Odtud

$$\alpha^k = \frac{(\boldsymbol{r}^k)^T\boldsymbol{r}^k}{(\boldsymbol{r}^k)^T\mathsf{A}\boldsymbol{r}^k}. \tag{1.8}$$

Stejný předpis můžeme odvodit, použijeme-li místo pomocné funkce funkci f a položíme její derivaci ve směru $\boldsymbol{r}^k$ rovnu nule (vzpomeňme si, že platí $\frac{\mathrm{d}f(\boldsymbol{x})}{\mathrm{d}\boldsymbol{h}} = (\nabla f(\boldsymbol{x}))^T\boldsymbol{h}$):

$$\begin{aligned} \frac{\mathrm{d}f(x^{k+1})}{\mathrm{d}\boldsymbol{r}^k} &= \boldsymbol{o} \\ (\nabla f(\boldsymbol{x}^{k+1}))^T\boldsymbol{r}^k &= \boldsymbol{o} \\ (-\boldsymbol{r}^{k+1})^T\boldsymbol{r}^k &= \boldsymbol{o} \end{aligned}$$

Dosazením $\boldsymbol{r}^{k+1} = \boldsymbol{r}^k - \alpha^k\mathsf{A}\boldsymbol{r}^k$ do předchozí rovnice a jednoduchou úpravou dostaneme stejný předpis pro α^k jako v předchozím případě (1.8). Předchozí odvození nám také prozradilo důležitou vlastnost metody největšího spádu – každý směr $\boldsymbol{v}^k = \boldsymbol{r}^k$ je kolmý na předchozí směr. Jak brzy uvidíme, není to vždy žádaná vlastnost.

Algoritmus tedy počítá jednotlivé aproximace pomocí následujících předpisů:

$$\begin{aligned} \boldsymbol{r}^k &:= \boldsymbol{b} - \mathsf{A}\boldsymbol{x}^k = \boldsymbol{b} - \mathsf{A}(\boldsymbol{x}^{k-1} + \alpha^{k-1}\boldsymbol{r}^{k-1}) = \underbrace{\boldsymbol{b} - \mathsf{A}\boldsymbol{x}^{k-1}}_{\boldsymbol{r}^{k-1}} - \alpha^{k-1}\mathsf{A}\boldsymbol{r}^{k-1} = \\ &= \boldsymbol{r}^{k-1} - \alpha^{k-1}\mathsf{A}\boldsymbol{r}^{k-1} \\ \alpha^k &:= \frac{(\boldsymbol{r}^k)^T\boldsymbol{r}^k}{(\boldsymbol{r}^k)^T\mathsf{A}\boldsymbol{r}^k} \\ \boldsymbol{x}^{k+1} &:= \boldsymbol{x}^k + \alpha^k\boldsymbol{r}^k \end{aligned}$$

Díky úpravě předpisu pro výpočet $\boldsymbol{r}^k$ jsme ušetřili jedno násobení matice-vektor ($\mathsf{A}\boldsymbol{x}^k$) – výsledek $\mathsf{A}\boldsymbol{r}^{k-1}$ si totiž můžeme zapamatovat z předchozí iterace.

Ukažme si nyní, že metoda konverguje. Konvergenci budeme dokazovat v tzv. energetické normě $\|\cdot\|_A$, tedy normě indukované skalárním součinem $(\mathsf{A}\boldsymbol{x}, \boldsymbol{x}) = \boldsymbol{x}^T\mathsf{A}\boldsymbol{x} : \|\boldsymbol{x}\|_A = \sqrt{\boldsymbol{x}^T\mathsf{A}\boldsymbol{x}}$.

$$\begin{aligned}
\|\boldsymbol{e}^{k+1}\|_A &= (\boldsymbol{e}^{k+1})^T\mathsf{A}\boldsymbol{e}^{k+1} = (\boldsymbol{e}^k + \alpha^k\boldsymbol{r}^k)^T\mathsf{A}(\boldsymbol{e}^k + \alpha^k\boldsymbol{r}^k) = \\
&= (\boldsymbol{e}^k)^T\mathsf{A}\boldsymbol{e}^k + 2\alpha^k(\boldsymbol{r}^k)^T \underbrace{\mathsf{A}\boldsymbol{e}^k}_{=-\boldsymbol{r}^k} + (\alpha^k)^2(\boldsymbol{r}^k)^T\mathsf{A}\boldsymbol{r}^k = \\
&= \|\boldsymbol{e}^k\|_A^2 - 2\underbrace{\frac{(\boldsymbol{r}^k)^T\boldsymbol{r}^k}{(\boldsymbol{r}^k)^T\mathsf{A}\boldsymbol{r}^k}}_{=\alpha^k}(\boldsymbol{r}^k)^T\boldsymbol{r}^k + (\frac{(\boldsymbol{r}^k)^T\boldsymbol{r}^k}{(\boldsymbol{r}^k)^T\mathsf{A}\boldsymbol{r}^k})^2(\boldsymbol{r}^k)\mathsf{A}\boldsymbol{r}^k = \\
&= \|\boldsymbol{e}^k\|^2 - \frac{((\boldsymbol{r}^k)^T\boldsymbol{r}^k)^2}{(\boldsymbol{r}^k)^T\mathsf{A}\boldsymbol{r}^k} = \|\boldsymbol{e}^k\|_A^2\left(1 - \frac{((\boldsymbol{r}^k)^T\boldsymbol{r}^k)^2}{((\boldsymbol{r}^k)^T\mathsf{A}\boldsymbol{r}^k)((\boldsymbol{e}^k)^T\mathsf{A}\boldsymbol{e}^k)}\right)
\end{aligned}$$

References

[1] Trefethen, L. N, Bau, D. Numerical Linear Algebra. SIAM. 1997.

[2] Schewchuk, J. R. An Introduction to the Conjugate Gradient Method Without the Agonizing Pain. 1994. Dostupné z `https://www.cs.cmu.edu/~quake-papers/painless-conjugate-gradient.pdf`

[3] Lukáš, D. Zápisky z přednášek. Dostupné z `https://homel.vsb.cz/~luk76`

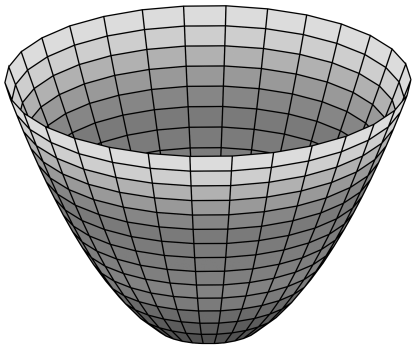

Figure 1.2: Graf kvadratické formy s pozitivně definitní maticí A (zdroj Wikipedia)