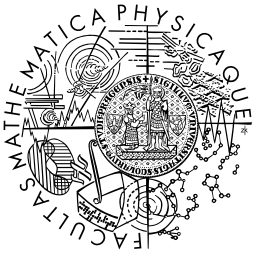


MATEMATICKO-FYZIKÁLNÍ FAKULTA
Univerzita Karlova

BAKALÁŘSKÁ PRÁCE

Ilona Riegerová

Neúplné faktorizace pro řešení problému nejmenších čtverců

Katedra numerické matematiky

Vedoucí bakalářské práce: prof. Ing. Miroslav Tůma, CSc.
Studijní program: Matematika
Studijní obor: Obecná matematika

Praha 2018

Prohlašuji, že jsem tuto bakalářskou práci vypracoval(a) samostatně a výhradně s použitím citovaných pramenů, literatury a dalších odborných zdrojů.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona v platném znění, zejména skutečnost, že Univerzita Karlova má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle §60 odst. 1 autorského zákona.

V ..... dne ..... Podpis autora

Název práce: Neúplné faktorizace pro řešení problému nejmenších čtverců

Autor: Ilona Riegerová

Katedra: Katedra numerické matematiky

Vedoucí bakalářské práce: prof. Ing. Miroslav Tůma, CSc., Katedra numerické matematiky

Abstrakt: Problém nejmenších čtverců (dále jen LS problém) je aproximační úloha řešení soustavy lineárních rovnic. Tato matematicko-statistická metoda patří k nejzákladnějším úlohám numerické lineární algebry a má mnoho aplikací v přírodovědných a inženýrských problémech, jako jsou například molekulární struktury, zpracování signálu, geodetika, tomografie a další. Práce je zaměřena na přehled současných technik řešení LS problému a jeho variace pro rozsáhlé úlohy. V první kapitole je popsána známá teorie a přímé řešiče pro obecné husté matice. V případě LS problému s velkou řídkou maticí se využívají iterační metody, které jsou urychlovány nepřesnými maticovými rozklady. Těm je věnována druhá kapitola, konkrétně pak metodě sdružených gradientů předpodmíněné neúplnou Choleského faktorizací. V numerických experimentech je demonstrován vliv různé řídkosti řádků matice na stabilitu a časovou i výpočetní náročnost úlohy.

Klíčová slova: neúplné faktorizace, problém nejmenších čtverců, předpodmíněné iterační metody

Title: Incomplete factorizations for solving the least squares problem

Author: Ilona Riegerová

Department: Department of Numerical Mathematics

Supervisor: prof. Ing. Miroslav Tůma, CSc., Department of Numerical Mathematics

Abstract: The Least Squares problem (LS problem) is a task of finding an approximate solution of linear systems. This mathematical-statistical method is considered as one of the most fundamental tasks of numerical linear algebra and it has a wide range of applications in science and engineering problems, such as molecular structures, signal processing, geodesy, tomography and many more. Focus of this paper is on the overview of current techniques for solving the LS problem and its variations for large problems. The first chapter describes the known theory and direct solvers for general dense matrices. In the case of the LS solution with a large sparse matrix, iterative methods are used and accelerated by incomplete matrix decompositions. The second chapter is therefore dedicated to this area, namely the method of conjugate gradients preconditioned by the incomplete Cholesky factorization is studied. Numerical experiments demonstrate the effect of rows with different densities on stability, time and computational cost of the task.

Keywords: incomplete factorizations, the least squares problem, preconditioned iterative methods

Na tomto místě bych ráda poděkovala vedoucímu své bakalářské práce panu prof. Ing. Miroslavu Tůmovi, CSc. za cenné rady, ochotu a trpělivost, kterou mi po celý čas věnoval. Dále bych chtěla poděkovat své rodině za veškerou podporu během studia.

Obsah

Úvod

Jedním z nejzákladnějších úkolů lineární algebry je řešení soustavy lineárních rovnic. V praxi tyto soustavy vznikají z mnoha velmi různých aplikací, a proto jsou i jejich vlastnosti značně odlišné. Obecně jsou ale velké a většinou zatížené chybami, které mohou být důsledkem nepřesného měření nebo konečné aritmetiky. Toto přirozeně vede k metodě nejmenších čtverců (dále jen LS problém, z anglického termínu *Least Squares Problem*), neboť právě ona poskytuje řešení soustavy lineárních rovnic, kde vektor pravé strany je dán nepřesně. Pro LS problém existuje jednoznačné řešení, které minimalizuje normu residua, tj.

$$x^{LS} = \underset{x}{\operatorname{argmin}} \|Ax - b\|_2 .$$

V první kapitole této práce je odvozena základní teorie LS problému. Dále pak obsahuje přímé metody řešení pro velké husté problémy. Ačkoliv jsou předně rozebrány tři nejpodstatnější metody, tj. QR rozklad, soustavy normálních rovnic a SVD rozklad, pozornost je věnována i ostatním méně významným metodám. U všech těchto metod jsou shrnuty jejich možné aplikace, výhody i nevýhody a numerické vlastnosti.

Úlohy pracující s reálnými daty naráží na paměťovou a výpočetní náročnost výpočtů. Proto není možné používat ve všech případech přímých řešičů a odvozují se specializované metody pro řešení soustav s velkou řídkou maticí. V druhé kapitole se ukazuje, že iterační metody jsou efektivnější než řídké varianty přímých řešičů, a ideálních výsledků se může dosáhnout nejlépe předpodmíněnou iterační metodou. Z analýzy jednotlivých přístupů je vyzdvihnuta projekční metoda sdružených gradientů pro LS problém a její předpodmínění neúplným Choleského rozkladem. Právě ona je využita v numerických experimentech třetí kapitoly.

Na maticích z lineárního programování jsou testovány dva druhy neúplné Choleského faktorizace, tzv. IC(τ) a IC(*level*). IC(τ) odpovídá neúplné Choleského faktorizaci, která nuluje prvky pod danou tolerancí τ, zatímco IC(l) označuje neúplnou Choleského faktorizaci s předem předepsanou řídkou strukturou. První experiment ilustruje závislost počtu iterací na velikosti tolerance τ. Druhý experiment znázorňuje rozdíl mezi jednotlivými předpodmíněními IC(l). Konkrétně pak porovnává závislost počtu iterací metody PCGLS a počtu nenulových prvků ve faktoru L vzhledem k hodnotě parametru úrovně l.

1. Úvod do problému nejmenších čtverců a jeho řešení

Řešení soustav lineární rovnic je jedním z nejstarších a nejzákladnějších problémů, se kterým se numerická matematika potýká. Je známo, že soustava reprezentována v maticovém tvaru $Ax = b$, kde A je čtvercová regulární matice, má vždy jednoznačné řešení. V případě, že předpoklad není splněn, nemusí již takové tvrzení obecně platit. Tato práce se zabývá případem, kdy není soustava dána přesně nebo nemá pěkné vlastnosti.

1.1 Zavedení pojmů

Buď $Ax = b$ soustava lineárních rovnic. Bez újmy na obecnosti buď $m \geq n$, kde $m, n \in \mathbb{N}$. Na obecnou matici $A \in \mathbb{R}^{m\times n}$ lze pohlížet jako na matematický model. Vektor $b \in \mathbb{R}^m$ pravé strany odpovídá naměřeným hodnotám a vektor $x \in \mathbb{R}^n$ původním datům splňujícím soustavu. Problém nejmenších čtverců (dále jako LS problém, z anglického termínu *Least Squares*) se zabývá v praxi vzniklou úlohou $Ax \approx b$, která zohledňuje nepřesnosti v měření, a tedy zatížení vektoru b chybami. Nalezením v normě minimální změny f pravé strany se úloha změní na rovnost $Ax = b + f$. Odtud je odvozeno řešení aproximační úlohy $Ax \approx b$ ve smyslu nejmenších čtverců, tj. takové, které minimalizuje normu residua r definovaného předpisem $r = b - Ax$.

Definice 1 (Problém nejmenších čtverců)**.** *Buď $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$. Problémem nejmenších čtverců je nazývána úloha určení vektoru $x \in \mathbb{R}^n$ takového, že platí*

$$\min_{x,f} \|f\| \qquad \textit{za podmínky} \qquad Ax = b + f.$$

Poznámka. Analogicky lze LS problém definovat i pro komplexní čísla. Tento text se ale zaměří pouze na reálný případ LS problému.

Za stejných předpokladů na matici A a vektor b lze LS problém formulovat jako určení vektoru $x \in \mathbb{R}^n$ řešícího

$$\min_x \|Ax - b\|_2 , \tag{1.1}$$

kde $\|\cdot\|_2$ značí Eukleidovskou vektorovou normu. Definice 1 odpovídá odvození metody v této práci, ale formulace (1.1) je v jiných textech častější a pro další úpravy vhodnější.

Existence a jednoznačnost řešení LS problému je rozebrána v následujícím.

Tvrzení 1. *Nechť množina všech řešení problému nejmenších čtverců (1.1) je definována předpisem*

$$\mathcal{S} = \{x \in \mathbb{R}^n | \, \|Ax - b\|_2 = min\}.$$

Potom vektor $x \in \mathcal{S}$ právě tehdy, když platí tato ortogonalita

$$A^T(b - Ax) = 0. \tag{1.2}$$

Důkaz. Nechť nejdřív $\hat{x}$ splňuje ortogonalitu (1.2), tj. $A^T\hat{r} = 0$, kde $\hat{r} = b - A\hat{x}$. Potom pro $x \in \mathbb{R}^n$ platí

$$r = b - Ax = b - A\hat{x} + A\hat{x} - Ax = \hat{r} + A(\hat{x} - x) \equiv \hat{r} + Ae.$$

Umocněná tato rovnost, tj.

$$r^T r = (\hat{r} + Ae)^T(\hat{r} + Ae) = \hat{r}^T\hat{r} + \|Ae\|_2^2,$$

je minimální pro $x = \hat{x}$. Tedy libovolné $x \in \mathbb{R}^n$ je LS řešení, tj. $x \in \mathcal{S}$, pokud splňuje ortogonalitu (1.2). Zbývá dokázat obrácenou implikaci. Nechť ortogonalita (1.2) neplatí pro $\hat{x}$, tj. $A^T\hat{r} = z \neq 0$. Buď $x = \hat{x} + \varepsilon z$. Potom

$$r = \hat{r} + A(\hat{x} - x) = \hat{r} + A\hat{x} - Ax = \hat{r} + A\hat{x} - A(\hat{x} + \varepsilon z) = \hat{r} - \varepsilon Az.$$

Z umocnění této rovnosti, tj.

$$r^T r = (\hat{r} - \varepsilon Az)^T(\hat{r} - \varepsilon Az) = \hat{r}^T\hat{r} - 2\varepsilon z^T z + \varepsilon^2(Az)^T Az < \hat{r}^T\hat{r},$$

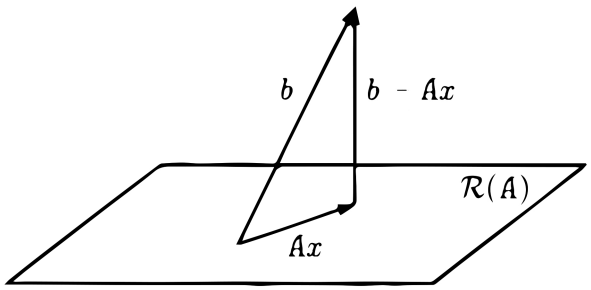
plyne nerovnost $r^T r < \hat{r}^T\hat{r}$ pro dostatečně malé ε, a tedy $\hat{x}$ není LS řešení, tj. $x \notin \mathcal{S}$.

□

Z Tvrzení 1 plyne, že residuální vektor $r = b - Ax$ je prvkem jádra $\mathcal{N}(A^T)$, a tedy LS řešení x jednoznačně rozkládá pravou stranu $b = Ax + r$ do dvou ortogonálních komponent

$$Ax \in \mathcal{R}(A), \qquad r \in \mathcal{N}(A^T),$$

viz Obrázek **??**. Neboť nejlepší aproximace vektoru b je prvkem $\mathcal{R}(A)$, potom musí být jednoznačně určena ortogonální projekcí $b|_{\mathcal{R}(A)}$ vektoru b na prostor $\mathcal{R}(A)$.



Obrázek 1.1: Geometrická interpretace LS problému pro $n = 2$.

Z těchto úvah je formulována následující věta shrnující existenci LS řešení.

Věta 1 (Existence LS řešení)**.** *Nechť $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$. Vektor x je řešením problému nejmenších čtverců právě tehdy, když*

$$Ax = b|_{\mathcal{R}(A)}, \qquad \|b - Ax\| = \|b|_{\mathcal{N}(A^T)}\|.$$

Důkaz. Důkaz lze nalézt v knize [9].

□

Sloupcová hodnost $rank(A)$, resp. lineární závislost sloupců $a_1, a_2, \ldots, a_n$, matice A, zaštiťuje jednoznačnost LS řešení. Buď $rank(A) = n$, potom lze $b|_{\mathcal{R}(A)}$ vyjádřit jednoznačně lineární kombinací sloupců A, a tedy existuje právě jedno LS řešení. To je triviální právě tehdy, když $b \perp \mathcal{R}(A)$. Jinak, tj. buď $rank(A) < n$, jednoznačné LS řešení neexistuje, neboť matice A má netriviální jádro $\mathcal{N}(A)$. V takovém případě se za LS řešení uvažuje to minimální v normě. Lze snadno dokázat, že takové existuje pouze jedno.

Věta 2 (Jednoznačnost LS řešení)**.** *Nechť $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$. Potom existuje právě jedno řešení x problému nejmenších čtverců minimální v normě, které je dáno vztahy*

$$Ax = b|_{\mathcal{R}(A)} \qquad a \qquad x \in \mathcal{R}(A^T).$$

Důkaz. Důkaz lze nalézt v knize [9].

□

Doposud byl chybami zatížen pouze vektor b. Často je však matice A dána nepřesně, neboť se i ona v praxi získává jako výsledek nějakého měření. Úloha pak spočívá v nalezení v normě nejmenší změny E matice A a nejmenší změny f pravé strany b tak, aby vektor x splňoval

$$(A + E)x = b + f.$$

Tato úloha se nazývá Úplný problém nejmenších čtverců (zkráceně TLS z anglického výrazu *Total Least Squares*) a je definována následovně.

Definice 2 (Úplný problém nejmenších čtverců)**.** *Buď $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$. Úplným problémem nejmenších čtverců je nazývána úloha určení vektoru $x \in \mathbb{R}^n$ řešícího*

$$\min_{x,E,f} \|[f,E]\|_F \qquad \textit{za podmínky} \qquad (A + E)x = b + f.$$

kde $\|\cdot\|_F$ značí Frobeniovu normu.

Euklidovská norma vektoru a Frobeniova norma matice jsou unitárně invariantní, a tedy přenásobením rovnic

$$Ax = b + f, \qquad \text{nebo} \qquad (A + E)x = b + f$$

unitární maticí U se normy $\|f\|$ a $\|[f,E]\|_F$ nezmění. Proto jsou LS problém a TLS problém unitárně invariantní úlohy. Pokud je matice A špatně podmíněná, neposkytuje metoda TLS problému dobré výsledky a je nutné ji modifikovat na tzv. *Úplné nejmenší čtverce s omezením hodnosti*. Problému úplných nejmenších čtverců se ale dále text věnovat nebude.

1.2 Standardní metody LS řešení pro husté matice

Metod řešení LS problému (1.1) je mnoho a jejich využití a vhodnost se liší v přímé závislosti na matici A, ale i na faktorech jako jsou například časová a paměťová náročnost, složitost výpočtu nebo jeho robustnost. Nechť má v následujícím matice A plnou sloupcovou hodnost, tj. buď $rank(A) = n$, s ohledem na existenci a jednoznačnost LS řešení vyloženou v podkapitole 1.1. Než se přikročí k rozboru tří nejpoužívanějších metod, tj. řešení pomocí soustavy normálních rovnic, QR rozkladu a SVD rozkladu, je vhodné se zmínit o těch metodách, které nejsou tolik běžně využívané.

V první řadě je to charakterizace LS problému pracující s rozšířenou maticí

$$\begin{pmatrix} I & A \\ A^T & 0 \end{pmatrix} \begin{pmatrix} y \\ x \end{pmatrix} = \begin{pmatrix} b \\ c \end{pmatrix}. \tag{1.3}$$

Tato čtvercová, symetrická soustava, která je indefinitní pro $A \neq 0$ a nesingulární pro $rank(A) = n$, podmiňuje řešení primárního i duálního LS problému

$$\min_x \{\|Ax - b\|_2^2 + 2c^T x\}, \tag{1.4}$$

$$\min_y \|y - b\|_2^2, A^T y = c. \tag{1.5}$$

Volbou $c = 0$ v (1.4) se obdrží LS problém (1.1). Takto zformulovaná úloha může být velmi obtížně řešitelná, ale v některých případech může být užitečná.

Další přístupy jsou založeny například na Gaussově eliminaci s částečnou pivotací a uplatňují se při práci s nesymetrickou lineární soustavou. Jako příklad poslouží Petersova-Wilkinsonova metoda, která je postavena na LU rozkladu s částečnou pivotací.

$$\Pi_1 A \Pi_2 = \begin{pmatrix} A_1 \\ A_2 \end{pmatrix} = LU = \begin{pmatrix} L_1 \\ L_2 \end{pmatrix} U,$$

kde $L_1 \in \mathbb{R}^{n \times n}$ je jednotková dolní trojúhelníková a $U \in \mathbb{R}^{n \times n}$ je nesingulární horní trojúhelníková. Buď $\hat{x} = \Pi_2^T x$, $\hat{b} = \Pi_1 b$, pak je místo LS problému (1.1) řešena analogická úloha

$$\min_y \left\| Ly - \hat{b} \right\|_2, \qquad U\hat{x} = y.$$

Díky pivotaci je matice L dobře podmíněná a dále lze postupovat pomocí soustavy normálních rovnic, která má tvar

$$L^T L y = L^T \hat{b}.$$

V tomto případě nedochází ke ztrátě přesnosti jako u $A^T A x = A^T b$, viz podkapitola 1.2.1, ale výpočet vyžaduje $n^2(m - \frac{1}{3}n)$ operací, což je dražší než metoda soustavy normálních rovnic aplikovaná přímo na LS problém (1.1). Je však nutno podotknout, že LU rozklad je jen podmíněně zpětně stabilní, neboť může dojít k nechtěnému nárůstu prvků v matici U.

1.2.1 Soustava normálních rovnic

Z ortogonality (1.2) snadnou úpravou plyne, že LS řešení vyhovuje soustavě normálních rovnic

$$A^T Ax = A^T b. \tag{1.6}$$

Konzistence (1.6) je zřejmá z rovnosti $A^T b \in \mathcal{R}(A^T) = \mathcal{R}(A^T A)$. Matice $A^T A \in \mathbb{R}^{n\times n}$ je regulární a symetrická. Protože sloupce matice A jsou lineárně nezávislé, je matice $A^T A$ pozitivně definitní. Tedy jednoznačné LS řešení x a odpovídající residuální vektor r lze vyjádřit pomocí inverze matice ve tvaru

$$x = (A^T A)^{-1} A^T b \qquad a \qquad r = b - A(A^T A)^{-1} A^T b.$$

Prvním krokem metody soustavy normálních rovnic je generovat matici $C = A^T A \in \mathbb{R}^{n\times n}$ a vektor $d = A^T b \in \mathbb{R}^n$. Existuje dvojí reprezentace, tzv. vnější a vnitřní součinový tvar. Vnitřní součinový tvar vyjadřuje prvky matice C a vektoru d pomocí sloupců $a_1, a_2, \ldots, a_n$ matice A předpisem

$$\begin{aligned} c_{jk} &= a_j^* a_k, \qquad 1 \le j \le k \le n, \\ d_j &= a_j^* b, \qquad\quad 1 \le j \le n. \end{aligned}$$

Tento tvar není vhodný, pokud pracujeme s řídkými maticemi ze sekundárního úložiště, kvůli potřebě procházet jednotlivé sloupce vícekrát. Vnější součinový tvar zachází s řádky matice A, tedy k průchodu dat dojde jen jednou. Uložit stačí pouze vektor d a horní nebo dolní trojúhelník matice C. Buď $\tilde{a}_i$ i-tý řádek matice A, potom vnější součinový tvar je dán předpisem

$$C = \sum_{i=1}^{m} \tilde{a}_i \tilde{a}_i^*, \qquad d = \sum_{i=1}^{m} b_i \tilde{a}_i.$$

Dále lze postupovat více způsoby, např. Choleského metodou nebo sdruženými gradienty. Pro účely práce je příhodnější Choleského metoda, která je založena na Choleského rozkladu $C = L^T L$ a následném výpočtu dvou trojúhelníkových systémů

$$L^T z = d \qquad a \qquad Lx = z.$$

Definice 3 (Choleského rozklad)**.** *Pro každou hermitovskou pozitivně definitní matici A existuje jednoznačný rozklad*

$$A = LL^*,$$

kde L je dolní trojúhelníková matice s kladnými prvky na diagonále.

Choleského faktorizace je bezpodmínečně zpětně stabilní algoritmus nevyžadující pivotace. Avšak lze ji modifikovat, a poté použít pro řešení LS problému s maticí neúplné hodnosti. Totiž pokud $rank(A) < n$, potom je pivotace Choleského faktorizace nutná. Určení matice C a jejího Choleského rozkladu vyžaduje $mn^2 + \frac{1}{3}n^3$ operací, vektor d a substituce řešící trojúhelníkové systémy $mn + n^2$ operací. Celková náročnost metody odpovídá $\frac{1}{2}mn^2 + \frac{1}{6}n^3 + O(mn)$ krokům. Ačkoliv Choleského rozklad je bezpodmínečně stabilní, zaokrouhlovací chyby výpočtu

se vážou na podmíněnost soustavy, která v případě této metody vzroste kvadraticky, neboť $\kappa(A^TA) = [\kappa(A)]^2$. Velké ztrátě přesnosti řešení u špatně podmíněné matice se lze jen částečně vyhnout, a to díky zaokrouhlování na nižší řády během generování soustavy a při samotném výpočtu. Proto metoda soustav normálních rovnic patří k rychlejším, ale nikoliv ke zpětně stabilním algoritmům. Bezpečně lze tuto metodu použít pouze, platí-li

$$[\kappa(A)]^2 \ll \frac{1}{\varepsilon}.$$

Za předpokladu $m \gg n$, pro n malé, je výhodné tuto metodu zvolit, neboť výpočet pracuje s malou maticí $A^TA \in \mathbb{R}^{n\times n}$. Obecně se ale matice A^TA explicitně nepočítá, místo toho se využívá iteračních metod. Jako příklad postačí metoda *CGLS*, *CGNR* nebo *CGNE*.

Poznámka. Soustavu normálních rovnic lze získat i násobení AA^T. Toho je možné využít v případě, že $n \gg m$, pro m malé.

1.2.2 QR rozklad

Díky vlastnosti unitárních transformací zachovávat Eukleidovskou délku lze LS problém (1.1) matematicky ekvivalentně přepsat jako

$$\min_x \left\| Q^T(Ax - b) \right\|_2,$$

kde $Q^T \in \mathbb{R}^{m\times m}$ je unitární matice. Tomuto se říká unitární invariantnost LS problému. Využitím QR rozkladu matice $A = QR$ se LS problém (1.1) převede na úlohu snadněji řešitelnou

$$\min_x \|b - Ax\| = \min_x \|b - QRx\| = \min_x \left\| Q^Tb - Rx \right\| = \left\| (Q_m^\perp)^Tb \right\|. \tag{1.7}$$

Definice 4 (QR rozklad)**.** *Buď $A \in \mathbb{R}^{m\times n}$ obecná obdélníková matice. Rozklad tvaru*

$$A = QR,$$

kde $Q \in \mathbb{R}^{m\times m}$ je matice s ortonormálními sloupci, tj. $Q^TQ = I$, a $R \in \mathbb{R}^{m\times n}$ má všechny prvky pod hlavní diagonálou nulové, tj. $R = [r_{i,j}], r_{i,j} = 0$ pro $i > j$, se nazývá QR rozklad matice A.

Poslední rovnost (1.7) plyne z následujících úvah. Buď matice Q zapsána sloupcově jako $Q = [q_1, \ldots, q_m, q_{m+1}, \ldots, q_n] = [Q_m, Q_m^\perp]$ a buď $\hat{R} \in \mathbb{R}^{n\times n}$ regulární horní trojúhelníkový blok R. Potom platí

$$Q^Tb - Rx = \begin{bmatrix} Q_m^Tb \\ (Q_m^\perp)^Tb \end{bmatrix} - \begin{bmatrix} \hat{R} \\ 0 \end{bmatrix} x = \begin{bmatrix} Q_m^Tb - \hat{R}x \\ (Q_m^\perp)^Tb \end{bmatrix}.$$

Tedy pouze m prvků vektoru $Q^Tb - Rx$ je ovlivněno vektorem x, proto ekvivalentní řešení minimalizující normu rezidua splňuje soustavu

$$\hat{R}x = Q_m^Tb. \tag{1.8}$$

Odtud plyne, že provádět explicitní výpočet QR rozkladu je zbytečné. Stačí totiž generovat matici $\hat{R}$ a vektor Q_m^Tb. Následující tvrzení popisuje, jak získat LS řešení z QR rozkladu matice A.

Tvrzení 2. *Buď $A \in \mathbb{R}^{m\times n}$, $m \geq n$, $b \in \mathbb{R}^m$ a $c \in \mathbb{R}^n$ dáno. Buď $rank(A) = n$ a QR rozklad $A = QR$. Potom řešení rozšířeného systému*

$$\begin{pmatrix} I & A \\ A^T & 0 \end{pmatrix} \begin{pmatrix} y \\ x \end{pmatrix} = \begin{pmatrix} b \\ c \end{pmatrix}$$

lze vypočítat z

$$z = R^{-T}c, \qquad \begin{pmatrix} d_1 \\ d_2 \end{pmatrix} = Q^T b,$$
$$x = R^{-1}(d_1 - z), \qquad y = Q \begin{pmatrix} z \\ d_2 \end{pmatrix}.$$

Důkaz. Lze nalézt v knize [2].

□

Volbou $c = 0$, $r = y = b - Ax$ platí, že řešení LS problému (1.1) je určeno jako

$$\begin{pmatrix} d_1 \\ d_2 \end{pmatrix} = Q^T b, \qquad Rx = d_1, \qquad r = Q \begin{pmatrix} 0 \\ d_2 \end{pmatrix}.$$

Za unitární transformace, které jsou principem metody, se volí Householderovy reflexe, Givensovy rotace nebo Gram-Schmidtův ortogonalizační proces. Při výběru je nutné zohledňovat možnou ztrátu ortogonality. Výpočet pomocí Householderových reflexí zajišťuje ztrátu ortogonality na strojové přesnosti, přesněji je měřena jako $\left\|Q^TQ - I\right\|$. Určení R faktoru Householderovými reflexemi, resp. Givensovými rotacemi vyžaduje $n^2(m - \frac{1}{3}n)$, resp. $2n^2(m - \frac{1}{3}n)$ operací, což je srovnatelné s metodou soustavy normálních rovnic popsané v podkapitole 1.2.1. Tedy pro $m = n$ jsou metody stejně pracné, ale v případě $m \gg n$ je QR metoda dvojnásobně náročná. U Gram-Schmidtova procesu je nutno se přiklánět k tzv. modifikované verzi algoritmu (dále jen MGS), u které má ztráta ortogonality předvídatelnější chování a je úměrná $\kappa(A)\varepsilon$. Přesto ale výpočet řešení x z Tvrzení 2 kalkulující s maticí Q_1, u které dochází ke ztrátě ortogonality, nedává přesné řešení. Aplikováním MGS na matici (A, b) a rozkladem

$$(A, b) = (Q_1, q_{n+1}) \begin{pmatrix} R & z \\ 0 & \rho \end{pmatrix}$$

lze přesné LS řešení x vyjádřit rovnicemi

$$Rx = z, \qquad r = \rho q_{n+1}.$$

Takto se získá zpětně stabilní algoritmus, a dokonce se numerickými experimenty ukazuje, že jde i o lehce přesnější algoritmus výpočtu než jiné unitární metody, snad kvůli necitlivosti na řádkové permutace.

Řešení LS problému pomocí QR rozkladu se používá, má-li matice A dobře určenou plnou sloupcovou hodnost, tj.

$$\frac{\sigma_1(A)}{\sigma_n(A)} \ll \frac{1}{\varepsilon},$$

kde $\sigma_1(A)$ a $\sigma_n(A)$ značí největší a nejmenší singulární číslo matice A a ε je strojová přesnost. Zjevně R faktor z QR rozkladu má stejná singulární čísla a pravé singulární vektory jako matice A. Z rovnosti

$$A^TA = (R^T 0)Q^TQ\begin{pmatrix}R\\0\end{pmatrix} = R^TR$$

plyne, že má-li R faktor kladné prvky na diagonále, potom se rovná Choleskému faktoru matice A^TA. Následující výpočet potvrzuje ekvivalenci řešení soustavy normálních rovnic s řešením soustavy (1.8), tedy

$$\begin{aligned}A^TAx = A^Tb &\Leftrightarrow R^TRx = R^TQ^Tb\\ &\Leftrightarrow \hat{R}^T\hat{R}x = \hat{R}^T\hat{Q}_m^Tb\\ &\Leftrightarrow \hat{R}x = Q_m^Tb.\end{aligned}$$

1.2.3 SVD rozklad

Singulární rozklad matice je nejmocnějším nástrojem pro řešení LS problému, neboť nevyžaduje plnou sloupcovou hodnost matice A, která byla doposud nutnou podmínkou ke všem předchozím úvahám. Tedy singulární rozklad pomáhá řešit obecný LS problém, jak je popsáno v následující větě.

Věta 3. *Buď následující předpis obecný LS problém*

$$\min_{x\in\mathcal{S}} \|x\|_2, \qquad \mathcal{S}= \{x\in\mathbb{R}^n \,|\, \|Ax-b\|_2 = \min\},$$

kde $A\in\mathbb{R}^{m\times n}$ a $rank(A) = r \leq \min(m,n)$. Tato úloha má vždy jednoznačné řešení, které za použití ekonomického SVD rozkladu matice $A = U_r\Sigma_rV_r^*$ má tvar

$$x = V_r\begin{pmatrix}\Sigma_r^{-1} & 0\\ 0 & 0\end{pmatrix}U_r^*b = A^\dagger b,$$

kde $A^\dagger$ je Mooreova-Penroseova pseudoinverze.

Věta vychází z ekonomického SVD rozkladu a jeho aplikace na podmínky řešení minimálního v normě z Věty 2, tj. $Ax = b|_{\mathcal{R}(A)}$ a $x\in\mathcal{R}(A^*)$. Dosazením $b|_{\mathcal{R}(A)} = U_rU_r^*b$ a úpravou přejde úloha na tvar

$$V_r^*x = \Sigma_r^{-1}U_r^*b.$$

V dalším kroku se vyjádří řešení x jako lineární kombinace ortonormální báze $\mathcal{R}(A^*)$, tj. $x = V_ry$, kde souřadnice $y = V_r^*x = \Sigma_r^{-1}U_r^*b$.

Singulární čísla matice A se rovnají kladným odmocninám z vlastních čísel matice A^TA nebo matice AA^T. U malých hodnot singulárních čísel může vést generování těchto matic ke značné ztrátě přesnosti, proto tento přístup neposkytuje stabilní metodu výpočtu SVD rozkladu. Stabilní algoritmus pro SVD rozklad je možné získat, pokud se matice A převede na horní bidiagonální tvar. Ten se získá například aplikací Lanczosova procesu, nicméně efektivnější je nasazení SVD algoritmu na matici R z QR rozkladu matice A. Pak platí

$$A = Q\begin{pmatrix}U_r\\0\end{pmatrix}\Sigma V^T.$$

QR rozklad a Householderova redukce na bidiagonální tvar vyžaduje $n^2(m - \frac{1}{3}n)$ a $\frac{4}{3}n^3$ kroků, tedy tento proces modifikace vyžaduje $n^2(m + n)$ kroků. Na rychlejší algoritmus přišla dvojice Demmel-Kahan. Jejich výpočet je speciální verze QR rozkladu, díky které mohou být všechna singulární čísla bidiagonální matice spočtena s vysokou přesností. Jedná se o spojení tzv. posunutého QR algoritmu a stabilní implementace s nulovým posunutím. Značnou výhodou je rychlost výpočtu, neboť nulové posunutí využívá pouze okolo třetiny operací v porovnání s klasickým posunutím. Starší metody výpočtu SVD rozkladu jsou tzv. Jacobiho metody. Ty jsou sice pomalejší než QR algoritmus, ale dokáží spočítat singulární čísla přesněji než jakákoliv jiná metoda založená na transformaci na bidiagonální tvar.

1.2.4 Metody řešení pro špatně podmíněné matice s neúplnou sloupcovou hodností

Jeden z přístupů k řešení LS problému s maticí A, pro kterou platí $rank(A) < n$, byl popsán v podkapitole 1.2.3. Tento případ však stojí za detailnější rozebrání.

Je nutné si uvědomovat váhu zaokrouhlovacích chyb a fundamentální vliv hodnosti matice. Pro malou matici chyby E a matici A s neúplnou sloupcovou hodností sice nejpravděpodobněji platí, že $rank(\hat{A}) = rank(A + E) = n$, kde $\hat{A} = A + E$ je perturbovaná matice, ale $\hat{A}$ je velmi blízko maticím s neúplnou hodností, a měla by se proto označovat jako matice s neúplnou numerickou hodností (z anglického výrazu *numerically rank-deficient matrix*). Takto nadhodnocená hodnost matice může vést k řešení, které má velikou normu, neboť výpočet pracuje s hodnotami singulárních čísel velmi blízkými nule.

Buď pro tuto podkapitolu

$$\min_x \|Ax - b\|_2$$

LS problém s maticí A špatně podmíněnou, která nemá plnou sloupcovou hodnost. Pro takovou matici bude minimální residuum stále veliké. Proto se uvažuje tzv. zkrácené SVD řešení (dále jako TSVD řešení, z anglického výrazu *truncated SVD solution*), které dokáže udržet residuum příhodně malé. Pro dané δ a

$$k = min\{rank(B) | \, \|A - B\|_2 \leq \delta\}$$

platí, že k je numerická sloupcová hodnost matice A. Dále buďte singulární čísla $\sigma_i = 0$ pro $i > k$ a SVD rozklad matice A buď tvaru $A = U\Sigma V^T = \sum_{i=1}^{n} u_i \sigma_i v_i^T$. Odpovídající TSVD řešení potom splňuje

$$x = \sum_{i=1}^{k} \frac{c_i}{\sigma_i} v_i, \qquad c = U^T b.$$

Tedy TSVD řešení odpovídá příbuznému LS problému

$$\min_x \|A_k x - b\|_2 , \qquad A_k = \sum_{i=1}^{k} u_i \sigma_i v_i^T ,$$

kde A_k je nejlepší aproximace matice A hodnosti k z Eckart-Young-Mirského věty.

SVD rozklad je nejspolehlivější nástroj k určení numerické hodnosti, ale je velmi pracný. Proto je častější méně výpočetně náročný QR rozklad se sloupcovou pivotací, který může mít i stejně dobré vlastnosti jako SVD rozklad. Buď P permutační matice a

$$AP = Q_1 \begin{bmatrix} R_{1,1} \\ R_{1,2} \end{bmatrix}$$

QR rozklad matice AP. Sloupce matice Q_1 tvoří ortonormální bázi oboru hodnot $\mathcal{R}(A)$ matice A a je možné ho použít pro řešení LS problému. Potom problém minimalizace $\left\| b - \tilde{A}\tilde{x} \right\|_2$, kde $\tilde{A} = AP$ a $\tilde{x} = P^T x$, je ekvivalentní LS problému (1.1). LS řešení v obecném případě pomocí QR rozkladu splňuje

$$\begin{bmatrix} R_{1,1} & R_{1,2} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \tilde{x}_1 \\ \tilde{x}_2 \end{bmatrix} = \begin{bmatrix} Q_1^T b \\ Q_2^T b \end{bmatrix} \equiv \begin{bmatrix} c \\ d \end{bmatrix}, \tag{1.9}$$

kde $\tilde{x}_1 \in \mathbb{R}^r$, $\tilde{x}_2 \in \mathbb{R}^{m-r}$. Norma residua je minimální pro $R_{1,1}\tilde{x}_1 + R_{1,2}\tilde{x}_2 = c$. Z regularity $R_{1,1}$ plyne, že množina všech LS řešení problému (1.9) je tvaru

$$\tilde{x}(w) = \begin{bmatrix} R_{1,1}^{-1}(c - R_{1,2}w) \\ w \end{bmatrix},$$

kde $w \in \mathbb{R}^{m-r}$ je vektor volných parametrů. Potom množina LS řešení problému (1.1) obsahuje taková řešení splňující

$$x(w) = P\tilde{x}(w), \qquad \|b - Ax(w)\|_2 = \|d\|_2 .$$

Tedy tato metoda nalezla LS řešení, obecně ale není minimální v normě. Řešení minimální v normě je obtížně určitelné. Proto se za něj považuje to s nulovým vektorem w, jež je dobrou aproximací LS řešení.

2. Problém velkých řídkých matic

V současné době se v praxi pracuje s velkým množstvím dat, proto je záhodno se i v teorii zabývat úlohami s velkými maticemi. Tím jsou myšleny matice o milionech a více řádcích. Výpočty s takovými maticemi jsou nejen paměťově náročné ale i komplikovaně proveditelné v takovém měřítku. Naštěstí velká část matic má tu vlastnost, že většina jejich prvků je nulová. Tyto matice se nazývají řídké. Otázkou stále zůstává, která matice už je řídká, a která ne. To není snadné definovat, ačkoliv se pojem řídkosti zdá býti intuitivní. Matematik J. H. Wilkinson definoval v [16] řídkou matici jako jakoukoliv matici s dostatečným počtem nulových prvků, že se vyplatí jejich nulovost využít. To v praxi vede na výrazně větší efektivitu algoritmů, a nebo tato vlastnost dokonce podmiňuje řešitelnost. Příkladem takto datově rozsáhlých úloh mohou být problémy geodetiky, molekulární struktury, tomografie, fotogrammetrie nebo gravitačního pole Země. Prvním podstatným pravidlem při výpočtech s velkou řídkou maticí je ukládání pouze nenulových prvků a druhým logicky plynoucím je počítání jenom s těmito prvky. Tedy prominentní snahou výpočtů by mělo být zabránit zaplnění matice, tj. vznikání nových nenulových prvků.

V následujících úvahách nechť má matice A plnou sloupcovou hodnost, tedy buď $rank(A) = n$. LS problém s velkou řídkou maticí A, tj.

$$\min_x \|Ax - b\|_2, \qquad A \in \mathbb{R}^{m\times n}, \qquad m \geq n, \tag{2.1}$$

bude dále nazýván řídký LS problém (z anglického výrazu *Sparse Least Squares*). Řídký LS problém lze řešit přímými metodami, jako jsou metoda normálních rovnic nebo QR rozklad, nebo iteračními metodami. Často se během výpočtu uchyluje k předpodmíněným iteračním metodám, které stojí mezi těmito dvěma přístupy řešení. Kritériem volby mezi metodami je jak ukládání dat, tak i následná řídkost operací. Je zřejmé, že SVD rozklad ztrácí pro řídký LS problém na svém smyslu, neboť je časově náročný, a proto nepoužitelný.

2.1 Metody řešení LS problému pro velké řídké matice pomocí rozkladů

Metody řešení LS problému pro velké řídké matice pomocí rozkladů jsou typicky založeny na faktorizaci matice A a následném řešení systému využívající faktorizace a trojúhelníkových řešičů. V první řadě je třeba nalézt vhodnou permutaci, která zajistí minimální zaplnění během výpočtu, čímž zefektivní numerický výpočet samotné faktorizace. K nalezení takové permutace se v praxi využívá heuristických přístupů, neboť rozhodovací verze tohoto problému je tzv. *NP-úplný problém*. Po tomto kroku se může analyzovat eliminační strom, řídká struktura a další klíčové vlastnosti faktorizace, ale jsou i jednodušší postupy. Hlubšímu studiu analýzy, předcházející numerické faktorizaci a přímým metodám pro řídké problémy, se věnuje například text [7]. Dále budou rozebrány nejvýznamější rozklady, resp. jejich řídké varianty.

2.1.1 QR faktorizace

V případě QR rozkladu nedochází ani ke kvadratickému růstu čísla podmíněnosti ani k explicitnímu vyjadřování matic A^TA, A^Tb, neboť pracuje přímo s maticí A, a proto bývají ztráty na přesnosti a zpětné stabilitě menší. Řešení řídkého LS problému pomocí QR rozkladu má tři hlavní úskalí, neboť dochází k zaplnění během mezivýpočtů, neúplný QR rozklad se provádí složitě a navíc je nesnadné jej následně zkombinovat s iterační metodou. Householderovy reflexe, na kterých QR rozklad většinou stojí, vyžadují kvůli zaplnění jistou modifikaci. Aplikací těchto nejefektnější ortogonálních transformací na posloupnost malých hustých podproblémů se lze nechtěnému zaplnění vyhnout. Řádkové permutace mohou pozitivně ovlivnit počet operací nutných ke QR rozkladu, proto jsou zařazeny do algoritmu.

Algoritmus (QR faktorizace)**.**

Data: A, b
Result: $P_c, R\ faktor$
Nalézt vhodnou charakterizaci struktury matice A ;
Určit sloupcovou permutaci P_c pro dosažení řídkého R faktoru QR rozkladu matice $P_c^TAP_c$;
Výpočet R faktoru matice $P_c^TAP_c$;
Určit vhodnou řádkovou permutaci P_r a generovat matici P_rAP_c ;
Výpočet R faktoru matice P_rAP_c a vektoru $d = P_rb$ pomocí ortogonálních transformací ;
Řešení systému $Ry = d$, kde pak LS řešení je dáno rovností $x = P_cy$;

Nejvýhodnější adaptací QR rozkladu pro řídké matice je takzvaný Multifrontální QR rozklad (dále jen jako MQR rozklad, z anglického termínu *Multifrontal QR Decomposition*), který může být významně méně časově náročný za cenu nízkého nárůstu paměti potřebné k ukládání výpočtu. Podstata MQR rozkladu spočívá v přetvoření přímého rozkladu řídké matice na posloupnost částečných rozkladů malých hustých matic a za použití Householderových reflexí modifikování pro QR rozklad. Vzniklé husté podproblémy mohou být řešeny paralelně a i lépe využívají vyrovnávací paměť počítače. Tímto se snižuje časová náročnost výpočtu.

Přístup k řešení pomocí QR rozkladu se hodí na problémy s maticí, která nemá plnou sloupcovou hodnost. Zde se však musí dávat pozor na sloupcové permutace, které nezachovávají řídkost. Stále ale platí, že řídký QR rozklad je drahý, neboť vyžaduje obecně větší počet operací, navíc dochází k většímu zaplnění. Tím je myšlena jak časová, tak i výpočetní náročnost. V některých případech se odhad zaplnění může lišit od skutečnosti. Konkrétním příkladem jsou úlohy s maticí, která nemá silnou Hallovu vlastnost.

Definice 5 (Silná Hallova vlastnost)**.** *Buď $A \in \mathbb{R}^{m\times n}$, $m \geq n$ má silnou Hallovu vlastnost, pokud pro každou podmnožinu k sloupců, $0 < k < m$, má odpovídající matice nenulové prvky alespoň v $k + 1$ řádcích.*

Pro takové matice bude odhad zaplnění přemrštěný.

2.1.2 Choleského faktorizace

Choleského faktorizace je hlavním pilířem metody řešení LS problému pomocí soustavy normálních rovnic, která je vyložena v podkapitole 1.2.1.

Algoritmus (Choleského faktorizace)**.**

Data: A, b
Result: $P_c, Lfaktor$
Nalézt vhodnou charakterizaci struktury matice A^TA ;
Určit sloupcovou permutaci P_c pro dosažení řídkého Choleského faktoru L matice $P_c^TA^TAP_c$;
Výpočet Choleského faktorizace matice $P_c^TA^TAP_c$ (neúplný, stačí ukládat strukturu matice L) ;
Generovat matici $C = P_c^TA^TAP_c$ a vektor $d = P_c^TA^Tb$ (numericky) a ukládat matici C do struktury matice L ;
Výpočet Choleského faktoru L a řešení dvou systémů $L^Tz = d$ a $Ly = z$, kde pak LS řešení je dáno rovností $x = P_cy$;

Charakterizace struktury z prvního kroku by měla například zohledňovat pivotace během rozkladu. Ačkoliv Choleského rozklad je bezpodmínečně zpětně stabilní algoritmus nutně nevyžadující pivotace, permutační matice P_c v druhém kroku zajistí, že nedojde ke zbytečnému zaplnění Choleského faktoru L, a tedy algoritmus bude nadále pracovat s řídkou maticí. Stále platí závěry o numerické stabilitě této metody uvedené v podkapitole 1.2.1, tzn. číslo podmíněnosti A^TA roste kvadraticky a explicitně zformulované matice A^TA a A^Tb zapříčiňují ztrátu přesnosti díky zaokrouhlování. Tedy pro dobře podmíněné matice dává tato metoda uspokojivě přesné výsledky, ale bohužel pro špatně podmíněné matice je vhodnější využít metodu QR rozkladu. Ta se sice potýká se zaplněním a špatně se kombinuje s iteračními metodami, ale zaručuje lepší numerickou stabilitu než Choleského faktorizace. Při zacházení se špatně podmíněnou maticí, která je však blízko dobře podmíněným, se může vyplatit použít metodu soustavy normálních rovnic modifikovanou na iterační.

Tato řídká přímá metoda není vhodná například v inverzních problémech nebo v úlohách vznikajících z rekonstrukčních problémů, viz [2]. Takové úlohy jsou popsány maticí A s náhodnou řídkou strukturou, tj. prvek a_{ij} je nenulový s pravděpodobností $p < 1$. Potom platí, že prvek $(A^TA)_{ij}$ je nenulový s pravděpodobností $q = 1-(1-p^2)^m \approx 1-e^{-mp^2}$. Takže matice A^TA bude nejspíše hustá, když průměrný počet nenulových prvků ve sloupci je okolo $m^{1/2}$.

Na druhou stranu, může Choleského faktorizace poskytovat výborné neúplné rozklady na rozdíl od QR rozkladu. Podkapitola 2.3 se věnuje neúplným variantám těchto rozkladů, kterých se využívá při předpodmiňování.

2.2 Iterační metody

Iterační metody, tedy metody aproximující přesné řešení pomocí postupného vylepšování počáteční aproximace, jsou výhodným nástrojem, pokud matice A je

reprezentována nikoliv svou strukturou, ale svým vlivem na vektor. Tedy například funkcí f_A, která vektoru v přiřadí součin Av. Na druhou stranu postrádají iterační metody na robustnosti a bývají často úzce specializované. Tím je myšleno, že například v případě symetrického pozitivně definitního lineárního systému lze aplikovat v zásadě libovolnou iterační metodu a to na systém normálních rovnic $A^TAx = A^Tb$, avšak musí být zajištěna vhodná implementace. Explicitní formulaci matice A^TA se lze vyhnout využitím ortogonality (1.2). Stejně tak mohou být iterační metody aplikovány na charakterizaci LS problému s rozšířenou maticí (1.3), jenomže pouze ty, které jsou vhodné pro symetrické indefinitní úlohy. Shrnuto, iteračních metod se využívá, když je přesný rozklad příliš drahý nebo jej nelze vůbec sestrojit, nebo pokud není matice A explicitně k dispozici. V praxi pak iteračním metodám předchází předpodmínění za účelem zrychlení konvergence. V případě, že je matice reprezentována svým vlivem na vektor, nastává s předpodmíněním problém, neboť k němu je matice A potřeba. V knize [2] lze nalézt předpodmiňovací postupy, které se touto komplikací zabývají. Iterační metody se dají rozdělit do dvou skupin, tedy stacionární metody a projekční metody.

2.2.1 Stacionární iterační metody

Stacionární iterační metody, často nazývané klasické, jsou odvozeny na principu štěpení matice A, tj.

$$\begin{aligned} &(M+N)x = b \\ &x = M^{-1}(b - Nx) = M^{-1}(b + Mx - Ax) = x + M^{-1}(b - Ax), \end{aligned} \tag{2.2}$$

kde M a N jsou matice vhodně voleny, ale M je vždy regulární a snadno invertovatelná. Samotný iterační proces odvozený z (2.2) je pak obecně tvaru

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

Stacionární iterační metody tvaru

$$Mx^{(k+1)} = Nx^{(k)} + b \qquad k = 0,1,\cdots \tag{2.3}$$

patří k nejzákladnějším řešičům soustavy normálních rovnic. Jejich tvar (2.3) odpovídá štěpení $A^TA = M - N$, kde M je regulární a jí příslušný systém lineárních rovnic je snadno řešitelný. V textu [6] se nachází důkaz, že pokud A má plnou hodnost, pak pro každou konzistentní iterační metodu tvaru (2.3) existuje štěpení takové, že metoda a její iterační matice se dají vyjádřit jako

$$x^{(k+1)} = M^{\dagger}(Nx^{(k)} + b), \qquad G = I - C^{-T}A^TA,$$

pro iterační matici G, kde C je nějaká regulární matice. Odtud je zřejmé, že obecná iterační metoda (2.3) pro LS problém je ekvivalentní s Richardsonovou metodou prvního řádu aplikovanou na lineární systém $C^{-T}A^T(Ax - b) = 0$, kde C je regulární. Ke stacionárním metodám vhodným pro řešení soustav normálních rovnic patří Jacobiho a Gaussova-Seidelova metoda, metoda SOR (z anglického termínu *Successive Overrelaxation method*) nebo její symetrická varianta, tj. SSOR (z anglického termínu *Symetric Successive Overelaxation method*). O konvergenci stacionárních iteračních metod a konkrétních příkladech je více například v knize [9].

2.2.2 Projekční iterační metody

Do druhé skupiny iteračních metod, která je v současné praxi mnohem významnější, patří projekční metody. Jejich smyslem je projektovat původní úlohu na jiný podprostor tak, aby ideálně vznikla menší úloha, lépe řešitelná a hlavně dobře aproximující tu původní v co nejmenším počtu kroků. Tyto metody konstruují posloupnost aproximací řešení x_k takových, že $x_k \in x_0 + \mathcal{S}_k$ a $r_k \perp \mathcal{C}_k$, kde $\mathcal{S}_k \subseteq \mathbb{R}^n$ je k-dimenzionální podprostor nazýván *prostor aproximace* (z anglického termínu *Search space*) a $\mathcal{C}_k \subseteq \mathbb{R}^n$ je k-dimenzionální podprostor nazýván *prostor podmínek* (z anglického termínu *Constraint space*). Předpoklad kolmosti residua je standardní volbou a nazývá se *Petrov-Galerkinova podmínka*. Rostou-li dimenze prostorů $\mathcal{S}_k$ a $\mathcal{C}_k$, pak projekční metoda nalezne řešení původní úlohy nejpozději v n krocích, protože pak platí, že $\mathcal{C}_n = \mathbb{R}^n$, což znamená $r_n = 0$ z podmínky na ortogonalitu a $x_n = x$. Veškeré úvahy se snaží o takové metody, které optimalizují aproximace řešení ve smyslu minimality chyby v nějaké normě a jsou založeny na krátkých rekurencích. Důležitým výsledkem je Faber-Manteuffelova věta, která popisuje skupinu matic, pro které jsou tyto dvě vlastnosti možné. Bohužel je tato skupina úzká a obecně se musí mezi optimalitou a krátkými rekurencemi volit.

Krylovovské iterační metody jsou základní projekční metody, které projektují původní úlohu na Krylovův podprostor. Aproximace řešení x_k se potom nalézají v odpovídající varietě $x_0 + \mathcal{K}_k$, kde $\mathcal{K}_k$ značí Krylovův podprostor. Základem pro efektivitu těchto metod je tedy generování vhodné báze Krylovova podprostoru, k čemuž dobře slouží Arnoldiho algoritmus pro nesymetrické matice a Lanczosův algoritmus pro symetrické matice. Krylovovské metody se liší volbou prostorů $\mathcal{S}_k$ a $\mathcal{C}_k$, jejich stručný přehled je shrnut v Tabulce 2.1. Pro tuto práci jsou však nejpodstatnější sdružené gradienty, o kterých je více v podkapitole 2.2.3. Doporučeným zdrojem pro studium Krylovovských metod je kniha [13].

Krylovovská metoda	**Prostor $\mathcal{S}_k$**	**Prostor $\mathcal{C}_k$**
CG, FOM	$\mathcal{K}_k(A, r_0)$	$\mathcal{K}_k(A, r_0)$
GMRES	$\mathcal{K}_k(A, r_0)$	$A\mathcal{K}_k(A, r_0)$
CGLS, LSQR	$\mathcal{K}_k(A^TA, A^Tr_0)$	$\mathcal{K}_k(AA^T, r_0)$
CGNR	$\mathcal{K}_k(A^TA, A^Tb)$	$\mathcal{K}_k(A^TA, A^Tb)$

Tabulka 2.1: Volby Krylovových podprostorů pro příslušné Krylovovské metody.

2.2.3 Sdružené gradienty pro řešení LS problému

Řídký LS problém lze iteračně řešit pomocí metody CGLS (z anglického termínu *Conjugate Gradients for Least Squares*), která aplikuje sdružené gradienty na soustavu normálních rovnic A^TA. Právě metoda CGLS bude využita v numerických experimentech této práce. Následující odstavec se proto věnuje krátké teorii o CG sdružených gradientech.

Metoda CG je mocným nástrojem pro řešení soustav lineárních rovnic s velkými řídkými maticemi, protože se jedná o paměťově nenáročným algoritmus.

Konkrétně se v každé iteraci uchovávají pouze čtyři vektory, a navíc každá iterace provádí jedno násobení matice s vektorem, kde typicky matice A není explicitně dána. Z následující věty plyne, že konvergence CG závisí na velikosti projekcí normalizovaného počátečního residua do invariatních podprostorů určených vlastními vektory a na rozložení vlastních čísel matice A. Důkladnějšímu rozboru CG, komplikovaných provázaností s mnoha dalšími objekty a samotné konvergenci CG se věnuje kniha [13]. Pro podstatnost následujícího výsledku, jeho netrivialitu i podceňovanost v praxi je uveden i s důkazem.

Věta 4 (Konvergence CG a závislost na rozložení vlastních čísel)**.** *Buď $A \in \mathbb{R}^{n\times n}$ symetrická pozitivně definitní matice. Potom metoda sdružených gradientů CG pro soustavu $Ax = b$ konverguje nejpozději v n-té iteraci. Navíc pro A-normu k-té chyby $e^{(k)}$ platí*

$$\|e^{(k)}\|_A = \|r_0\| \min_{p\in\pi_k} \left(\sum_{j=1}^{n} \frac{|\omega_j|^2}{\lambda_j} p(\lambda_j)^2 \right)^{1/2}, \tag{2.4}$$

a tedy odhad relativní A-normy chyby je tvaru

$$\frac{\|e^{(k)}\|_A}{\|e^{(0)}\|_A} \leq \min_{p\in\pi_k} \max_{i=1,\ldots,n} |p(\lambda_i)|. \tag{2.5}$$

Důkaz. Metoda CG se zastaví, pokud $\|r_k\| = 0$, neboť to je ekvivalentní rovnosti $x_k = x$, kde x je přesné řešení. A protože $r_k \perp \mathcal{C}_k$ a $\mathcal{C}_n = \mathbb{R}^n$, pak $\|r_k\| = 0$ je splněno nejvýše v n krocích.

Chybu aproximace v k-tém kroku $e^{(k)}$ lze psát ve tvaru

$$e^{(k)} \equiv x - x_k = (x - x_0) - (x_k - x_0) = (x - x_0) - q(A)r_0 = e^{(0)} - Aq(A)e^{(0)} = p(A)e^{(0)},$$

kde $q(\cdot)$ je vhodný polynom stupně nejvýše $k-1$ odpovídající příslušné varietě a $p(x) \equiv 1 - xq(x)$ je polynom stupně nejvýše k takový, že $p(0) = 1$. Z minimality A-normu k-té chyby $e^{(k)}$ platí pro π_k množinu všech polynomů stupně nejvýše k takový, že $p(0) = 1$ rovnost

$$\|e^{(k)}\|_A = \min_{p\in\pi_k} \|p(A)e^{(0)}\|_A. \tag{2.6}$$

Buď $A = Q\Lambda Q^*$ spektrální rozklad matice A, $QQ^* = I$ a $\Lambda = diag(\lambda_1, \ldots, \lambda_n)$. Dále buď $\omega = [\omega_1, \ldots, \omega_n]^T \equiv Q^* \dfrac{r_0}{\|r_0\|}$. Potom

$$\begin{aligned} \|p(A)e^{(0)}\|_A^2 &= \|p(A)A^{1/2}e^{(0)}\|^2 = \|p(A)A^{-1/2}r_0\|^2 \\ &= \|r_0\|^2 \|p(\Lambda)\Lambda^{1/2}\omega\|^2 = \|r_0\|^2 \sum_{j=1}^{n} \frac{|\omega_j|^2}{\lambda_j} p(\lambda_j)^2, \end{aligned}$$

odkud přímo plyne rovnost (2.4).

Odhad (2.5) se získá snadnou úpravou z (2.6), spektrálního rozkladu matice A a nerovnosti

$$\begin{aligned} \|p(A)e^{(0)}\|_A &= \|p(A)A^{1/2}e^{(0)}\| \leq \|p(A)\| \|A^{1/2}e^{(0)}\| = \|p(Q\Lambda Q^*)\| \|e^{(0)}\|_A \\ &= \|Qp(\Lambda)Q^*\| \|e^{(0)}\|_A = \|p(\Lambda)\| \|e^{(0)}\|_A = \max_{i=1,\ldots,n} |p(\lambda_i)| \|e^{(0)}\|_A. \end{aligned}$$

□

Poznámka. Zde je vhodné připomenout, že konvergence nejvýše v n krocích platí obecně pro každou iterační metodu, ale jejich podstatou je, aby docílili dostatečně přesné aproximace výrazně dřív.

Díky teorii lze CG přímo aplikovat na soustavu normálních rovnic, avšak to vede na numericky nestabilní algoritmus a je tedy nutná implementační variace standardního algoritmu pro CG. Algoritmus CGLS byl původně odvozen M. R. Hestenesem a E. Stiefelem v [11] a je podrobně rozebrán v článku [3]. V každé iteraci se jednou násobí Ap_k a jednou $A^T r_k$. Ukládají se dva n-složkové vektory x, p a dva m-složkové vektory r, q. Pro účely této práce bude použita předpodmíněná varianta CGLS, která je k nalezení například v knize [2].

Algoritmus (PCGLS)**.**

Data: A, b, x_0, M, tol
Result: x_k

$r^{(0)} := b - Ax^{(0)}$;
$s^{(0)} = p^{(0)} := M^{-T}(A^T r^{(0)})$;
$\gamma_0 := \left\|s^{(0)}\right\|^2$;

for $k = 0, 1, 2 \ldots$ **do**
  while $\gamma_k > tol$ **do**
    $t^{(k)} = M^{-1} p^{(k)}$;
    $q^{(k)} = A t^{(k)}$;
    $\alpha_k = \gamma_k / \left\|q^{(k)}\right\|^2$;
    $x^{(k+1)} = x^{(k)} + \alpha_k t^{(k)}$;
    $r^{(k+1)} = r^{(k)} - \alpha_k q^{(k)}$;
    $s^{(k+1)} = M^{-T}(A^T r^{(k)})$;
    $\gamma_{k+1} = \left\|s^{(k+1)}\right\|^2$;
    $\beta_k = \gamma_{k+1} / \gamma_k$;
    $p^{(k+1)} = s^{(k+1)} + \beta_k p^{(k)}$;
  end
end

Pokud má matice k různých singulárních čísel, potom metoda CGLS konverguje v přesné aritmetice nejvýše v k krocích, a navíc pokud jsou tato singulární čísla nenulová, CGLS spočítá přesné řešení z pseudoinverze $A^\dagger b$. Z tohoto důvodu je CGLS značně efektivní pro matice s malou hodností. V případě zaokrouhlovacích chyb je proces konvergence mnohem složitější.

Za zmínku pak stojí matematicky ekvivalentní a numericky stabilní metoda LSQR od Ch. C. Paige a M. Saunderse, která nevyžaduje žádné restriktivní podmínky na matici A. LSQR je založená na Lanczosově bidiagonalizaci, kterou generuje stejnou posloupnost aproximací x_k jako CGLS. V každé iteraci dochází k násobení matice A a A^T s vektorem, tedy není potřeba mít matici A explicitně k dispozici. LSQR je numericky spolehlivější než CGLS, pokud je zapotřebí více iterací a matice A je špatně podmíněná.

Metoda	Výpočetní náročnost	Paměťová náročnost
CGLS	$2m + 3n$ násobení	ukládají se 2 m-složkové a 2 n-složkové vektory
LSQR	$3m + 5n$ násobení	ukládají se 2 m-složkové a 3 n-složkové vektory

Tabulka 2.2: Srovnání počtu operací a zabíraného místa v paměti pro metody CGLS a LSQR.

2.3 Předpodmínění neúplným rozkladem

Předpodmíněním se myslí proces transformace matice A a celé soustavy lineárních rovnic maticí M aproximující A, pro kterou se nově vzniklý systém bude snadněji řešit. Matice M se nazývá *předpodmiňovací matice* nebo *předpodmiňovač*. Jelikož se jedná o aproximaci, přesné řešení původní úlohy, tj. $Ax = b$, se získává iteračním procesem. Ten v každé iteraci řeší předpodmíněnou úlohu $Mz = w$ s vhodně volenými pravými stranami a limitně se jeho řešení blíží přesnému. Zjevně volba předpodmiňovací matice M je rozhodujícím krokem. Řád konvergence úlohy je přímo závislý na rozložení vlastních čísel předpodmíněné matice. Malé číslo podmíněnosti je v tomto ohledu často považováno za dobré znamení konvergence. Úvahy shrnuté v sekcích 2.1.1, 2.1.2 a 2.2 vedou na kombinaci rozkladu a iterační metody, neboť z nich vyplývá, že preferovanějším řešičem řídkého LS problému jsou iterační metody. Avšak k dosažení rychlé konvergence je potřeba dobrého předpodmínění.

Protože LS problém vzniká z velmi odlišných aplikací, může vyžadovat pokaždé jiné předpodmínění. Vždy ale platí, že předpodmínění vztažené na LS problém je ekvivalentní transformaci proměnných. Matematicky zapsáno, transformace odpovídá problémům

$$\min_y \left\|(AM^{-1})y - b\right\|_2, \; Mx = y \qquad nebo \qquad \min_x \left\|(M^{-1}Ax - M^{-1}b\right\|_2 .$$

První předpis představuje tzv. předpodmínění zprava, které je vhodné pro nesymetrické matice, a druhý předpis je tzv. předpodmínění zleva. Numerické experimenty v kapitole 3 budou předpodmiňovány zleva. Explicitní vyjádření součinů $AM^{-1}y$ a M^TA^Tr obejde iterační metoda díky operátorové reprezentaci zmíněné v podkapitole 2.2. Předpodmínění nám tedy umožní řešit snadnější úlohu, avšak přibude práce s výpočtem dalšího systému rovnic $Mx = y$. Shrnuto, předpodmiňovač M musí splňovat tři podmínky:

- M je dostatečně řídký.
- Soustavy rovnic s M a případně s M^T jsou snadno řešitelné.
- Iterační řešič aplikovaný na transformovanou soustavu konverguje rychleji.

2.3.1 Předpodmínění neúplným LU rozkladem

Velké množství řešení řídkých lineárních systémů je založeno na LU rozkladu, protože nedochází k tak velkému nechtěnému zaplnění nenulovými prvky, jako například u QR rozkladu. Proto se text dále zaměří na algebraická předpodmínění aplikovaná na velkou řídkou matici. Přesněji řečeno na neúplné rozklady, které budou chápány jako aproximace přesného LU rozkladu obcházející zaplnění faktorizace.

Definice 6 (LU rozklad)**.** *Nechť $A \in \mathbb{C}^{m\times m}$ je regulární matice. Rozklad tvaru*

$$A = LU,$$

kde L je dolní trojúhelníková matice s jednotkovou diagonálou a U je horní trojúhelníková matice, je nazýván LU rozkladem matice A.

Poznámka. LU rozklad se dá v zásadě ztotožňovat s jedním z nejzákladnějších algoritmů, tj. Gaussovou eliminací, neboť právě tohoto algoritmu je výstupem. Je nutné mít na paměti, že samotná Gaussova eliminace nemusí být numericky stabilní algoritmus, a proto se uvažuje s částečnou pivotací. LU rozklad lze definovat i pro matice singulární nebo obdélníkové.

Text [5] se věnuje ILU (z anglického termínu *Incomplete LU factorization*) a MILU (z anglického termínu *Modified Incomplete LU factorization*) předpodmiňovačům aplikovaným na matici A. Podstata těchto předpomínění tkví v umožnění zaplnění nenulovými prvky během Gaussovy eliminace, avšak na předem předepsaných pozicích LU faktorizace, nebo na pozicích získaných v průběhu rozkladu. Jednoduchou modifikací se z ILU předpodmínění dostane MILU, které se hodí na numerické řešení některých jednoduchých eliptických problémů parciálních diferenciálních rovnic. Obecně je přístup LU rozkladu založen na řádkové perturbaci matice A a předpodmíněním zprava, tj.

$$PA = \begin{pmatrix} A_1 \\ A_2 \end{pmatrix}, \qquad \min_y \left\| (AA_1^{-1})y - b \right\|_2, \qquad A_1 x = y.$$

Předpodmiňovač A_1^{-1} lze použít i pro předpodmínění soustavy normálních rovnic, neboť

$$(PA)^T(PA) = I_n + (A_2A_1^{-1})^T(A_2A_1^{-1}),$$

což je snadněji řešitelná úloha. Na takovým způsobem transformovaný LS problém lze aplikovat například stacionární metodu SOR, a pokud navíc je k dispozici dobře podmíněný L faktor matice A_1 z úplného LU rozkladu, lze LS problém řešit Petersovou-Wilkinsonovou metodou zmíněnou v úvodu kapitoly 1.2.

2.3.2 Předpodmínění soustavy normálních rovnic

Pravděpodobně nejtradičněji se předpodmiňuje soustava normálních rovnic neúplným rozkladem symetrické positivně definitní matice A^TA, nejčastěji neúplným Choleského rozkladem. Hlavním důvodem pro předpodmínění je fakt, že systémová normální matice A^TA je nutně hustší než matice A. To může způsobit, že iterační metoda nemusí konvergovat. Hlavním důvodem volby neúplné

Choleského faktorizace je její jednoduchost a praktičnost navzdory horším numerickým vlastnostem, které jsou rozebrány v kapitole 1.2.1 a 2.1.2. Článek [4] srovnává čtyři neúplné rozklady sloužící jako předpodmiňovače pro iterační metodu CGLS aplikované na normální matici A^TA. Konkrétně BIF (z anglického termínu *Balanced Incomplete Factorization*), IC (z anglického termínu *Incomplete Cholesky Factorization*), AINV (z anglického termínu *Approximate Inverze*) a CIMGS (z anglického termínu *Cholesky Incomplete Modified Gram-Schmidt*). Ukazuje se, že BIF je z těchto předpodmiňovačů nejrobustnější a lze jej modifikovat tak, aby se vyhnul tzv. breakdownu pro symetrické pozitivně definitní matice. Na jejím začátku stojí posunutý $(I - (A^TA)^{-1})^{-1}$-bikonjugovaný proces aplikovaný na soustavu normálních rovnic, odkud se získá LDL^T rozklad, ale zároveň i inverzní Choleského faktor. Z numerických experimentů v [4] plyne, že BIF předpodmiňovač vyžaduje obecně více iterací stejně tak jako IC, který bývá navíc citlivý na breakdown. Dále pak je vidět, že BIF stojí svou náročností na konskrukci mezi IC a AINV. Přístup pomocí AINV je vhodný pro velmi řídké úlohy. Pokud se u CIMGS přistoupí na větší zaplnění, pak takový předpodmiňovač může zajistit velmi rychlou konvergenci iterační metody.

Přepis ortogonality (1.2) pomocí předpodmiňovací matice dává tvar soustavy normálních rovnic ve faktorech

$$M^{-T}A^T(AM^{-1}y - b) = M^{-T}A^T(Ax - b) = 0.$$

To znamená, že diskuse o konvergenci, která je uvedena ve Větě 4, musí být založena na tomto předpodmíněném tvaru soustavy.

V numerických experimentech této práce v kapitole 3 je předpodmínění iterační metody založeno na Choleského faktorizaci matice A^TA, protože pro úplnou verzi Choleského faktorizaci při zanedbání zaokrouhlovacích chyb platí, že $\kappa(AM^{-1}) = 1$ a tedy ke konvergenci iterační metody stačí teoreticky jedna iterace. Obecně, předpodmiňovací matice M, které aproximují Choleského L faktor (a M^{-1} aproximující L^{-1} faktor) jsou velmi efektivní. Existuje mnoho různých algoritmů pro neúplnou Choleského faktorizaci. Každý z nich se liší například zaplněním, volbou indexů pro nenulové prvky nebo složitostí výpočtu. Rozboru některých z nich se věnuje kapitola 3.

2.4 Matice s řádky různé řídkosti

Pokud matice A je řídká s jen jedním hustým řádkem, potom už je matice soustavy normálních rovnic $C = A^TA$ nutně hustá a již se nelze bavit o řešení řídkého LS problému. Obecně tedy matice s řádky různé řídkosti činí podstatnou obtíž pro řídký LS problém. V numerických experimentech práce je toto demonstrováno na konkrétnějších příkladech. Tento odstavec krátce shrnuje, jak se pro takový druh matic vyhnout zaplnění soustavy normálních rovnic C. Za prvé, pokud matice obsahuje relativně málo hustých řádků, je možné je ignorovat, vypočítat faktorizaci pro řídké řádky pomocí řídkých řešičů a husté řádky zohlednit následně. Další variantou přístupu je rozlišit řídké a husté řádky a počítat jejich faktorizace zvlášť, tj.

$$A = \begin{pmatrix} A_s \\ A_d \end{pmatrix}, A_s \in \mathbb{R}^{m_s \times n}, A_d \in \mathbb{R}^{m_d \times n}, b = \begin{pmatrix} b_s \\ b_d \end{pmatrix}, b_s \in \mathbb{R}^{m_s}, b_d \in \mathbb{R}^{m_d},$$

kde $m = m_s + m_d$, $m_s \geq n$, $m_d \geq 1$. Potom řídký LS problém přejde na tvar

$$\min_x \left\| \begin{pmatrix} A_s \\ A_d \end{pmatrix} x - \begin{pmatrix} b_s \\ b_d \end{pmatrix} \right\|_2 .$$

Dále se předpokládá, že matice A_s má plnou sloupcovou hodnost. Buď $C_s = A_s^T A_s$ redukovaná normální matice, pak s využitím Woodburyho formule lze inverzi matice C soustavy normálních rovnic $A^T A$ zapsat jako

$$C^{-1} = (C_s + A_d^T A_d)^{-1} = C_s^{-1} - C_s^{-1} A_d^T (I_{m_d} + A_d C_s^{-1} A_d^T)^{-1} A_d C_s^{-1}.$$

Díky tomu se snadno vyjádří explicitní tvar řešení LS problému, tj.

$$\begin{aligned} x &= x_s + C_s^{-1} A_d^T (I_{m_d} + A_d C_s^{-1} A_d^T)^{-1} (b_d - A_d x_s), \\ x_s &= (A_s A_s^T)^{-1} A_s^T b_s. \end{aligned}$$

Poslední uvedenou možností, jak řešit problém různé řídkosti řádků, je tzv. natažení. Tato metoda byla poprvé navržena v [10]. Její princip spočívá v roztržení hustých řádků matice A_d do více řádků tak, aby řešení LS problému zůstalo stejné, ale matice soustavy normálních rovnic zůstala řídká. Hlubším studiem řešení LS problému s maticí s různě řídkými řádky se zabývá článek [15].

3. Numerické experimenty

Tato kapitola se věnuje předpodmínění soustavy normálních rovnic neúplnou Choleského faktorizací. Buď předpodmiňovací matice $M = LL^T$, kde L je řídký Choleského faktor matice soustavy A^TA. Potom

$$A^TA = LL^T - E,$$

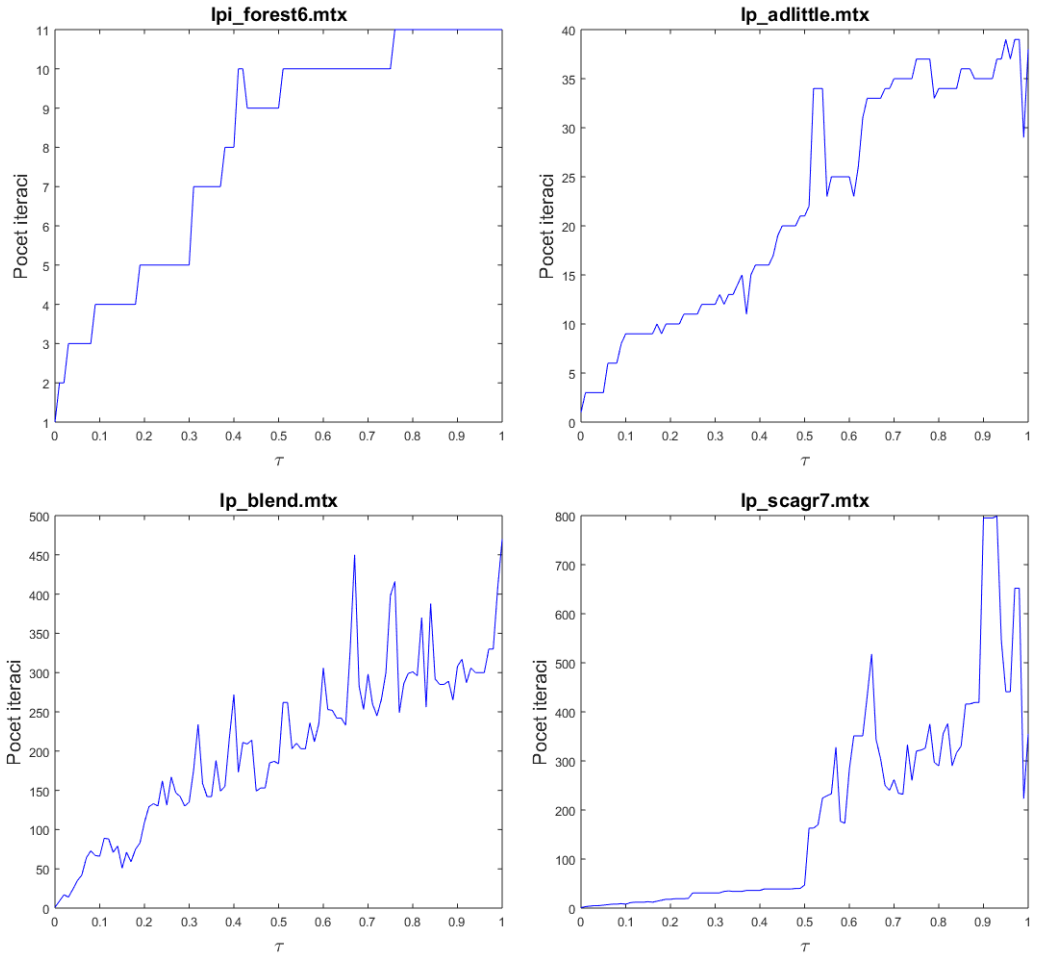
kde E je chybová matice s ideálně malou normou. V článku [8] se ukazuje, že řád konvergence CG předpodmíněným neúplnou Choleského faktorizací je přímo ovlivněn velikostí normy E, a nikoliv plněním faktoru L. Faktor L lze generovat více postupy. Proto je neúplná Choleského faktorizace (dále jen IC z anglického termínu *Incomplete Cholesky Factorization*) mnoha různých typů. Obecně ale matici transformuje na tvar LL^T, ve kterém některé nenulové prvky jsou ignorovány. Tedy pokud byl prvek na pozici $(i,j)_A$ v původní matici A nulový a prvek na stejné pozici $(i,j)_L$ ve faktorovém tvaru je nenulový, pak může být tento prvek uvažován jako nulový.

Jedním druhem IC jsou tzv. úrovňové Choleského faktorizace IC(l) (z anglického termínu *level-based IC*). Tyto úrovňové Choleského faktorizace IC(l) mají předem dánu řídkou strukturu v množině indexů původní matice. Během symbolické fáze faktorizace je každému prvku přiřazena úroveň (z anglického termínu *level*) a tento prvek se ve faktorizaci projeví, pokud je jeho úroveň nejvýše l. Je zřejmé, že s rostoucím l se zvětšuje zaplnění předpodmiňovače, což zvyšuje paměťovou i výpočetní náročnost nejen během faktorizace, ale i během aplikování předpodmínění. Proto má smysl v praxi využívat pouze několik úrovní, což bude ověřeno numerickými experimenty této kapitoly. Nejběžněji se využívá IC(l) pro hodnoty parametru $l \in \{0, 1, 2, 3\}$. Konkrétně se v numerických experimentech objeví tzv. level-zero neúplná faktorizace IC(0), což je přesná řídká Choleského faktorizace, a pak neúplné faktorizace několika dalších úrovní. Ve variantě IC(0) může nastat breakdown, protože pivoty mohou být nulové nebo záporné, a pak předpodmiňovač není pozitivně definitní. Proto je nutné přistoupit k jisté modifikaci diagonálních prvků. Toto je k nalezení například v knize [2]. Zdrojem robustního algoritmu neúplné Choleského faktorizace předpodmiňující metodu sdružených gradientů CG je například článek [1]. Paralelní implementaci neúplných Choleského předpodmiňovačů je věnován článek [12].

Místo předepsání řídké struktury je možné vynechat prvky Choleského faktorizace, které jsou v absolutní hodnotě menší než předem stanovená tolerance. Tento typ neúplných Choleského faktorizací je obecně značen IC(τ) (z anglického termínu *Threshold-based Incomplete Cholesky Factorization*). Nevýhodou tohoto přístupu je, že nelze apriori zjistit paměťovou náročnost pro danou toleranci. Pokud je zvolena moc vysoká, pak může konvergenci metody spíše zpomalit, což je v rozporu s účelem předpodmínění. Bohužel volba vhodné tolerance τ je přímo závislá na problému, a je tedy nutné využívat heuristických přístupů. IC(τ) se snižující se velikostí tolerance τ může poskytnout lepší předpodmiňovač, ale obvykle roste zaplnění. Důsledkem toho je nutné volit mezi kvalitou a řídkostí.

Někdy se ke generování L faktoru využívá ortogonálních rozkladů, konkrétně pak modifikovaného neúplného Gram-Schmidtova rozkladu, jehož algoritmická reprezentace je například v knize [14].

Jak bylo zmíněno v podkapitole 2.4, v praxi mohou činit značné obtíže úlohy, kde je řídkost různých řádků velmi odlišná. To může vést k nestabilitě metody PCGLS. V následujících experimentech je ilustrováno, jak se může měnit počet iterací při změnách parametru IC, protože právě závislostí na tomto parametru je myšlena nestabilita metody PCGLS. Jako vstupní data byly voleny matice z lineárního programování, tj. matice A byla z praxe získaná matice modelu a vektor b pravé strany byl volen náhodně. Dále je nutné poznamenat, že použité implementace jsou předběžné, a tedy nikoliv optimální. Především nemají ideální složitost, například námi aplikovaná implementace faktorizace IC(τ) vyžaduje až n^3 operací. Zde je pozornost ale věnována zjištění a ověření některých kvalitativních vlastností neúplných rozkladů, a proto není složitost tolik podstatná. Seznam použitých kódů je uveden v přílohách 3.



Obrázek 3.1: Srovnání závislosti počtu iterací metody PCGLS na toleranci τ.

Kromě výše uvedeného patří k analýze neúplné Choleského faktorizace IC(τ) i následující úvahy. V první řadě se obvykle traduje, že pro vyšší hodnotu τ dává IC(τ) méně přesný faktor L. Toto by mělo vést na zvyšující se počet iterací metody PCGLS nutných k získání dostatečně přesné aproximace, viz první graf Obrázku 3.1. Ukazuje se, že tomu tak není a dokonce to potvrzují už i velmi jednoduché matice, viz druhý graf Obrázku 3.1. Tedy se zvyšující se velikostí τ, nemusí nutně počet iterací narůstat. Navíc je zjevné, že IC(τ) může konvergenci i poškodit, a tak činit závislost počtu iterací na hodnotě parametru τ nesta-

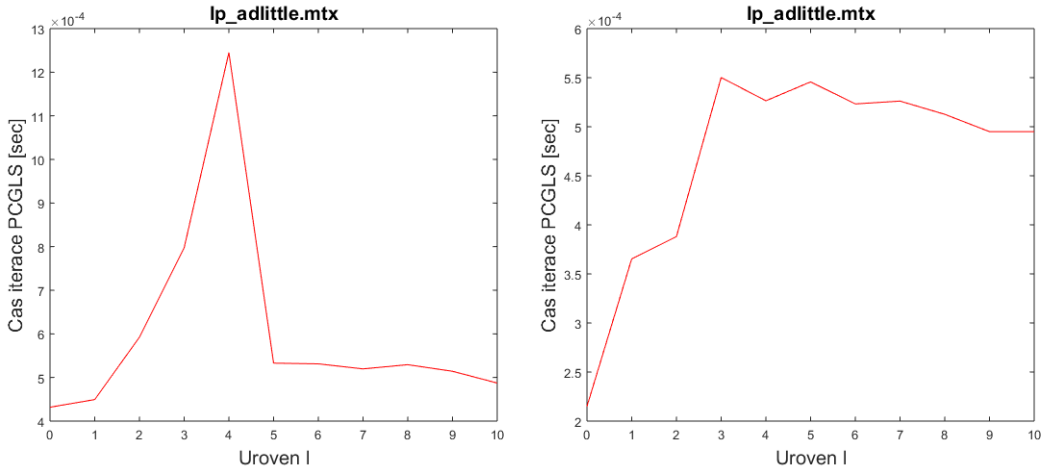
bilní. Platí totiž, že je možné dosáhnout stejně přesných výsledků ve stejném počtu iterací s vyšší hodnotou tolerance τ. Tento výsledek dokazují poklesy křivek na Obrázku 3.1. V analýze těchto jevů je nutné totiž nezohledňovat řídkou strukturu. Faktor L sice bude se zvyšující se tolerancí τ řidší, avšak to neznamená, že se z hustých řádků stanou řádky řídké. Proto se výpočetní náročnost nemusí snižovat a křivka počtu iterací nemusí být neklesající. Domněnkou tedy je, že v tomto případě je důvodem nestabilního efektu aplikace neúplné faktorizace na matici normálních rovnic. Právě taková nestabilita celé metody pracující s maticí s řádky různé řídkosti motivuje výzkum nových přístupů.

Dále se text věnuje úrovňové neúplné Choleského faktorizaci IC(l). Konkrétně jejím vlastnostem ovlivňovaným velikostí úrovní. Počet iterací s rostoucí velikostí úrovně l zpravidla prudce klesá. To je způsobeno větší řídkostí faktoru L. Avšak lze pozorovat, že křivka počtu iterací není nutně nerostoucí, viz Obrázek 3.3. To může být způsobeno právě různou řídkostí řádků matice.

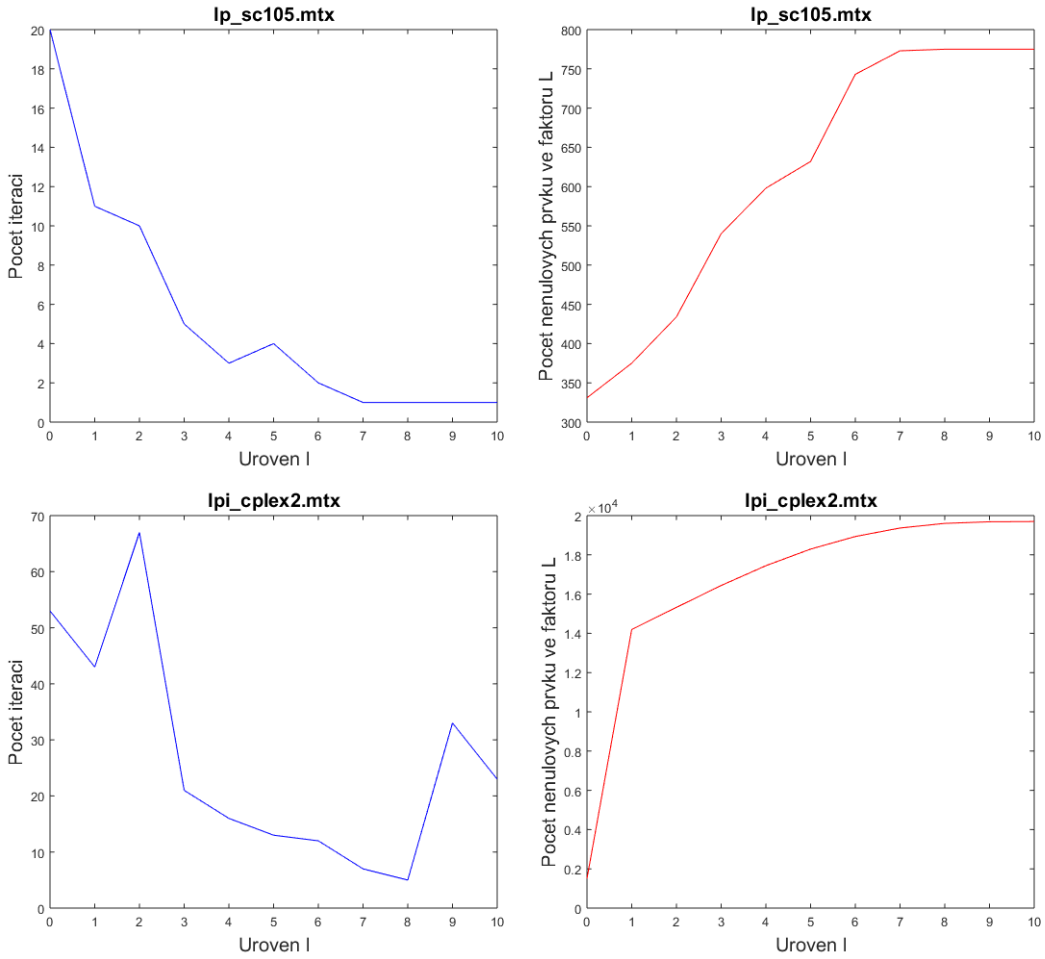
Na druhou stranu čas potřebný na každou iteraci bude prudce narůstat. Toto není možné v prostředí Matlabu dobře ilustrovat. Důvodem je, že v Matlabu lze počítat pouze s relativně malými maticemi kvůli paměťové a časové náročnosti výpočtu. Takové výpočty se provádějí tzv. *in cache*, tj. ve vyrovnávací paměti počítače. To znamená, že výpočet bude vždy velmi rychlý, což navíc i někdy způsobí, že měření času výpočtu bude dávat pokaždé jiné výsledky pro stejný vstup. V absolutních číslech se tyto různé výstupy budou lišit řádově v tisícinách sekund, avšak průběh křivky bude vždy velmi odlišný, viz Obrázek 3.2. Pro malé úlohy se proto z křivek času iterací nedají vyvozovat žádné výsledky.

Platí ale, že pro velké úlohy je čas výpočtu úměrný velikosti faktoru L, resp. počtu nenulových prvků faktoru L. Toto lze snadno vysledovat ze schématu iterační metody. Neúplná Choleského faktorizace IC(0) nedovoluje žádné zaplnění během výpočtu, a proto se rovná přesné Choleského faktorizaci. Tudíž pro hodnotu úrovně $l = 0$ počet nenulových prvků faktoru L odpovídá počtu nenulových prvků původní matice A. S vyšší hodnotou úrovně l je dovoleno větší zaplnění, tedy křivka počtu nenulových prvků prudce roste, viz Obrázek 3.3.

Z experimentů vyplývá, že nestabilita vzhledem k parametru IC je výrazně nižší pro úrovňovou neúplnou Choleského faktorizaci IC(l) než pro IC(τ). Na druhou stranu, se ale IC(l) potýká s výraznějším zaplněním. Odtud je zřejmé, že v praxi má smysl využívat IC(l) pouze pro několik málo úrovní.



Obrázek 3.2: Časy iterací metody PCGLS v sekundách pro různé úrovně IC(l).



Obrázek 3.3: Srovnání počtu iterací metody PCGLS a počtu nenulových prvků faktoru L vůči velikosti úrovně l předpodmínění IC(l).

Závěr

V práci byly shrnuty základní poznatky o LS problému, o existenci a jednoznačnosti jeho řešení a jednotlivých přístupech samotného výpočtu LS řešení. Podrobněji byly rozebrány přímé metody a jejich numerické vlastnosti. Ukazuje se, že přímé metody jsou vhodné pro relativně malé a husté úlohy. Práce tudíž směřuje k efektivnějším metodám pro řešení LS problému s velkou řídkou maticí, které jsou v praxi podstatně běžnější. Takové jsou iterační metody. Z nich je vyzdvižena metoda sdružených gradientů CG, pro kterou je i dokázána věta o konvergenci a závislosti na rozložení vlastních čísel.

Tématem práce jsou neúplné faktorizace pro řešení LS problému. Těmi jsou myšleny variace přímých metod a jejich rozkladů, které jsou častými předpodmiňovači iteračních metod. Konkrétně je rozebrán neúplný LU rozklad, BIF, neúplná Choleského faktorizace IC, AINV a Choleského neúplný Gram-Schmidt CIMGS. Tyto předpodmiňovače jsou vhodné pro LS problém řešený pomocí soustavy normálních rovnic.

Numerické experimenty se věnují metodě PCGLS sdružených gradientů předpodmíněné neúplnou Choleského faktorizací IC. Algoritmus pro sdružené gradienty je implementován pro řešení LS problému pomocí soustavy normálních rovnic. Jeho konvergence je urychlována IC. Úplná Choleského faktorizace by při výpočtu v přesné aritmetice měla zaručovat konvergenci metody již v jedné iteraci, ale je příliš výpočetně náročná. Z tohoto důvodu je uvažována její neúplná varianta IC. Ukazuje se, že metoda PCGLS je nestabilní vzhledem ke změnám parametru IC. První experiment demonstuje tuto nestabilitu pro IC(τ) na závislosti počtu iterací vůči velikosti tolerance τ. Druhý experiment ukazuje nestabilitu pro IC(l) na závislosti počtu iterací a počtu nenulových prvků faktoru L. Obvykle se předpokládá, že by tyto závislosti měly odpovídat hladkým křivkám, tomu ale tak není. Tato nestabilita může být způsobena nestejnou řídkostí řádků matice. Proto se práce zmiňuje o maticích s řádky různé řídkosti. Otevřenou otázkou zůstává, jak s takovými maticemi zacházet, aby se zabránilo nepříznivému vlivu hustých řádků.

Seznam použité literatury

[1] Ajiz, M. A. a Jennings, A. (1984). A robust incomplete Choleski-conjugate gradient algorithm. **20**(5), 949–966.

[2] Björck, Å. (1996). *Numerical methods for Least Squares Problems*. SIAM, Philadelphia.

[3] Björck, Å., Elfving, T. a Strakoš, Z. (1998). Stability of conjugate gradient and lanczos methods for linear least squares problems. *Matrix Analysis and Applications*, **19**(3), 720–736.

[4] Bru, R., Marín, J., Mas, J. a Tůma, M. (2014). Preconditioned iterative methods for solving linear least squares problems. *SIAM J. Sci. Comput.*, **36**(4), A2002–A2022.

[5] Chan, T. F. a van der Vorst, H. A. (1997). Approximate and incomplete factorizations. In *D.E. Keyes, A.Sameh and V. Venkatakrishnan, eds., Parallel Numerical Algorithms, ICASE/LaRC Interdisciplinary Series in Science and Engineering*, pages 167–202, Amsterdam – Lausanne – New York – Oxford – Shannon – Singapore – Tokyo, 1997. Elsevier.

[6] Chen, Y. (1975). *Iterative methods for linear Least Squares Problems*. Technical Report CS-75-04, Canada.

[7] Davis, T. A., Rajamanickam, S. a Sid-Lakhdar, W. M. (2016). A survey of direct methods for sparse linear systems. *Acta Numerica*, pages 383–566.

[8] Duff, I. S. a Meurant, G. A. (1989). The effect of ordering on preconditioned conjugate gradients. *BIT Numerical Mathematics*, **29**, 635–657.

[9] Duintjer Tebbens, J., Hnětynková, I., Plešinger, M., Strakoš, Z. a Tichý, P. (2012). *Analýza metod pro maticové výpočty. Základní metody*. Vydání první. Matfyzpress, MFF UK.

[10] Grcar, J. F. (1990). Matrix stretching for linear equations. Technical Report SAND90-8723, Sandia National Laboratories.

[11] Hestenes, M. R. a Stiefel, E. (1952). Methods of conjugate gradients for solving linear systems. *J. of Research of the National Bureau of Standards*, **49**, 409–435.

[12] Hysom, D. a Pothen, A. (2001). A scalable parallel algorithm for incomplete factor preconditioning. *SIAM J. Sci. Comput.*, **22**, 2194–2215.

[13] Liesen, J. a Strakoš, Z. (2013). *Krylov subspace methods*. Numerical Mathematics and Scientific Computation. Oxford University Press, Oxford. Principles and analysis.

[14] Saad, Y. (2003). *Iterative Methods for Sparse Linear Systems*. Society for Industrial and Applied Mathematics, Philadelphia, PA, second edition.

[15] Scott, J. A. a Tůma, M. (2017). Solving mixed sparse-dense linear least squares problems by preconditioned iterative methods. *SIAM J. Sci. Comput.*, **39**, 16.

[16] Wilkinson, J. a Reinsch, C. (1971). *Handbook for Automatic Computation*, volume II: Linear Algebra. Springer.

Seznam obrázků

Seznam tabulek

Přílohy - Popis programů pro Matlab

numexp1.m	Algoritmus prvního experimentu, tj. závislosti počtu iterací na parametru IC(τ)
numexp2.m	Algoritmus druhého experimentu, tj. závislosti počtu iterací a počtu nenulových prvků na parametru IC(l)
pcgls_for_Lfactor.m	Algoritmus metody PCGLS
ictau.m	Algoritmus neúplné Choleského faktorizace IC(τ)
iclevel.m	Algoritmus neúplné Choleského faktorizace IC(l)