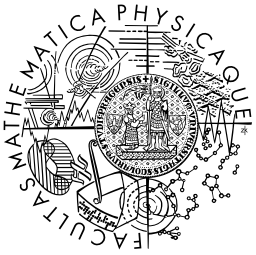


MATEMATICKO-FYZIKÁLNÍ FAKULTA
Univerzita Karlova

DIPLOMOVÁ PRÁCE

Jan Dvořák

Vlastnosti a konstrukce core problému v úlohách fitování dat s násobným pozorováním

Katedra numerické matematiky

Vedoucí diplomové práce: RNDr. Hnětynková Iveta, Ph.D.

Studijní program: Matematika

Studijní obor: Numerická a výpočtová matematika

Praha 2021

Prohlašuji, že jsem tuto diplomovou práci vypracoval(a) samostatně a výhradně s použitím citovaných pramenů, literatury a dalších odborných zdrojů. Tato práce nebyla využita k získání jiného nebo stejného titulu.

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

Rád bych poděkoval své vedoucí práce RNDr. Hnětynkové Ivetě, Ph.D. za ochotu, trpělivost a všechny dobré rady, které mi poskytla.

Název práce: Vlastnosti a konstrukce core problému v úlohách fitování dat s násobným pozorováním

Autor: Jan Dvořák

Katedra: Katedra numerické matematiky

Vedoucí diplomové práce: RNDr. Hnětynková Iveta, Ph.D., Katedra numerické matematiky

Abstrakt: V této prací studujeme řešení lineárních aproximačních problémů s násobným pozorováním. Konkrétně se zaměříme na metodu úplných nejmenších čtverců, která spadá mezi ortogonálně invariantní úlohy. Pro uvažovaný problém bude popsána tak zvaná core redukce. Jejím cílem je zredukovat problém na úlohu menších rozměrů při zachování stejného řešení, pokud existuje. Uvedeme dva způsoby konstrukce core problému, jeden přímý pomocí singulárního rozkladu a druhý využívající zobecněnou Golub-Kahanovu iterační bidiagonalizaci. Dále prozkoumáme vlastnosti core problému a metod pro jeho numerický výpočet. Na závěr provedeme numerické experimenty v prostředí Matlab za účelem otestování spolehlivosti uvažovaných algoritmů.

Klíčová slova: lineární aproximační problém, násobná pozorování, core problém, Golub-Kahanova iterační bidiagonalizace, blokové metody

Title: Properties and construction of core problem in data fitting problems with multiple observations

Author: Jan Dvořák

Department: Department of numerical mathematics

Supervisor: RNDr. Hnětynková Iveta, Ph.D., Department of numerical mathematics

Abstract: In this work we study the solution of linear approximation problems with multiple observations. Particulary we focus on the total least squares method, which belogs to the class of ortogonaly invariant problems. For these problems we describe the so called core reduction. The aim is to reduce dimesions of the problem while preserving the solution, if it exists. We present two ways of constructing core problems. One is based on the singular value decomposition and the other uses the generalized Golub-Kahan iterative bidiagonalization. Further we investigate properties of the core problem and of the methods for its construction. Finally we preform numerical experiments in the Matlab enviroment in order to test the reliability of the discussed algorithms.

Keywords: linear aproximation problem, multiple observations, core problem, Golub-Kahan iterative bidiagonalization, block methods

Obsah

Úvod **2**

1 Problém úplných nejmenších čtverců **4**
1.1 Základní definice a značení 4
1.2 Existence a jednoznačnost řešení 5
1.3 Klasický TLS algoritmus 8

2 Core problém **11**
2.1 Zavedení core problému 11
2.2 Přímá konstrukce core problému pro úlohy s jednonásobnou pravou stranou 12
2.3 Přímá konstrukce core problému pro úlohy s násobnou pravou stranou 19

3 Iterační konstrukce core problému **27**
3.1 Golub-Kahanova bidiagonalizace 27
3.2 Konstrukce pro úlohy s jednonásobnou pravou stranou 31
3.3 Konstrukce pro úlohy s vícenásobnou pravou stranou 34

4 Numerické experimenty **40**
4.1 Popis implementace 40
4.1.1 Klasický TLS algoritmus 40
4.1.2 Iterační konstrukce core problému 40
4.1.3 Generování testovacích úloh podle klasifikace TLS řešitelnosti 41
4.1.4 Generování testovacích úloh dle velikosti core problému . . 43
4.2 Chování algoritmu pro iterační výpočet core problému 44
4.2.1 Problémy s proměnnou velikostí pravé strany 44
4.2.2 Problémy s proměnnou velikostí core problému 47
4.3 Srovnání TLS řešení originální úlohy a core problému 50

Závěr **52**

Literatura **54**

Úvod

V této práci se budeme zabývat úlohami

$$AX \approx B, \tag{1}$$

kde $A \in \mathbb{R}^{m\times n}$ představuje model a $B \in \mathbb{R}^{m\times d}$ obsahuje ve sloupcích vícenásobná pozorování. Obě matice A i B obsahují chyby, které mohou být například zaokrouhlovací, diskretizační nebo chyby modelu. Z tohoto důvodu nemají často aproximační úlohy (1) přesné řešení. Na úlohy tohoto typu můžeme narazit mimo jiné při zpracování obrazu, ve statistických aplikacích nebo v medicíně (tomografie, renografie).

Konkrétně se zaměříme na řešení úloh (1) ve smyslu úplných nejmenších čtverců. Problém úplných nejmenších čtverců je detailně rozebrán v [5] a [17]. Tento přístup je pro námi uvažovaný typ aproximačních problémů vhodný, neboť předpokládáme, že chyby jsou přítomny jak v matici A tak v pravé straně B. Metoda úplných nejmenších čtverců hledá minimální opravu matic A a B takovou, aby opravená úloha již měla přesné řešení. Metodu popíšeme a charakterizujeme řešitelnost problému (1) uvedeným postupem. Úplná klasifikace řešitelnosti úloh ve smyslu úplných nejmenších čtverců byla představena v [9]. Zmíníme také klasický TLS algoritmus sloužící k výpočtu řešení ve smyslu úplných nejmenších čtverců, který je popsán v [18] a [19].

Dále se budeme zabývat hledáním core problému v úloze (1). Core problémem pro úlohy s jednou pravou stranou se zabývá [15]. Případ s vícenásobnou pravou stranou je rozebrán v [11]. Myšlenkou tohoto přístupu je, za pomoci ortogonálních transformací, zredukovat původní úlohu (1) na úlohu menší dimenze, kterou nazýváme core problém. Redukce probíhá odstraněním přebytečné informace z původních dat. Konkrétně se jedná o odstranění nulových singulárních čísel, některých násobností vícenásobných singulárních čísel, lineárně závislých sloupců pravé strany B nebo sloupců B nekorelovaných s modelem A a podobně. Nejprve vždy uvedeme metody výpočtu core problému na zjednodušeném případu, kdy $d = 1$. Poté vysvětlíme zobecnění na případ více pravých stran. V první řadě se zaměříme na přímou core redukci, která je popsána v [11]. Dále shrneme iterační přístup založený na zobecněné Golub-Kahanově iterační bidiagonalizaci, jenž lze nalézt v [10]. Jak TLS algoritmus pro výpočet řešení ve smyslu úplných nejmenších čtverců, tak iterační metodu pro výpočet core problému implementujeme v prostředí Matlab. Core redukci lze využít také k řešení úlohy (1) ve smyslu úplných nejmenších čtverců. Nejprve spočítáme příslušný core problém, potom aplikujeme na menší úlohu klasický TLS algoritmus. Na závěr získané řešení zpětně transformujeme. Problém úplných nejmenších čtverců a core problém dává do souvislosti [12], kde se analyzuje řešitelnost core problému ve smyslu úplných nejmenších čtverců. Nakonec provedeme numerické experimenty s cílem studovat chování popsaných algoritmů.

Celá práce je rozdělena do čtyř kapitol. První kapitola je věnována problému úplných nejmenších čtverců. Problém je definován a je rozebrána existence a jednoznačnost řešení. Také je zde popsán klasický TLS algoritmus pro výpočet řešení ve smyslu úplných nejmenších čtverců. V druhé kapitole je uvedena defi-

nice core problému a jeho přímá konstrukce pomocí singulárního rozkladu, která je nejprve představena pro případ $d = 1$ a poté zobecněna pro $d > 1$. Na závěr jsou vyslovena dvě tvrzení charakterizující vlastnosti core problému. Třetí kapitola obsahuje druhou metodu pro výpočet core problému. Jedná se o iterační metodu založenou na krylovovské iterační metodě zvané Golub-Kahanova bidiagonalizace. Nejprve proto zavedeme pojem Krylovova prostoru a popíšeme základní a zobecněnou Golub-Kahanovu iterační bidiagonalizaci. Poté s její pomocí zkonstruujeme core problém. Stejně jako v předchozí kapitole je konstrukce nejdříve provedena pro případ $d = 1$ a až potom zobecněna na situaci, kdy $d > 1$. Poslední čtvrtá kapitola se zabývá numerickými experimenty. Testujeme chování iteračního algoritmu pro výpočet core problému, který je založen na konstrukci uvedené v třetí kapitole. Zkoumáme spolehlivost výpočtu pro různě velké úlohy. Na závěr také srovnáváme řešení získané klasickým TLS algoritmem aplikovaným na úlohu (1) s řešením získaným postupem, kdy nejprve spočteme core problém v (1), poté aplikujeme klasický TLS algoritmus na získaný core problém a výsledné řešení zpětně transformujeme.

1. Problém úplných nejmenších čtverců

V této sekci stručně zavedeme problém úplných nejmenších čtverců, což je jeden z přístupů pro nalezení přibližného řešení systému lineárních rovnic. Definujeme řešení ve smyslu úplných nejmenších čtverců a zavedeme klasifikaci jednotlivých případů podle řešitelnosti. Převážně se budeme zabývat problémy s vícenásobnou pravou stranou a pro doplnění uvedeme analogii s jednou pravou stranou. Popis a analýzu problému úplných nejmenších čtverců lze nalézt v [5] a [17]. Závěrem uvedeme také tzv. klasický TLS algoritmus sloužící k výpočtu řešení ve smyslu úplných nejmenších čtverců, který lze nalézt v [18] a [19].

1.1 Základní definice a značení

Mějme matice $A \in \mathbb{R}^{m\times n}$, $B \in \mathbb{R}^{m\times d}$ a $X \in \mathbb{R}^{n\times d}$. Nyní uvažujme řešit následující lineární aproximační problém:

$$AX \approx B. \tag{1.1}$$

Aproximační úlohu (1.1) budeme řešit ve smyslu úplných nejmenších čtverců. Tento přístup je založen na nalezení matic $E \in \mathbb{R}^{m\times n}$ a $F \in \mathbb{R}^{m\times d}$ splňujících:

$$[E,F] = \underset{[C,D]\in\mathbb{R}^{m\times(n+d)}}{\arg\min} |||[C,D]|||_F \quad \text{a} \quad R(B+F) \subset R(A+E). \tag{1.2}$$

Pokud takové matice E a F existují, označme $A' = A + E$ a $B' = B + F$. Formulace (1.2) nám tedy říká, že existuje přesné řešení soustavy $A'X = B'$, kterou lze ekvivalentně zapsat ve tvaru

$$\begin{bmatrix} B' & A' \end{bmatrix} \begin{bmatrix} -I_d \\ X \end{bmatrix} = 0. \tag{1.3}$$

Poznámka. Rovnice (1.3) se pro případ jedné pravé strany, tj. pro $d = 1$, zjednoduší do tvaru

$$\begin{bmatrix} b' & A' \end{bmatrix} \begin{bmatrix} -1 \\ x \end{bmatrix} = 0,$$

kde b',x jsou vektory zastupující matice B' a X z obecné formulace (1.3).

Nyní můžeme přistoupit k definici samotného řešení úlohy (1.1) ve smyslu úplných nejmenších čtverců.

Definice 1. *Uvažujme aproximační úlohu (1.1). Nechť $E \in \mathbb{R}^{m\times n}$ a $F \in \mathbb{R}^{m\times d}$ jsou libovolné matice splňující (1.2). Potom každou matici $X \in \mathbb{R}^{n\times d}$, která je řešením rovnice $A'X = B'$ nebo ekvivalentně (1.3) nazveme řešením úlohy (1.1) ve smyslu úplných nejmenších čtverců.*

V dalších částech budeme potřebovat využít singulárního rozkladu [6], proto zde zavedeme značení pro singulární rozklad rozšířené matice $[B,A]$. Nechť

$$[B,A] = USV^T. \tag{1.4}$$

je singulární rozklad matice $[B,A]$ a dále označme její singulární čísla seřazená sestupně jako $(\sigma_1,\ldots,\sigma_{n+d})$. Příslušné levé (resp. pravé) singulární vektory budeme značit $(u_1,\ldots,u_m)$ (resp. $(v_1,\ldots,v_{n+d})$).

Poznámka. Z rovnice (1.3) si můžeme všimnout, že hodnost matice $[B',A']$ musí být maximálně n, aby nějaké řešení mohlo existovat. Z Eckart-Young-Miskrovi věty (viz [2]) víme, že nejlepší aproximace matice $[B,A]$ maticí hodnosti n je

$$[B',A'] = \sum_{i=1}^{n} u_i \sigma_i v_i^T$$

a to ve smyslu minimalizování následující normy

$$||[B',A'] - [B,A]||_F.$$

Tedy pokud v jádru matice $[B', A']$ existuje d vektorů tvaru $[-I_d, X^T]^T$ jako v (1.3) získáme tím opravu splňující (1.2). V jednorozměrném případě by to znamenalo, že v jádru matice $[b', A']$ existuje vektor s nenulovou první složkou.

1.2 Existence a jednoznačnost řešení

Zde uvedeme rozbor případů, ve kterých řešení ve smyslu úplných nejmenších čtverců existuje, popřípadě je jednoznačné, a nebo naopak neexistuje. Pro tuto klasifikaci zavedeme následující notaci. Pro singulární číslo σ_{n+1} matice $[B,A]$ označíme r jeho pravou násobnost a l levou násobnost, což zachycuje následující schéma

$$\sigma_{n-l} > \sigma_{n-l+1} = \ldots = \sigma_n = \sigma_{n+1} = \ldots = \sigma_{n+r} > \sigma_{n+r+1}.$$

Na základě tohoto rozdělení singulárních čísel zavedeme dělení matice V z singulárního rozkladu (1.4) do bloků

$$V = \begin{bmatrix} V_{11} & V_{12} & V_{13} \\ V_{21} & V_{22} & V_{23} \end{bmatrix}. \tag{1.5}$$

Blok V_{11} má rozměry $d \times (n-l)$ a V_{21} $n \times (n-l)$, dohromady tedy obsahují pravé singulární vektory příslušné největším $n-l$ singulárním číslům. Analýza existence řešení je založena pouze na pravých singulárních vektorech příslušných k nejmenším $l+d$ singulárním číslům. Bloky, kterými se budeme dále zabývat tedy jsou V_{12}, V_{22}, V_{13} a V_{23}, a mají po řadě rozměry $d \times (l+r)$, $n \times (l+r)$, $d \times (d-r)$ a $n \times (d-r)$.

Poznámka. Pro jednorozměrný případ, kdy $d = 1$, je v našem značení i $r = 1$ a tedy bloky V_{13} a V_{23} neexistují.

Poznámka. Řešení (1.1) ve smyslu úplných nejmenších čtverců nemusí vždy existovat. Tuto skutečnost ilustrujeme na následujícím jednoduchém příkladu. Položme

$$A = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \quad \text{a} \quad B = \begin{pmatrix} 1 & 2 \\ 1 & 1 \end{pmatrix}. \tag{1.6}$$

Zde vidíme, že soustava $AX = B$ je nekompatibilní. Neexistenci dokážeme tím, že nelze najít matice E,F splňující (1.2). Pro libovolné $\epsilon > 0$ uvažme matice

$$E = \begin{pmatrix} 0 & 0 \\ 0 & \epsilon \end{pmatrix} \quad \text{a} \quad F = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}.$$

Snadno nahlédneme, že soustava $(A + E)X = B + F$ je už nyní kompatibilní a navíc Frobeniova norma matice $[E,F]$ je rovna ϵ a tedy může být libovolně malá, což dokazuje neexistenci minimální opravy a tím i neexistenci řešení ve smyslu úplných nejmenších čtverců.

Nyní můžeme přistoupit ke klasifikaci řešitelnosti úlohy (1.1) ve smyslu úplných nejmenších čtverců. Tato klasifikace byla zavedena a odvozena v [9] a rozpadá se do dvou skupin, které značíme S a F. Skupina F se poté dále dělí na další podskupiny F_1, F_2 a F_3.

- **Skupina S:** Do této skupiny patří aproximační úlohy charakterizované následující vlastností

  $$rank([V_{12},V_{13}]) < d.$$

  Pro tyto problémy řešení ve smyslu úplných nejmenších čtverců neexistuje.

- **Skupina F:** Zde se nachazí úlohy splňující podmínku

  $$rank([V_{12},V_{13}]) = d.$$

  Jedná se tedy o všechny úlohy nespadající do skupiny S. Dále se pak dělí na další tři podskupiny.

  - F_1: Zde jsou problémy, kde jsou splněny navíc další dvě podmínky

    $$rank(V_{12}) = r \quad \text{a} \quad rank(V_{13}) = d - r.$$

    Za těchto předpokladů existuje řešení ve smyslu úplných nejmenších čtverců. Navíc lze nalézt jednoznačné řešení minimální ve Frobeniově i dvojkové normě. Toto řešení lze spočítat klasickým TLS algoritmem ([17], sekce 3.6). Řešení lze vyjádřit explicitně pomocí Moore-Penrosovi pseudoinverse.

Definice 2. *Nechť $A \in \mathbb{R}^{n\times m}$, $rank(A) = k$ a $A = U\Sigma V^T$ je singulární rozklad matice A. Potom Moore-Penroseovou pseudoinverzí matice A nazveme matici $A^\dagger$ definovanou vztahem*

$$A^\dagger = V \begin{bmatrix} \Sigma_k^{-1} & 0 \\ 0 & 0 \end{bmatrix} U^T,$$

kde Σ_k je $k \times k$ hlavní minor matice Σ.

Explicitní vyjádření má tvar

$$X = -[V_{22},V_{23}][V_{12},V_{13}]^\dagger.$$

- F_2: V této podskupině jsou splněny dvě doplňující podmínky

$$rank(V_{12}) > r \quad \text{a} \quad rank(V_{13}) = d - r.$$

Potom existuje řešení, které není jednoznačné. Řešení minimální v normě se může lišit v závislosti na zvolené normě a navíc jej nelze získat klasickým TLS algoritmem. Podrobné vysvětlení lze nalézt v [9], Sekce 4.2.

- F_3: Poslední podskupina problémů je ta, kdy je splněno

$$rank(V_{12}) > r \quad \text{a} \quad rank(V_{13}) < d - r$$

a v takovém případě řešení ve smyslu úplných nejmenších čtverců neexistuje.

Poznámka. Pro jednonásobnou pravou stranu ($d = 1$) se klasifikace výše zjednoduší pouze na skupiny F_1 a S. Navíc dodatečné podmínky ve skupině F_1 přejdou do podoby $V_{12} \neq 0$. Obdobně podmínka ve skupině S se změní na $V_{12} = 0$.

Poznámka. U úloh, kde neexistuje řešení ve smyslu úplných nejmenších čtverců, vysvětlíme jiný způsob, jakým úlohu smysluplně řešit. V tomto případě hledáme negenerické řešení. Jedná se o minimalizační úlohu (1.2) doplněnou o dodatečnou podmínku na matice $[E,F]$ ([17], Definice 3.3). Myšlenkou je, že pokud neexistuje klasické řešení ve smyslu úplných nejmenších čtverců, zvětšíme prostor, ve kterém hledáme řešení natolik, aby v něm existovalo nějaké řešení úlohy

$$(A + E)X = B + F,$$

kde E,F splňují (1.2) doplněnou o dodatečnou podmínku zmíněnou výše. Libovolné řešení této úlohy pak nazveme negenerickým řešením ve smyslu úplných nejměnších čtverců. Některé varianty jeho konstrukce budou uvedeny v další sekci v rámci klasického TLS algoritmu. Více o vlastnostech negenerických řešení je možné nalézt v [17] Sekce 3.4.

1.3 Klasický TLS algoritmus

V této části popíšeme bodově klasický TLS algoritmus (viz [18] a [19]) v několika variantách vhodných pro výpočet v konečné aritmetice. Jedná se o přímou metodu založenou na singulárním rozkladu rozšířené matice $[B,A]$. Zdůrazněme, že v reálném světe obvykle nemáme k dispozici přesnou úlohu (1.1), ale náš model, tj. matice A, a pravá strana pozorování B jsou zatíženy chybou E_A resp. E_B. Máme tedy k dispozici úlohu

$$(A + E_A)X \approx B + E_B.$$

Verze tohoto algoritmu byla použita k numerickým experimentům uvedeným později v této práci. Vstupními argumenty jsou matice A a B a kladný parametr tolerance tol.

1. **SVD:** V prvním kroku algoritmu spočítáme singulární rozklad rozšířené matice $[B,A]$,

$$[B,A] = USV^T.$$

2. **Výběr singulárního podprostoru:** Nyní musíme vybrat správná singulární čísla a příslušné pravé singulární vektory, ze kterých budeme později konstruovat řešení. V přesné aritmetice bychom zvolili číslo σ_{n+1} a všechny jemu rovna nebo menší. K nim příslušející pravé singulární vektory by pak odpovídaly, podle značení zavedeném v (1.5), matici

$$V_{min} = \begin{bmatrix} V_{12} & V_{13} \\ V_{22} & V_{23} \end{bmatrix}.$$

   V konečné aritmetice musíme počítat s tím, že nemáme přesně spočtený singulární rozklad a naše data obsahují šum a proto musíme výběr singulárních čísel trochu pozměnit. Zde existuje více možností.

   - Jednou z nich, která je prezentována v [17] (str. 87), je zavedení vhodného parametru R_{tol} a výběr všech singulárních číslel menších než R_{tol}, tedy

$$\sigma_k > R_{tol} > \sigma_{k+1} \geq \ldots \geq \sigma_{n+d}.$$

     Možná volba takového R_{tol}, pro případ kdy chyba v našich datech je rovnoměrně rozložená s nulovou střední hodnotou a rozptylem σ^2, je odvozena a vysvětlena v [17] na straně 89. Jedná se o volbu

$$R_{tol} = \sqrt{2max\{m,n+d\}}\sigma.$$

Tento přístup se velmi podobá takzvané metodě truncated TLS, při které se také jistým parametrem oddělí nejmenší singulární čísla a jsou vnímána jako by byla nulová. Tato metoda je používána jako regularizační metoda pro ill-posed úlohy (tj. úlohy velmi citlivé na chyby ve vstupních datech). Více k tomuto tématu je možné nalézt v [3].

- Přístup, který jsme zvolili pro naší implementaci algoritmu, je přímočařejší a více vychází z postupu v přesné aritmetice. Zavedeme parametr tolerance, $tol > 0$. Pomocí tohoto parametru odhadneme pravou násobnost singulárního čísla σ_{n+1} tak, že větší singulární čísla počínaje σ_n testujeme, zda-li splňují

$$\frac{\sigma_i - \sigma_{n+1}}{\sigma_{n+1}} < tol. \tag{1.7}$$

V takovém případě je považujeme za stejná. Příslušný pravý singulární prostor potom vypadá následovně

$$V_{min} = \begin{bmatrix} v_{n-k+1} & \dots & v_{n+d} \end{bmatrix},$$

kde k je odhadnutá pravá násobnost singulárního čísla σ_{n+1} spočtená postupem výše.

3. **Transformace matice V_{min}:** Nyní pomocí Householderových reflexí, které reprezentujeme ortogonální maticí H, převedeme matici V_{min} do tvaru

$$V_{min}H = \begin{bmatrix} 0 & V_{min}^{12} \\ V_{min}^{21} & V_{min}^{22} \end{bmatrix},$$

kde $V_{min}^{12} \in \mathbb{R}^{d \times d}$ je dolní trojúhelníková matice. Abychom mohli pokračovat, je nejprve nutné ověřit podmínku, zda-li je matice V_{min}^{12} regulární. Pokud ano, můžeme pokračovat v dalším kroku, jinak řešení ve smyslu úplných nejmenších čtverců neexistuje a budeme hledat takzvané negenerické řešení, tím, že se vrátíme o ke kroku 2 a zvolíme novou matici V_{min}.

Nová volba matice V_{min} se provede tak, že v první variantě s parametrem R přidáme k již vybraným singulárním číslům $\sigma_{k+1}, \dots, \sigma_{n+d}$ ještě číslo σ_k a všechny další splňující (1.7) pro nějaký zvolený parametr tol.

V druhé variantě postupujeme tak, že zahodíme všechna již použitá singulární čísla a zopakujeme postup v druhém kroku. To znamená, že vezmeme dalších d nejmenších singulárních čísel a přidáme ještě všechna větší splňující (1.7).

4. **Výpočet řešení:** Jakmile jsme z kroku 3 získali vhodnou matici $V_{min}H$ můžeme přejít k výpočtu X, hledaného řešení. Získáme ho vyřešením soustavy

$$XV_{min}^{12} = -V_{min}^{22}.$$

Tuto soustavu, lze vyřešit přímou eliminací, neboť máme matici V_{min}^{12} v dolním trojúhelníkovém tvaru. Řešení této soustavy navíc existuje, neboť v kroku 3, jsem ověřili, že matice V_{min}^{12} je regulární.

2. Core problém

V této kapitole se budeme věnovat core problému v lineární aproximační úloze (1.1), zavedené v předchozí kapitole. Cílem bude zredukovat tuto úlohu na úlohu co nejmenší dimenze tím, že se pokusíme odstranit přebytečná data. Tato redukce bude prováděna pomocí ortogonálních transformací původní úlohy. Řešení originální úlohy potom získáme z řešení redukovaného problému zpětnou transformací. Core problémem pro úlohy s jednonásobnou pravou stranou se zabývá článek [15]. Případ, kdy máme mnohonásobnou pravou stranu, je popsán v [11] a řešitelnost core problému ve smyslu úplných nejmenších čtverců je rozebrána v [12].

2.1 Zavedení core problému

Stejně jako v minulé kapitole uvažujme aproximační úlohu

$$AX \approx B. \tag{2.1}$$

Předpokládejme, že lze nalézt ortogonální matice P,Q a R, které převedou matice B a A do tvaru

$$P^T B R = \begin{bmatrix} B_1 & 0 \\ 0 & 0 \end{bmatrix},$$
$$P^T A Q = \begin{bmatrix} A_{11} & 0 \\ 0 & A_{22} \end{bmatrix},$$

kde matice A_{22} má alespoň jeden řádek a jeden sloupec, a počet řádků bloků A_{11} a B_1 je stejný. Transformujeme-li nyní celou úlohu (2.1) získáme

$$\begin{bmatrix} A_{11} & 0 \\ 0 & A_{22} \end{bmatrix} (Q^T X R) \approx \begin{bmatrix} B_1 & 0 \\ 0 & 0 \end{bmatrix}. \tag{2.2}$$

Zavedeme dělení matice $Q^T X R$ na

$$Q^T X R = \begin{bmatrix} X_{11} & X_{12} \\ X_{21} & X_{22} \end{bmatrix},$$

kde bloky odpovídající blokům transformované matice A, což jsou X_{11} a X_{22}, mají po řadě stejný počet řádků jako je počet sloupců bloků A_{11}, A_{22}, a počet sloupců X_{11} odpovídá počtu sloupců matice B_1. Můžeme tedy úlohu (2.2) rozdělit na podúlohy

$$\begin{aligned} A_{11}X_{11} &\approx B_1, \\ A_{11}X_{12} &\approx 0, \\ A_{22}X_{21} &\approx 0, \\ A_{22}X_{22} &\approx 0. \end{aligned} \tag{2.3}$$

Kromě prvního aproximačního problému z (2.3) lze jako řešení zvolit $X_{12} = X_{21} = X_{22} = 0$ (viz diskuze v [15]). Zredukovali jsem tedy řešení původního problému (2.1) na úlohu menších rozměrů $A_{11}X_{11} \approx B_1$. Řešení X původní úlohy pak získáme jednoduše z X_{11} zpětnou transformací

$$X = Q \begin{bmatrix} X_{11} & 0 \\ 0 & 0 \end{bmatrix} R^T.$$

Naším cílem bude najít ortogonální matice P,Q a R vedoucí na transformaci popsanou výše, kde navíc chceme, aby matice A_{11} a B_1 měli co nejmenší rozměry. Tím zajistíme, že úloha $A_{11}X \approx B_1$, což je jediná úloha, kterou musíme vyřešit, bude co nejmenší. Tyto úvahy vedou k následující definici core problému. Definice core problému byla prezentována v [15] ($d = 1$) a [11] ($d > 1$).

Definice 3. *Mějme úlohu (2.1). Potom problém $A_{11}X_{11} \approx B_1$ nazveme core problémem v úloze (2.1), pokud jsou splněny následující podmínky.*

1. *Existují ortogonální matice P,Q a R takové, že*

$$P^T AQ(Q^T XR) = \begin{bmatrix} A_{11} & 0 \\ 0 & A_{22} \end{bmatrix} \begin{bmatrix} X_{11} & X_{12} \\ X_{21} & X_{22} \end{bmatrix} \approx P^T BR = \begin{bmatrix} B_1 & 0 \\ 0 & 0 \end{bmatrix}.$$

2. *Dimenze matice $[B_1,A_{11}]$ je minimální možná mezi všemi, které lze získat transformací splňující splňující bod 1.*

Poznámka. Bod 1. v definici 3 se pro případ jedné pravé strany, to jest $d = 1$, zjednoduší na požadavek existence ortogonálních matic P a Q splňujících

$$P^T AQ(Q^T x) = \begin{bmatrix} A_{11} & 0 \\ 0 & A_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \approx P^T b = \begin{bmatrix} b_1 \\ 0 \end{bmatrix}. \tag{2.4}$$

Navíc lze snadno nahlédnout, že platí následující ekvivalence. Soustava $Ax \approx b$ je kompatibilní právě tehdy, když je kompatibilní soustava $A_{11}x_1 \approx b_1$.

V úvodu kapitoly jsme psali, že cílem core redukce je odstranění přebytečných dat z původní úlohy. A získali lepší představu, co je tím myšleno, uvedeme v další sekci nejprve přímou konstrukci core problému pomocí singulárního rozkladu pro případ jedné pravé strany. Tato konstrukce je popsána v [11], Sekce 2.

2.2 Přímá konstrukce core problému pro úlohy s jednonásobnou pravou stranou

Uvažujme zjednodušenou situaci, kdy máme pouze jednu pravou stranu, tedy $d = 1$. Jako dříve zdůrazníme tento fakt použitím značení b pro pravou stranu v úloze (2.1). Konstrukcí core problému pro případ $d = 1$ se poprvé zabývá [15]. Naším cílem bude z úlohy odstranit přebytečnou informaci. Transformujeme báze levých singulárních prostorů matice A tak, aby měl vektor pravé strany b pro každé různé singulární číslo matice A nenulovou projekci na nejvýše jeden bázový vektor příslušného levého singulárního prostoru. Pokud je vektor b kolmý na celý tento prostor, pak nebude mít žádnou nenulovou projekci. Dále z úlohy

odstraníme nulová singulární čísla matice A a každé nenulové singulární číslo matice A ponecháme v úloze právě jednou, pokud má pravá strana nenulovou projekci na levý singulární prostor příslušný tomuto číslu. V opačném případě toto singulární číslo odstraníme úplně. Celý postup rozdělíme do tří kroků

1. Transformace úlohy pomocí singulárního rozkladu matice A.
2. Úprava pravé strany.
3. Aplikování stejných úprav i na levou stranu aproximačního problému za účelem získání ortogonální transformace celé úlohy.

Krok 1: Nechť

$$A = \hat{U}\hat{S}\hat{V}^T \tag{2.5}$$

je singulární rozklad matice A. Označíme-li $k \leq min\{m,n\}$ hodnost matice A, pak pro singulární čísla platí

$$s_1 \geq \ldots \geq s_k > 0.$$

Nyní transformujeme úlohu následovně

$$(\hat{U}^T A\hat{V})(\hat{V}^T x) \approx \hat{U}^T b. \tag{2.6}$$

Matice $\hat{U}^T A\hat{V} = \hat{S}$ je diagonální matice a na diagonále mám všechna singulární čísla matice A uspořádaná sestupně.

Krok 2: V tomto kroku za pomocí Householderových reflexí natočíme báze jednotlivých levých singulárních prostorů matice A tak, aby vektor b měl nenulové projekce dle diskuze výše. Tím docílíme toho, že vektor pravé strany z úlohy (2.6) bude obsahovat nenulové prvky pouze na pozicích s indexy odpovídající indexům singulárních čísel, která zůstanou v naší úloze dle diskuze v úvodu sekce. Nejprve sestavíme matici Householderovy reflexe H_0 takovou, že

$$\begin{bmatrix} I_k & 0 \\ 0 & H_0 \end{bmatrix} \hat{U}^T b = \begin{pmatrix} b_k \\ t \\ 0 \end{pmatrix}, \tag{2.7}$$

kde $b_k \in \mathbb{R}^k$ obsahuje prvních k prvků vektoru $\hat{U}^T b$ a $t \in \mathbb{R}$ je různé od nuly. Obecně se může stát, že tato matice neexistuje a bude možné získat pouze transformaci s $t = 0$. Tento případ budeme diskutovat později, zatím předpokládejme, že lze nalézt transformaci výše. Tímto jsme transformovali levý singulární prostor matice A odpovídající nulovému singulárnímu číslu a vynulovali tím příslušné pozice ve vektoru $\hat{U}^T b$. Dále uděláme stejný proces pro všechna zbylá singulární čísla matice A. Označme $b_k = (b_k^1, \ldots, b_k^k)$. Nechť pro nějaké j platí

$$s_1 = \ldots = s_j > s_{j+1}.$$

Zkonstruujeme matici Householderovy reflexe H_{1j} splňující

$$H_{1j}\begin{pmatrix} b_k^1 \\ \vdots \\ b_k^j \end{pmatrix} = \begin{pmatrix} t_1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \quad \text{kde} \quad t_1 \neq 0.$$

Nyní zopakujeme tento postup pro s_{j+1}. Takto pokračujeme, dokud neprojdeme všechna nenulová singulární čísla. Obecný krok tedy vypadá tak, že pro $0 < i \leq j \leq k$ splňující

$$s_{i-1} > s_i = \ldots = s_j > s_{j+1},$$

zkonstruujeme matici Householderovy reflexe H_{ij}, pro kterou platí

$$H_{ij}\begin{pmatrix} b_k^i \\ \vdots \\ b_k^j \end{pmatrix} = \begin{pmatrix} t_i \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \quad \text{kde} \quad t_i \neq 0. \tag{2.8}$$

I zde se může stát, že matice H_{ij} nemusí existovat a bude možné získat pouze transformaci s $t_i = 0$. Tento případ bude diskutován později. Nyní budeme předpokládat, že všechny existují. Uděláme-li tento postup pro všechna singulární čísla, docílíme vynulování prvků vektoru $\hat{U}^T b$ dle diskuze výše. Označme $\hat{H}_1^T \hat{U}^T b$ jako výsledný vektor po aplikaci všech Householerových reflexí na $\hat{U}^T b$ a pro zjednodušení značení položme $H_1^T = \hat{H}_1^T \hat{U}^T$. Posledním krokem bude udělat permutaci P_1 prvků $H_1^T b$ tak, aby

$$P_1 H_1^T b = \begin{pmatrix} b_1 \\ 0 \end{pmatrix},$$

kde b_1 obsahuje pouze všechny nenulové prvky vektoru $H_1^T b$ v původním pořadí.

Krok 3: Stejné úpravy, jaké jsme udělali ve vektoru $\hat{U}^T b$, musíme aplikovat i na levé straně v (2.6), abychom získali ortogonální transformaci úlohy (2.1). Nejprve aplikujeme Householderovu reflexi H_0 zleva na blok matice $\hat{U}^T$ tvořený sloupci a řádky s indexem větším než k. Potom aplikujeme matice H_{ij} zleva na bloky matice $\hat{U}^T$, resp. zprava na bloky $\hat{V}$ tvořené sloupci a řádky i až j. Označíme-li jako H_2 matici $\hat{V}^T$ po aplikaci všech matic H_{ij}, dojdeme ke tvaru

$$(H_1^T A H_2)(H_2^T x) \approx H_1^T b.$$

Matice H_{ij} jsou aplikovány z obou stran a tedy ponechají matici $\hat{S}$ ze singulárního rozkladu A nezměněnou. To lze snadno nahlédnout z faktu, že $\hat{U}^T A \hat{V} = \hat{S}$ a aplikováním Householderových reflexí H_{ij} zleva a zprava na diagonální $\hat{S}$ se nezmění. Matice H_0 je sice aplikována pouze zleva, ale ovlivňuje pouze řádky a

sloupce s indexem vyšším než k a zde jsou v matici $\hat{S}$ pouze nulové prvky a tedy zůstane zachována.

Nyní nám zbývá pouze poslední krok. Použijeme stejnou permutaci na levou stranu, jakou jsme použili pro přerovnání prvků ve vektoru b. Přerovnáme tedy analogicky řádky matice H_1^T a sloupce matice H_2. Získáme tím transformaci tvaru

$$(P_1 H_1^T A H_2 P_2)(P_2^T H_2^T x) \approx P_1 H_1^T b.$$

Matice P_1 byla zavedena výše při transformaci vektoru pravé strany a P_2 permutuje prvních $k+1$ sloupců matice H_2 stejně, jako P_1 permutuje $k+1$ prvních pozic vektoru $H_1^T b$. Těmito permutacemi přeuspořádáme prvky matice $\hat{S}$ tak, že získáme tvar (za předpokladu existence všech použitých Householderových reflexí)

$$\hat{S} = \begin{bmatrix} S_1 & 0 \\ 0 & S_2 \end{bmatrix}.$$

Označme p počet různých nenulových singulárních čísel matice A. Potom $S_1 \in \mathbb{R}^{(p+1)\times p}$ je diagonální, na diagonále má právě jednou všechna nenulová singulární čísla matice A v původním pořadí a poslední řádek je nulový. Matice S_2 je také diagonální a obsahuje všechna zbylá singulární čísla matice A.

Položme $P = H_1 P_1^T$ a $Q = H_2 P_2$. Obdrželi jsme tímto ortogonální transformaci úlohy (2.1) tvaru

$$(P^T A Q)(Q^T x) = \begin{bmatrix} S_1 & 0 \\ 0 & S_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \approx P^T b = \begin{bmatrix} b_1 \\ 0 \end{bmatrix}. \tag{2.9}$$

Za našich předpokladů na existenci všech použitých Householderových reflexí jsou matice S_1 a S_2 tvaru, který je popsán výše. Obecně, jak uvidíme dále, může být rozdělení singulárních čísel mezi tyto dvě matice jiné a S_1 nemusí obsahovat poslední nulový řádek. Dělení vektoru x rozměrově odpovídá blokům S_1 a S_2. Získali jsem tedy adepta na core problém v úloze (2.1) pro případ $d = 1$ a to

$$S_1 x_1 \approx b_1.$$

Nyní se vrátíme k otázce existence příslušných Householderových reflexí. Problém řeší následující lemma.

Lemma 1. *Uvažujme úlohu (2.1) pro případ $d = 1$. Potom platí následující vztahy:*

1. *Matice Householderovy reflexe H_0 z (2.7) existuje právě tehdy, když soustava $Ax \approx b$ není kompatibilní.*
2. *Matice Householderovy reflexe H_{ij} z (2.8) existuje právě tehdy, když b není kolmý na levý singulární prostor příslušný singulárnímu číslu s_i.*

Důkaz. Pokud je k, hodnost matice A, menší než m, pak Householderova reflexe H_0 neexistuje právě tehdy, když vektor $\hat{U}^T b$ je od pozice $k+1$ nulový, tedy

$\hat{U}^T b = (b_k, 0)^T$. Toto nastane tehdy a jen tehdy, když $u_i^T b = 0$ pro každé $i \in \{k+1, \ldots, m\}$, kde u_i značí i-tý sloupec matice U. Z vlastností singulárního rozkladu získáme vztah

$$R(A) = span\{u_{k+1}, \ldots, u_m\}^{\perp}.$$

Z čehož je vidět, že $b \in R(A)$. Dokázali jsme, že příslušná Householderova reflexe neexistuje právě tehdy, když $B \in R(A)$, což je ekvivalentní kompatibilitě soustavy. Pokud má matice A plnou sloupcovou hodnost m, potom je soustava vždy kompatibilní a vektor b_k je celý $\hat{U}^T b$ a H_0 neexistuje.

Pro bod číslo 2 použijeme podobný argument. Householderova reflexe H_{ij} neexistuje právě tehdy, když je vektor $(b_k^i, \ldots, b_k^j)^T$ nulový, což je právě tehdy, když $u_l^T b = 0$ pro každé $l \in \{i, \ldots, j\}$. Vektory $u_i, \ldots, u_j$ tvoří bázi levého singulárního prostoru příslušícího singulárnímu číslu s_i. Dohromady tedy dostaneme, že daná Householderova reflexe neexistuje právě tehdy, když je vektor b kolmý na levý singulární prostor příslušný singulárnímu číslu s_i, což dokazuje tvrzení 2.

□

Získali jsme tedy podmínky existence transformace (2.9). Pokud ovšem není splněna nějaká podmínka z Lemma 1 neznamená to, že podobnou transformaci nelze zkonstruovat. Nechť pro nějaké $i \in \{1, \ldots, k\}$ neexistuje Householderova transformace H_{ij}. Potom je dle Lemma 1 příslušná část vektoru $\hat{U}^T b$ již nulová, v závěrečné permutaci jí proto celou přesuneme do druhé části vektoru a singulární čísla $s_i = \ldots = s_j$ budou přesunuta na diagonálu matice S_2. Obdobně, pokud neexistuje Housholderova reflexe H_0 znamená to, z důkazu Lemma 1, že $\hat{U}^T b$ má všechny prvky od pozice $k+1$ nulové a tedy není třeba jeho nenulový prvek přesouvat permutací k ostatním nenulovým prvkům. V tomto případě matice S_1 nebude obsahovat nulový řádek. Pro představu to demonstrujeme na jednoduchém příkladu.

Příklad. Ukážeme rozdíl v rozměrech matice S_1 v závislosti na kompatibilitě soustavy.

1. Nechť soustava $Ax \approx b$ není kompatibilní. Dále nechť $m = n = 6$ a platí

$$H_1^T \hat{U}^T b = (1,0,2,0,3,0)^T,$$
$$\hat{S} = diag(2,2,1,1,0,0).$$

   Pozice ve vektoru $H_1^T \hat{U}^T b$ jsou tedy vynulovány v souladu s popisem výše. Permutace P_1 v tomto případě permutuje pozice ve vektoru $H_1^T \hat{U}^T b$ následovně $1 \to 1, \quad 3 \to 2, \quad 5 \to 3, \quad 2 \to 4, \quad 4 \to 5, \quad 6 \to 6$. Výsledkem je vektor

$$P_1 H_1^T \hat{U}^T b = \begin{pmatrix} 1 & 2 & 3 & 0 & 0 & 0 \end{pmatrix}^T.$$

Nyní je stejná permutace aplikována na řádky matice $\hat{S}$.

$$P_1\hat{S} = \begin{pmatrix} 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

Poté je stejná permutace aplikována k proházení prvních pěti sloupců matice P_1S. V případě, kdy $m \neq n$, bychom formálně kvůli jiným rozměrům aplikovali permutaci P_2, která je na prvních pěti prvcích totožná a zbylé pozice jsou jejími pevnými body.

$$P_1\hat{S}P_1^T = \begin{pmatrix} 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

Odtud získáme

$$S_1 = \begin{pmatrix} 2 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix}, \quad S_2 = \begin{pmatrix} 0 & 2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}.$$

2. Nyní nechť je soustava kompatibilní. Potom vektor $\hat{U}^T b = (b_k,0)^T$ je nulový od pozice $k+1$ dále. Mějme stejnou matici $\hat{S}$ jako v předchozím případě. Vektor $H_1^T\hat{U}^T b$ nyní nemůže obsahovat nenulový prvek na pozici 5 a tedy nechť je tvaru

$$H_1^T\hat{U}^T b = \begin{pmatrix} 1 & 0 & 2 & 0 & 0 & 0 \end{pmatrix}^T.$$

Permutace P_1 v tomto případě permutuje pozice ve vektoru $H_1^T\hat{U}^T b$ následovně $1 \to 1, \quad 3 \to 2, \quad 2 \to 3, \quad 4 \to 4, \quad 5 \to 5, \quad 6 \to 6$. Získáme vektor

$$P_1H_1^T\hat{U}^T b = \begin{pmatrix} 1 & 2 & 0 & 0 & 0 & 0 \end{pmatrix}^T.$$

Nyní opět aplikujeme stejný postup na matici $\hat{S}$.

$$P_1\hat{S} = \begin{pmatrix} 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad P_1\hat{S} = \begin{pmatrix} 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

V tomto případě získáme bloky

$$S_1 = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}, \quad S_2 = \begin{pmatrix} 2 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}.$$

Máme tedy zaručenou existenci transformace tvaru (2.9). Posledním krokem bude dokázat, že je také minimální (splňuje bod 2. v Definici 3). K tomu nám slouží následující věta. Obdobné tvrzení je možné nalézt v [15] Lemma 2.1 a Věta 2.2.

Věta 2. *Uvažujme úlohu (2.1) pro případ $d = 1$. Nechť vektor b není kolmý na právě p levých singulárních prostorů matice A příslušících, po přeznačení, různým nenulovým singulárním číslům $\hat{s}_1 > \ldots > \hat{s}_p > 0$. Potom existují ortogonální matice P a Q splňující*

$$(P^T A Q)(Q^T x) = \begin{bmatrix} S_1 & 0 \\ 0 & S_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \approx P^T b = \begin{bmatrix} b_1 \\ 0 \end{bmatrix},$$

a úloha $S_1 x_1 \approx b_1$ je core problémem v úloze (2.1) pro $d = 1$. Navíc $S_1 \in \mathbb{R}^{(p+1)\times p}$ pokud úloha není kompatibilní, jinak $S_1 \in \mathbb{R}^{p\times p}$. V obou případech platí, že $S_1 = diag(\hat{s}_1, \ldots, \hat{s}_p)$ a S_2 je diagonální matice obsahující na diagonále všechna zbylá singulární čísla matice A.

Důkaz. Z konstrukce výše jsme odvodili existenci matic P a Q a příslušné transformace (2.9) pro diagonální S_1 a S_2 obsahující dohromady všechna singulární čísla matice A včetně násobností na diagonálách. Dle Lemma 1 není soustava $Ax \approx b$ kompatibilní právě tehdy, když existuje Householderova reflexe H_0, což z konstrukce a úvah výše znamená, že v matici S_1 přibude nulový řádek, a tedy bude mít o jeden řádek více než je počet sloupců.

Pro každé i takové, že b není kolmé na levý singulární prostor příslušný singulárnímu číslu s_i, v našem přeznačení pak tomuto číslu odpovídá nějaké $\hat{s}_l$ pro $l \in \{1, \ldots, p\}$, z Lemma 1 víme, že existuje Householderova reflexe H_{ij}. Což z principu naší konstrukce znamená, že singulární číslo bude obsaženo na diagonále matice S_1. Naopak, pokud je b kolmé na nějaký levý singulární prostor, pak neexistuje příslušná Householderova reflexe a toto číslo a všechna jemu rovná budou v permutačním kroku konstrukce odsunuta do matice S_2.

Posledním krokem je dokázat minimalitu. Nechť existují ortogonální matice R a S takové, že dávají transformaci

$$\begin{bmatrix} \bar{A}_{11} & 0 \\ 0 & \bar{A}_{22} \end{bmatrix} \begin{bmatrix} \bar{x}_1 \\ \bar{x}_2 \end{bmatrix} \approx \begin{bmatrix} \bar{b}_1 \\ 0 \end{bmatrix},$$

kde $\bar{A}_{11}$ má $q < p$ sloupců a plnou sloupcovou hodnost (jinak by bylo možné přesunout více sloupců do matice $\bar{A}_{22}$). Uvažujme následující singulární rozklady

$$\bar{A}_{11} = U_1 S_{11} V_1^T,$$
$$\bar{A}_{22} = U_2 S_{22} V_2^T.$$

Potom tedy

$$R^T AS = \begin{bmatrix} \bar{A}_{11} & 0 \\ 0 & \bar{A}_{22} \end{bmatrix} = \begin{bmatrix} U_1 S_{11} V_1^T & 0 \\ 0 & U_2 S_{22} V_2^T \end{bmatrix} = \\ = \begin{bmatrix} U_1 & 0 \\ 0 & U_2 \end{bmatrix} \begin{bmatrix} S_{11} & 0 \\ 0 & S_{22} \end{bmatrix} \begin{bmatrix} V_1^T & 0 \\ 0 & V_2^T \end{bmatrix}.$$

Tímto jsme spočetli singulární rozklad matice A. Matice S_{11} a S_{22} obsahují singulární čísla matice A. Odsud jsme schopni získat i následující transformaci

$$\underbrace{\begin{bmatrix} U_1^T & 0 \\ 0 & U_2^T \end{bmatrix} R^T AS \begin{bmatrix} V_1 & 0 \\ 0 & V_2 \end{bmatrix}}_{\begin{bmatrix} S_{11} & 0 \\ 0 & S_{22} \end{bmatrix}} \begin{bmatrix} V_1^T & 0 \\ 0 & V_2^T \end{bmatrix} S^T \begin{bmatrix} \bar{x}_1 \\ \bar{x}_2 \end{bmatrix} \approx \begin{bmatrix} U_1^T & 0 \\ 0 & U_2^T \end{bmatrix} R^T \begin{bmatrix} \bar{b}_1 \\ 0 \end{bmatrix},$$

což je transformace typu (2.9), kde S_{11} obsahuje singulární čísla matice A a má méně než p sloupců. Toto tvrzení je však z již dokázaného ve sporu s předpokladem, že vektor b není kolmý na právě p levých singulárních prostorů příslušných různým nenulovým singulárním číslům.

□

Tímto jsme tedy ukázali, odvozením přímého postupu konstrukce core problému, jeho existenci pro případ jedné pravé strany.

2.3 Přímá konstrukce core problému pro úlohy s násobnou pravou stranou

V této sekci zobecníme konstrukci popsanou v minulé části na případ, kdy máme vícenásobnou pravou stranu ($d > 1$). Bude se opět jednat o přímou metodu založenou na singulárním rozkladu matice A. Tomuto tématu se věnuje [11]. Oproti jednorozměrnému případu nám přibude jeden krok navíc. Pravá strana může obsahovat lineárně závislá pozorování (lineárně závislé sloupce matice B). Tato pozorování nám nepřináší žádnou novou informaci, a proto se jich zbavíme. Další tři kroky jsou zobecněním kroků v jednorozměrném případě. Opět budeme transformovat báze levých singulárních prostorů tentokrát tak, aby měla pravá strana pro každé různé singulární číslo matice A nenulovou projekci na maximálně d bázových vektorů příslušného levého singulárního prostoru. Stejně jako pro případ $d = 1$ z úlohy odstraníme nulová singulární čísla a každé nenulové v úloze necháme maximálně d krát v závislosti na počtu nenulových projekcí pravé strany na bázové vektory příslušného levého singulárního prostoru. Konstrukce bude mít následující čtyři kroky:

1. Odstranění lineárně závislých sloupců matice B.
2. Transformace úlohy pomocí singulárního rozkladu matice A.
3. Úprava pravé strany.

4. Aplikování stejných úprav i na levou stranu rovnice za účelem získání ortogonální transformace celé úlohy.

Krok 1: Odstranění lineárně závislých sloupců lze provést více způsoby. My použijeme singulární rozklad, ale je také možné použít například LQ rozklad. Nejprve zavedeme singulární rozklad matice B:

$$B = U_B S_B V_B^T.$$

Pokud obsahuje matice B lineárně závislé sloupce, potom je alespoň jedno singulární číslo nulové a matici S_B lze rozdělit na bloky

$$S_B = \begin{bmatrix} S_{B1} & 0 \end{bmatrix},$$

kde počet sloupců matice S_{B1} odpovídá sloupcové hodnosti matice B. Nyní matice $U_B S_{B1}$ má plnou sloupcovou hodnost a má navzájem ortogonální sloupce. Tuto matici použijeme jako novou pravou stranu. Za tímto účelem aplikujeme matici V_B na celou úlohu (2.1) zprava a získáme

$$A(XV_B) \approx \begin{bmatrix} U_B S_{B1} & 0 \end{bmatrix}.$$

Rozdělíme-li matici XV_B na bloky X_1 a X_0, odpovídající blokům pravé strany, úloha se rozpadne na dva podproblémy

$$AX_1 \approx U_B S_{B1}, \quad AX_0 \approx 0.$$

Stejně jako dříve, u zavedení core problému, za řešení druhého problému vezmeme $X_0 = 0$. Označme $B_1 = U_B S_{B1}$. Úloha po odstranění lineárně závislých sloupců pravé strany tedy vypadá následovně

$$AX_1 \approx B_1. \tag{2.10}$$

Krok 2: Nyní použijeme singulární rozklad matice A z (2.5) stejně jako v jednorozměrném případě. Jediný rozdíl spočívá v tom, že transformujeme úlohu (2.10).

$$(\hat{U}^T A \hat{V})(\hat{V}^T X_1) \approx \hat{U}^T B_1.$$

Zde opět matice $\hat{U}^T A \hat{V} = \hat{S}$ je diagonální matice obsahující singulární čísla matice A na diagonále.

Krok 3: Analogicky k jednorozměrnému případu budeme transformovat levé singulární prostory matice A tak, aby pro každé různé singulární číslo matice A měla matice B_1 nenulovou projekci, v tomto případě, na maximálně d bázových vektorů příslušného levého singulárního prostoru. Což znamená, že vynulujeme některé řádky matice pravé strany $\hat{U}^T B_1$. Pro případ $d = 1$ jsme používali Householderovy reflexe. Nyní místo nich použijeme singulární rozklady příslušných bloků matice $\hat{U}^T B_1$.

Nejprve označme k hodnost matice A. Prvních k singulárních čísel je tedy nenulových. Rozdělme matici pravé strany $\hat{U}^T B_1$ na bloky

$$\hat{U}^T B_1 = \begin{bmatrix} B_k \\ B_0 \end{bmatrix},$$

kde B_k je tvořena prvními k řádky a B_0 obsahuje zbylé řádky. Nyní bude naším cílem vynulovat co nejvíce řádků matice B_0, což je analogií ke konstrukci Householderovy reflexe (2.7) v případě $d = 1$. Nechť

$$B_0 = U_0 S_0 V_0^T \tag{2.11}$$

je singulární rozklad matice B_0. Matice $S_0 \in \mathbb{R}^{(m-k)\times d}$ (připomeňme, že m značí počet řádků matic A a B a tudíž i matice $\hat{U}^T B_1$) obsahuje nulové řádky, pokud platí, že $k_0 = rank(S_0) < (m - k)$. Z toho vyplývá, že matice $S_0 V_0^T$ má k_0 nenulových řádků, přičemž triviálně platí, že $k_0 \leq d$. Aplikujeme tedy namísto Householderovy reflexe H_0 z jednorozměrného případu matici U_0^T zleva na matici B_0. Získáme tím tvar

$$\begin{bmatrix} I_k & 0 \\ 0 & U_0^T \end{bmatrix} \hat{U}^T B_1 = \begin{bmatrix} B_k \\ S_0 V_0^T \end{bmatrix}. \tag{2.12}$$

Zatím jsme tedy vynulovali některé řádky s indexy odpovídající nulovým singulárním číslům matice A a zůstalo nám maximálně d nenulových řádků. Stejný postup teď aplikujeme na řádky s indexy odpovídající ostatním singulárním číslům. Označme j číslo splňující

$$\sigma_1 = \ldots = \sigma_j > \sigma_{j+1}.$$

Nyní použijeme značení b_k^i pro í-tý řádek matice B_k. Dále označme

$$B_{ik} = \begin{bmatrix} b_k^i \\ \vdots \\ b_k^j \end{bmatrix}$$

blok matice B_k obsahující i-tý až j-tý řádek. Dále postupujeme stejně jako výše. Nechť

$$B_{1j} = U_{1j} S_{1j} V_{1j}^T$$

je singulární rozklad matice B_{1j}. Označme $k_{1j} = rank(S_{1j})$. Potom na základě stejné úvahy, jako pro případ (2.11), má matice $S_{1j} V_{1j}^T$ přesně $k_{1j} \leq d$ nenulových řádků. Opět, namísto H_{1j} z jednorozměrného případu, aplikujeme matici U_{1j}^T zleva.

$$U_{1j}^T B_{1j} = S_{1j} V_{1j}^T.$$

Stejný postup zopakujeme pro singulární číslo σ_{j+1}. Tímto způsobem pokračujeme, dokud neprojdeme všechna nenulová singulární čísla. Obecný krok vypadá následovně. Mějme $0 < i \leq j \leq k$, pro která platí

$$\sigma_{i-1} > \sigma_i = \ldots = \sigma_j > \sigma_{j+1}.$$

Spočteme singulární rozklad matice B_{ij}

$$B_{ij} = U_{ij} S_{ij} V_{ij}^T$$

a aplikujeme matici U_{ij}^T zleva. Tím získáme

$$U_{ij}^T B_{ij} = S_{ij} V_{ij}^T, \tag{2.13}$$

kde $S_{ij}V_{ij}^T$ obsahuje $k_{ij} = rank(S_{ij}) \leq d$ nenulových řádků. V tuto chvíli tedy všechny transformace výše převedou matici $\hat{U}^T B_1$ do tvaru

$$\begin{bmatrix} S_{1j}V_{1j}^T \\ \vdots \\ S_{ik}V_{ik}^T \\ S_0 V_0^T \end{bmatrix}. \tag{2.14}$$

Dalším krokem bude přeuspořádat řádky této matice. Všechny nulové řádky přesuneme na konec. Jednoduše tedy zkonstruujeme permutační matici P_1, která posune všechny existující nulové řádky na konec a nenulové ponechá v nezměněném pořadí. Jedná se o principiálně stejnou permutaci na řádcích matice (2.14), jakou jsme aplikovali na prvky vektoru $H_1^T b$ v minulé sekci. Získáme tím tvar

$$P_1 \begin{bmatrix} S_{1j}V_{1j}^T \\ \vdots \\ S_{ik}V_{ik}^T \\ S_0 V_0^T \end{bmatrix} = \begin{bmatrix} B_{11} \\ 0 \end{bmatrix},$$

kde B_{11} obsahuje pouze nenulové řádky.

Krok 4: Nyní musíme všechny úpravy pravé strany aplikovat na celou úlohu (2.1), abychom získaly ortogonální transformaci celé soustavy. V prvním kroku jsme zredukovali matici B na matici B_1, která již neobsahuje lineárně závislé sloupce, což vedlo na tvar

$$A(XV_B) = A \begin{bmatrix} X_1 & X_2 \end{bmatrix} \approx BV_B = \begin{bmatrix} B_1 & 0 \end{bmatrix}.$$

Dále jsme v kroku dva využili singulární rozklad matice A z (2.5).

$$(\hat{U}^T A \hat{V})(\hat{V}^T X V_B) = \hat{S} \begin{bmatrix} \hat{V}^T X_1 & \hat{V}^T X_2 \end{bmatrix} \approx \hat{U}^T B V_B = \begin{bmatrix} \hat{U}^T B_1 & 0 \end{bmatrix}.$$

Nyní musíme aplikovat matici U_0^T zleva, jako v (2.12), a poté všechny matice U_{ij}^T z (2.13) zleva na příslušné bloky matice $\hat{U}^T$ tvořené řádky s indexy i až j. Obdobě pak na bloky matice $\hat{V}$ tvořené sloupci i až j zprava. Pro zjednodušení označme U_1^T matici $\hat{U}^T$ po aplikaci matice U_0^T a všech matic U_{ij}^T zleva. Podobně V_1 bude značit matici $\hat{V}$ po aplikaci všech U_{ij}^T zprava. Získáme tedy

$$(U_1^T A V_1)(V_1^T X V_B) \approx U_1^T B V_B.$$

To lze ekvivalentně přepsat ve tvaru

$$\hat{S}\begin{bmatrix} V_1^T X_1 & V_1^T X_2 \end{bmatrix} \approx \begin{bmatrix} U_1^T B_1 & 0 \end{bmatrix},$$

přičemž $U_1^T B_1$ je rovna matici (2.14). Poznamenejme, že stejně jako v jednorozměrném případě matice $\hat{S}$ zůstane nezměněná aplikací matic U_{ij} zleva a zprava. Posledním krokem je tedy aplikace permutační matice P_1 k přeuspořádání řádků pravé strany a prvků na diagonále matice $\hat{S}$ na levé straně.

$$(P_1\hat{S}P_2)(P_2^T\begin{bmatrix} V_1^T X_1 & V_1^T X_2 \end{bmatrix}) \approx P_1\begin{bmatrix} U_1^T B_1 & 0 \end{bmatrix} = \begin{bmatrix} B_{11} & 0 \\ 0 & 0 \end{bmatrix}. \tag{2.15}$$

Matice P_2, stejně jako v minulé sekci, je stejná permutace na prvních $k + k_0$ sloupcích matice $\hat{S}$ jako P_1 na prvních $k + k_0$ řádcích. Zbylé pozice zůstávají nezměněny. Jediný rozdíl je v rozměrech těchto dvou matic. Matice $P_1\hat{S}P_2$ má nyní obdobný tvar jako v jednorozměrném případě. Rozdíl je v tom, že nemáme nejprve na diagonále všechna různá nenulová singulární čísla maximálně jednou, ale každé singulární číslo σ_l je tam tolikrát, kolik nenulových řádků měla příslušná matice $S_{ij}V_{ij}^T$ pro $i \leq l \leq j$. Počet těchto nenulových řádků navíc odpovídá hodnosti této matice a ta byla značena jako k_{ij}. Rozepíšeme-li (2.15) blokově, stejně jako v jednorozměrném případě získáme

$$\begin{bmatrix} S_{11} & 0 \\ 0 & S_{22} \end{bmatrix}\begin{bmatrix} X_{11} & X_{12} \\ X_{21} & X_{22} \end{bmatrix} \approx \begin{bmatrix} B_{11} & 0 \\ 0 & 0 \end{bmatrix}.$$

Označme K součet všech hodností k_{ij}. Potom platí, že $S_{11} \in \mathbb{R}^{(K+k_0)\times K}$, posledních k_0 řádků je nulových a jedná se o diagonální matici s nenulovými singulárními čísly matice A na diagonále. Každé singulární číslo se opakuje tolikrát, jaká je hodnost příslušného bloku pravé strany $S_{ij}V_{ij}^T$. Matice S_{22} je také diagonální a obsahuje zbylá singulární čísla na diagonále.

Úlohu lze rozdělit na čtyři podúlohy, stejně jako v (2.3), kde $A_{11} = S_{11}$ a $A_{22} = S_{22}$. Jediná úloha s netriviálním řešením je tedy

$$S_{11}X_{11} \approx B_{11}.$$

Nyní ukážeme, že se skutečně jedná o core problém. K tomu nám poslouží následující věta.

Věta 3. *Uvažujme úlohu (2.1). Nechť $K \in \mathbb{N}$ je součet hodností všech matic $S_{ij}V_{ij}^T$ z (2.13) a k_0 je hodnost matice $S_0V_0^T$ z (2.12). Potom existují ortogonální matice P, Q a R splňující*

$$(P^TAQ)(Q^TXR) = \begin{bmatrix} S_{11} & 0 \\ 0 & S_{22} \end{bmatrix}\begin{bmatrix} X_{11} & X_{12} \\ X_{21} & X_{22} \end{bmatrix} \approx P^TBR = \begin{bmatrix} B_{11} & 0 \\ 0 & 0 \end{bmatrix}$$

a úloha $S_{11}X_{11} \approx B_{11}$ je core problémem v (2.1). Navíc $S_{11} \in \mathbb{R}^{(K+k_0)\times K}$ je diagonální matice s nenulovými singulárními čísly matice A na diagonále, každé opakující se tolikrát, kolik je hodnost příslušného bloku $S_{ij}V_{ij}^T$, a posledních k_0 řádků je nulových. Matice S_{22} je diagonální obsahující zbylá singulární čísla matice A na diagonále.

Důkaz. Existence matic P, Q a R plyne z konstrukce výše. Konkrétně $P = U_1 P_1^T$, $Q = V_1 P_2$ viz (2.15) a $R = V_B$ z kroku 1.

U důkazu minimality uvedeme pouze hlavní myšlenku. Důkaz v celém znění je možné nalézt v [11]. Nejprve dokážeme minimalitu pravé strany B_{11} a v druhé fázi minimalitu matice S_{11}.

Předpokládejme, že máme ortogonální matice S,T a Z splňující

$$(S^T A T)(T^T X Z) = \begin{bmatrix} \bar{A}_{11} & 0 \\ 0 & \bar{A}_{22} \end{bmatrix} (S^T X T) \approx S^T B Z = \begin{bmatrix} \hat{B}_{11} & 0 \\ 0 & 0 \end{bmatrix}, \tag{2.16}$$

kde $\bar{A}_{11} \in \mathbb{R}^{\hat{K}_0 \times \hat{K}}$ a $\hat{B}_{11} \in \mathbb{R}^{\hat{K}_0 \times \hat{d}}$. Nyní z rovnosti na pravé straně, konkrétně

$$\begin{bmatrix} \hat{B}_{11} & 0 \\ 0 & 0 \end{bmatrix} = S^T B Z,$$

plyne, že $rank(\hat{B}_{11}) = rank(B)$, což je počet sloupců matice B_{11} a tedy počet sloupců matice $\hat{B}_{11} = \hat{d}$ je minimálně stejný nebo větší. Tímto jsme dokázali, že naše transformace má minimální počet sloupců pravé strany. Stejně jako v jednorozměrném případě využijeme singulárních rozkladů

$$\bar{A}_{11} = \bar{U}_1 \bar{S}_{11} \bar{V}_1^T,$$
$$\bar{A}_{22} = \bar{U}_2 \bar{S}_{22} \bar{V}_2^T.$$

Na jejich základě získáme alternativní vyjádření singulárního rozkladu matice A

$$A = S \begin{bmatrix} \bar{U}_1 & 0 \\ 0 & \bar{U}_2 \end{bmatrix} \begin{bmatrix} \bar{S}_{11} & 0 \\ 0 & \bar{S}_{22} \end{bmatrix} \begin{bmatrix} \bar{V}_1^T & 0 \\ 0 & \bar{V}_2^T \end{bmatrix} T^T.$$

Další kroky důkazu směřují k rozdělení matice

$$\begin{bmatrix} \bar{U}_1^T \hat{B}_{11} \\ 0 \end{bmatrix}$$

na stejné bloky příslušící různým singulárním číslům jako rozdělení matice $\hat{U}^T B_1$ v (2.14). Výsledné vyjádření, které získáme, je

$$\begin{bmatrix} \bar{U}_1^T \hat{B}_{11} \\ 0 \end{bmatrix} = W_1 \begin{bmatrix} S_{1j} V_{1j}^T \\ \vdots \\ S_{ik} V_{ik}^T \\ S_0 V_0^T \end{bmatrix} W_2,$$

kde W_1 a W_2 jsou nějaké ortogonální matice. Prostřední matice je z (2.14), o které víme, že má $K + k_0$ nenulových řádků. Počet řádků matice $\bar{U}_1^T \hat{B}_{11}$, který je roven počtu řádků matice $\bar{A}_{11}$, je proto stejný nebo větší. Tímto jsme zjistili, že matice S_{11} má minimální počet řádků.

Posledním krokem je dokázat, že S_{11} má minimální počet sloupců. Tento fakt se odvodí z vyjádření

$$\begin{bmatrix} \bar{A}_{11} & 0 \\ 0 & \bar{A}_{22} \end{bmatrix} = S^T A T = S^T P \begin{bmatrix} S_{11} & 0 \\ 0 & S_{22} \end{bmatrix} Q^T T$$

a struktury matice $S^T P$ obdobnou úvahou o hodnostech jako v důkazu minimality pravé strany.

□

Na základě odvozené konstrukce core problému v této sekci a předchozí věty můžeme ukázat následující důsledky, které charakterizují vlastnosti core problému.

Důsledek 4. *Nechť $\bar{A}_{11}\bar{X}_{11} \approx \bar{B}_{11}$ je core probléme v úloze (1.1). Potom mají matice $\bar{A}_{11}$ a $\bar{B}_{11}$ plnou sloupcovou hodnost.*

Důkaz. Pro spor předpokládejme, že matice $\bar{A}_{11}$ nemá plnou sloupcovou hodnost. Označme $\bar{A}_{11} = \bar{U}_1\bar{S}_{11}\bar{V}_1^T$ singulární rozklad matice $\bar{A}_{11}$. Potom matice $\bar{S}_{11}$ obsahuje nulové sloupce a lze rozdělit na

$$\bar{S}_{11} = \begin{bmatrix} \bar{S}_{11,1} & 0 \end{bmatrix}.$$

Označme $\bar{P}$,$\bar{Q}$ a $\bar{R}$ ortogonální matice z bodu 1. v Definici 3. Potom ortogonální transformace

$$\bar{P}^T A \bar{Q} \begin{bmatrix} \bar{V}_1 & 0 \\ 0 & I \end{bmatrix} = \begin{bmatrix} \bar{U}_1\bar{S}_{11,1} & 0 & 0 \\ 0 & 0 & \bar{A}_{22} \end{bmatrix}$$
$$\bar{P}^T B = \begin{bmatrix} \bar{B}_{11} & 0 \\ 0 & 0 \end{bmatrix}$$

nám dá úlohu $(\bar{U}_1\bar{S}_{11,1})(\bar{V}_1^T X_{11}) \approx B_{11}$, která splňuje bod 1. z Definice 3 a matice $\bar{U}_1\bar{S}_{11,1}$ má menší počet sloupců, než matice $\bar{A}_{11}$, což je spor. Stejný postup lze použít v případě matice $\bar{B}_{11}$.

□

Důsledek 5. *Uvažujme úlohu (1.1). Dále mějme úlohu $\bar{A}_{11}\bar{X}_{11} \approx \bar{B}_{11}$, pro kterou existují ortogonální matice P,Q a R splňující*

$$P^T A Q (Q^T X R) = \begin{bmatrix} \bar{A}_{11} & 0 \\ 0 & \bar{A}_{22} \end{bmatrix} \begin{bmatrix} X_{11} & X_{12} \\ X_{21} & X_{22} \end{bmatrix} \approx P^T B R = \begin{bmatrix} \bar{B}_{11} & 0 \\ 0 & 0 \end{bmatrix}. \tag{2.17}$$

Označme $\bar{A}_{11} = \bar{U}_1\bar{S}_{11}\bar{V}_1^T$ singulární rozklad matice $\bar{A}_{11}$. Rozdělme matici $\bar{U}_1$ na bloky $\bar{U}_{1,i}$ obsahující báze levých singulárních prostorů příslušných singulárnímu číslu $\hat{s}_i$ a blok $\bar{U}_{1,0}$ obsahující bázi $N(\bar{A}_{11}^T)$. Potom je úloha $\bar{A}_{11}\bar{X}_{11} \approx \bar{B}_{11}$ core problémem v (1.1) právě tehdy, když jsou splněny následující tři podmínky

1. *Matice $\bar{A}_{11}$ má plnou sloupcovou hodnost.*
2. *Matice $\bar{B}_{11}$ má plnou sloupcovou hodnost.*
3. *Matice $\bar{U}_{1,l}^T\bar{B}_{11}$ má plnou řádkovou hodnost rovnou násobnosti singulárního čísla $\hat{s}_l$ matice $\bar{A}_{11}$ a matice $\bar{U}_{1,0}^T\bar{B}_{11}$ má plnou řádkovou hodnost rovnou $dim(N(\bar{A}_{11}^T))$.*

Důkaz. Stejně jako v důkazu Věty 3 použijeme alternativní vyjádření singulárního rozkladu matice A,

$$A = P \begin{bmatrix} \bar{U}_1 & 0 \\ 0 & \bar{U}_2 \end{bmatrix} \begin{bmatrix} \bar{S}_{11} & 0 \\ 0 & \bar{S}_{22} \end{bmatrix} \begin{bmatrix} \bar{V}_1^T & 0 \\ 0 & \bar{V}_2^T \end{bmatrix} Q^T.$$

Dále z důkazu této věty víme, že existují ortogonální matice matice W_1 a W_2 takové, že

$$\begin{bmatrix} \bar{U}_1^T \bar{B}_{11} \\ 0 \end{bmatrix} = W_1 \begin{bmatrix} S_{1j} V_{1j}^T \\ \vdots \\ S_{ik} V_{ik}^T \\ S_0 V_0^T \end{bmatrix} W_2 = W_1 P_1^T \begin{bmatrix} B_{11} \\ 0 \end{bmatrix} W_2. \tag{2.18}$$

Pokud je tedy úloha $\bar{A}_{11}\bar{X}_{11} \approx \bar{B}_{11}$ core problémem, pak z Důsledku 4, jsou splněny body 1. a 2. Na základě Věty 3 je počet řádků matice $\bar{A}_{11}$ a tedy i matice $\bar{U}_1^T \bar{B}_{11}$ roven $K + k_0$. Vyjádření (2.18) rozepsané po blocích nám proto říká, že řádková hodnost bloku $\bar{U}_{1,l}^T \bar{B}_{11}$ je rovna řádkové hodnosti bloku matice B_{11} tvořeným odpovídajícími řádky, což není nic jiného, než nenulové řádky z té matice $S_{ij}V_{ij}^T$, která přísluší singulárnímu číslu matice A rovnému $\hat{s}_l$. Obdobně hodnost bloku $\bar{U}_{1,0}^T \bar{B}_{11}$ odpovídá hodnosti bloku matice B_{11} obsahující nenulové řádky matice $S_0 V_0^T$. Odtud plyne bod 3.

Naopak nechť jsou splněny body 1.-3. Důkaz je analogií k důkazu Věty 3. V tomto důkazu jsme porovnávali rozměry úlohy (2.15) získané konstrukcí popsanou v této kapitole s obecnou úlohou typu (2.16). Pokud místo této obecné úlohy vezmeme (2.17) splňující body 1.-3., dojdeme stejným postupem k závěru, že úloha (2.17) musí mít stejné rozměry jako úloha ve Větě 3. Konkrétně bod 1. zajistí, že matice $\hat{A}_{11}$ má stejný počet sloupců jako S_{11}. Díky bodu 2. má matice $\bar{B}_{11}$ stejný počet sloupců jako matice B_{11}. Konečně bod 3. zajistí, že počet řádků matice $\bar{A}_{11}$ je stejný jako počet řádků matice S_{11}.

□

3. Iterační konstrukce core problému

Tato kapitola je zaměřena na další variantu konstrukce core problému v úloze

$$AX \approx B. \tag{3.1}$$

V minulé kapitole jsme rozebrali přímou konstrukci pomocí singulárního rozkladu. Nyní se bude jednat o iterační konstrukci založenou na Krylovovské iterační metodě zvané Golub-Kahanova bidiagonalizace. Nejprve proto představíme pojem Krylovova prostoru a obecnou Golub-Kahanovu bidiagonalizaci, která byla poprvé zavedena v [4] a rozšířena na vícerozměrný případ ($d > 1$) v [1]. Poté ji aplikujeme na úlohu (3.1) a pomocí ní získáme core problém v této aproximační úloze. Tato metoda konstrukce byla představena v [15] pro jednorozměrný případ a v [10] pro vícerozměrný. V této kapitole budeme často používat QR rozklad. Pro upřesnění uveďme, že QR rozkladem matice $A \in \mathbb{R}^{m\times n}$ rozumíme

$$A = QR, \quad Q \in \mathbb{R}^{m\times n}, \quad R \in \mathbb{R}^{n\times n},$$

přičemž matice Q má ortonormální sloupce a matice R je horní trojúhelníková.

3.1 Golub-Kahanova bidiagonalizace

Jak již bylo zmíněno, Golub-Kahanova bidiagonalizace je jednou z Krylovovských iteračních metod, proto zde nejprve zavedeme definici Krylovova prostoru. Problematika Krylovových prostorů a Krylovovských iteračních metod je rozebrána v [14].

Definice 4. *Nechť $Y \in \mathbb{R}^{n\times n}$ a $z \in \mathbb{R}^n$. Potom Krylovovým prostorem řádu k příslušným matici Y a vektoru z nazveme*

$$K_k(Y,z) = span\{z,Yz,\ldots,Y^{k-1}z\}.$$

Uvažujme počáteční vektory

$$s_1 = \frac{b}{||b||}, \quad w_0 = 0 \in \mathbb{R}^m.$$

Golub-Kahanova bidiagonalizace matice A pracuje s iteracemi

$$\alpha_j w_j = A^T s_j - \beta_j w_{j-1}, \tag{3.2}$$
$$\beta_{j+1} s_{j+1} = A w_j - \alpha_j s_j, \tag{3.3}$$

pro $j = 1,2,\ldots$. Koeficienty α_j a β_{j+1} jsou kladné normalizační koeficienty zajišťující, že $||w_j|| = 1$ a $||s_{j+1}|| = 1$. Nechť α_j a β_{j+1} jsou nenulové až do iterace k. Potom vektory $W_k = [w_1,\ldots,w_k]$ tvoří ortogonální bázi prostoru $K_k(A^TA,A^Tb)$ a vektory $S_k = [s_1,\ldots,s_k]$ tvoří ortogonální bázi prostoru $K_k(AA^T,b)$. V maticovém tvaru dostaneme

$$A^T S_k = W_k L_k^T,$$
$$AW_k = S_k L_k + \beta_{k+1} s_{k+1} e_k^T,$$

kde matice L_k je bidiagonální matice obsahující koeficienty α_j a β_j.

$$L_k = \begin{pmatrix} \alpha_1 & 0 & \dots & \dots & 0 \\ \beta_2 & \ddots & & & \vdots \\ 0 & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \beta_k & \alpha_k \end{pmatrix}.$$

Iterace popsané výše se zastaví, pokud dojdeme do situace, kdy buď pro nějaké j je $\alpha_j = 0$ nebo $\beta_{j+1} = 0$. V tomto případě je možné v iteracích pokračovat volbou nového nenulového vektoru w_j nebo s_{j+1}, který je ortogonální vůči všem předchozím již spočteným vektorům W_{j-1} nebo S_j (viz. [4]).

Poznámka. Golub-Kahanova bidiagonalizace je úzce spjatá s Lancozsovou tridiagonalizací [13] aplikovanou na symetrickou matici AA^T a vektor s_1. Tato metoda zkonstruuje ortogonální bázi $\hat{S}_k$ prostoru $K_k(AA^T,s_1)$ pomocí trojčlenné rekurence

$$\delta_{j+1}\hat{s}_{j+1} = AA^T\hat{s}_j - \gamma_j\hat{s}_j - \delta_j\hat{s}_{j-1}.$$

Více k této souvislosti lze nalézt v [16], Sekce 7.3.

Poznámka. Iterační konstrukci popsanou výše lze odvodit z přímé verze bidiagonalizace (lze nalézt v [4], Sekce 2). Tato přímá verze konstruuje dvě sady Householderových reflexí. První sada aplikovaná zleva nuluje, v našem případě, postupně poddiagonální prvky matice $[b,A]$ v jednotlivých sloupcích a druhá aplikovaná zprava nuluje prvky v řádcích nacházejících se od druhé naddiagonální dále. Výsledkem jsou tedy ortogonální matice P a Q splňující

$$P^T[b,A]Q = \begin{bmatrix} L \\ 0 \end{bmatrix},$$

kde posledních $m-(n+1)$ řádků je nulových a matice L je bidiagonální tvaru

$$L = \begin{pmatrix} \beta_1 & \alpha_1 & 0 & \dots & \dots & 0 \\ 0 & \beta_2 & \ddots & & & \vdots \\ \vdots & 0 & \ddots & \ddots & & \vdots \\ \vdots & \vdots & & \ddots & \ddots & 0 \\ \vdots & \vdots & \dots & 0 & \beta_n & \alpha_n \\ 0 & 0 & \dots & \dots & 0 & \beta_{n+1} \end{pmatrix}$$

Pokud je j index iterace, ve které se zastaví Golub-Kahanova iterační bidiagonalizace, potom se, při naší volbě počátečních hodnot, matice S_j, respektive W_{j-1}, shodují s prvními j, respektive $j-1$, sloupci matice P, respektive Q.

Golub-Kahanovu bidiagonalizaci lze zobecnit do blokové verze, kde v každé iteraci místo dvojice nových vektorů s_{j+1} a w_j spočteme celé dvě matice P_{j+1} a Q_j. Nejprve budeme muset představit blokové Krylovovi prostory. Více o vlastnostech blokových Krylovových prostorů a blokových iteračních metod je možné nalézt v [7].

Definice 5. *Nechť $Y \in \mathbb{R}^{n\times n}$ a $Z \in \mathbb{R}^{n\times m}$. Potom blokovým Krylovovým prostorem řádu k příslušným matici Y a Z nazveme*

$$K_k(Y,Z) = span\{Z,YZ,\ldots,Y^{k-1}Z\},$$

přičemž lineárním obalem množiny matic rozumíme

$$span\{Z,YZ,\ldots,Y^{k-1}Z\} = \{C \in \mathbb{R}^{n\times m} : C = \sum_{i=0}^{k-1} Y^i Z C_i, \quad C_i \in \mathbb{R}^{m\times m}\}.$$

Bloková Golub-Kahanova bidiagonalizace aplikovaná na matici A z úlohy (3.1) s počátečními maticemi

$$P_1,R_1, \quad kde \quad B = P_1R_1 \quad je\ QR\ rozklad\ matice\ B, \quad Q_0 = 0 \in \mathbb{R}^{n\times d},$$

produkuje ortogonální matice $P_1,\ldots,P_j$, které tvoří bázi blokového Krylovova prostoru $K_j(AA^T,B)$, a ortogonální matice $Q_1,\ldots,Q_j$ tvořící bázi blokového Krylovova prostoru $K_j(A^TA,A^TB)$. Iterace jsou blokovým zobecněním iteracím z (3.2)-(3.3). Mají tvar

$$\begin{aligned} \bar{Q}_j &= A^TP_j - Q_{j-1}R_j^T \\ \bar{Q}_j &= Q_jD_j^T \quad \text{(QR rozklad matice } \bar{Q}_j\text{)} \\ \bar{P}_{j+1} &= AQ_j - P_jD_j \\ \bar{P}_{j+1} &= P_{j+1}R_{j+1} \quad \text{(QR rozklad matice } \bar{P}_{j+1}\text{)} \end{aligned} \quad . \tag{3.4}$$

Předchozí iterace lze zapsat také ve tvaru

$$\begin{aligned} Q_jD_j^T &= A^TP_j - Q_{j-1}R_j^T, \\ P_{j+1}R_{j+1} &= AQ_j - P_jD_j, \end{aligned}$$

kde je více patrná analogie s původními iteracemi. Stejně jako jsme iterace (3.2)-(3.3) přepsali do maticového tvaru, zde můžeme udělat něco podobného. Označme $\hat{P}_k = [P_1,\ldots,P_k]$ a $\hat{Q}_k = [Q_1,\ldots,Q_k]$, potom můžeme (3.4) zapsat v podobě

$$\begin{aligned} A^T\hat{P}_k &= \hat{Q}_k\hat{L}_k^T, \\ A\hat{Q}_k &= \hat{P}_{k+1}\hat{L}_{k+1,k}. \end{aligned}$$

Matice $\hat{L}_k$ a $\hat{L}_{k+1,k}$ jsou blokově bidiagonální matice obsahující bloky R_j a D_j,

$$\hat{L}_k = \begin{pmatrix} D_1 & 0 & \ldots & \ldots & 0 \\ R_2 & \ddots & & & \vdots \\ 0 & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ 0 & \ldots & 0 & R_k & D_k \end{pmatrix}, \quad \hat{L}_{k+1,k} = \begin{pmatrix} D_1 & 0 & \ldots & 0 \\ R_2 & \ddots & & \vdots \\ 0 & \ddots & \ddots & \vdots \\ \vdots & & \ddots & D_k \\ 0 & \ldots & 0 & R_{k+1} \end{pmatrix}.$$

Navíc bloky R_j jsou horní trojúhelníkové matice a D_j jsou dolní trojúhelníkové matice a tedy matice $\hat{L}_k$ i $\hat{L}_{k+1,k}$ mají nenulové prvky pouze na hlavní diagonále a dalších d poddiagonálách. Jsou to tak zvané pásové matice.

Iterace (3.2)-(3.3) se zastavili, pokud se jeden z koeficientů α_j nebo β_{j+1} rovnal nule. Analogií k této situaci v blokovém případě je případ, kdy v nějaké iteraci j nemá matice $\bar{Q}_j$ nebo $\bar{P}_{j+1}$ plnou sloupcovou hodnost a obsahuje tedy alespoň jeden lineárně závislý sloupec. Nechť je i-tý sloupec matice $\bar{Q}_j$ lineární kombinací předchozích sloupců. Potom, spočítáme-li v dalším kroku iterace QR rozklad této matice, $\bar{Q}_j = Q_j D_j^T$, bude i-tý řádek matice D_j^T nulový. Je tedy možné vynechat i-tý sloupec matice Q_j a i-tý řádek matice D_j^T. Označíme $Q_j = [q_1, \ldots, q_d]$ sloupce matice Q_j a $D_j = [d_1, \ldots, d_d]$ řádky matice D_j^T. Potom platí

$$\bar{Q}_j = \begin{bmatrix} p_1 & \ldots & p_{i-1} & p_{i+1} & \ldots & p_d \end{bmatrix} \begin{bmatrix} d_1 \\ \vdots \\ d_{i-1} \\ d_{i+1} \\ \vdots \\ d_d \end{bmatrix}.$$

Obecně pro každý sloupec matice $\bar{Q}_j$, který je lineární kombinací předchozích, vynecháme příslušný sloupec matice Q_j a nulový řádek matice D_j^T. Za matice Q_j a D_j^T v iteracích (3.4) dosadíme matice Q_j, resp. D_j^T s vynechanými sloupci, resp. řádky a pokračujeme dále. Tímto snížíme rozměry všech dalších matic D_i^T, R_i, Q_i a P_i spočtených v následujících iteracích. Konkrétně, pokud má matice $\bar{Q}_j$ právě l sloupců, které jsou lineární kombinací předchozích, získáme, že rozměry zredukovaných matic Q_j a D_j^T jsou po řadě $n \times (d-l)$ a $(d-l) \times d$. Z iterací (3.4) vidíme, že potom

$$\begin{aligned} P_{j+1} &\in \mathbb{R}^{m\times(d-l)}, \quad R_{j+1} \in \mathbb{R}^{(d-l)\times(d-l)}, \\ Q_{j+1} &\in \mathbb{R}^{n\times(d-l)}, \quad D_{j+1}^T \in \mathbb{R}^{(d-l)\times(d-l)}. \end{aligned} \tag{3.5}$$

Jinými slovy se sníží rozměry všech dále spočtených matic. Tomuto jevu se říká horní deflace. Stejný postup nastává i pokud má matice $\bar{P}_{j+1}$ l lineárně závislých sloupců. V takovém případě mluvíme o dolní deflaci. Zde vynecháme l nulových řádků matice R_{j+1} a příslušné sloupce matice P_{j+1}. Zredukované matice R_{j+1} a P_{j+1} mají po řadě rozměry $m \times (d-l)$ a $(d-l) \times d$. Z iterací (3.4) opět snadno nahlédneme, že

$$\begin{aligned} Q_{j+1} &\in \mathbb{R}^{n\times(d-l)}, \quad D_{j+1}^T \in \mathbb{R}^{(d-l)\times(d-l)}, \\ P_{j+2} &\in \mathbb{R}^{m\times(d-l)}, \quad R_{j+2} \in \mathbb{R}^{(d-l)\times(d-l)}, \end{aligned} \tag{3.6}$$

a tedy i zde dojde k snížení rozměrů všech následujících matic. Tyto deflace budou hrát podstatnou roli při konstrukci core problému ve vícerozměrném případě pomocí blokové Golub-Kahanovi bidiagonalizace a budeme se jim v této sekci více věnovat.

Poznámka. Deflace popsané výše souvisí s faktem, že dimenze Krylovova prostoru $K_{j+1}(A,B)$ nemusí být o d vyšší než dimenze předchozího $K_j(A,B)$. Tím rozumíme, že matice A^jB může obsahovat sloupce, které jsou lineární kombinací sloupců předchozích matic A^iB pro $i < j$. Více k tomuto tématu se lze dočíst v [8].

3.2 Konstrukce pro úlohy s jednonásobnou pravou stranou

V této sekci se zaměříme na konstrukci core problému v úloze (1.1), pro $d = 1$, za pomoci Golub-Kahanovi bidiagonalizace, představené v předchozí části. Využití bidiagonalizace bylo uvedeno poprvé v [15]. Přesné využití Golub-Kahanovi bidiagonalizace je poté přesněji rozebráno v [10].

Začneme tím, že aplikujeme Golub-Kahanovu bidiagonalizaci na matici A. Pro $j = 1,2,\ldots$ počítáme iterace (3.2)-(3.3). Jako počáteční vektory použijeme

$$s_1 = \frac{b}{||b||}, \quad w_0 = 0, \quad \text{kde} \quad \beta_1 = ||b||.$$

Tyto iterace se zastaví v nějakém kroku j, pokud nastane jedna ze dvou možností. Buďto $\alpha_j = 0$ nebo $\beta_{j+1} = 0$. Poznamenejme, že za předpokladu $b \not\perp R(A)$ máme zajištěno, že alespoň $\alpha_1 \neq 0$. Dojdeme tedy k jedné z následujících možností:

$$S_j^T A W_{j-1} = L_{j,j-1} = \begin{pmatrix} \alpha_1 & 0 & \ldots & 0 \\ \beta_2 & \ddots & & \vdots \\ 0 & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \alpha_{j-1} \\ 0 & \ldots & 0 & \beta_j \end{pmatrix} \tag{3.7}$$

pokud $\alpha_j = 0$, nebo

$$S_j^T A W_j = L_j = \begin{pmatrix} \alpha_1 & 0 & \ldots & \ldots & 0 \\ \beta_2 & \ddots & & & \vdots \\ 0 & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ 0 & \ldots & 0 & \beta_j & \alpha_j \end{pmatrix} \tag{3.8}$$

pokud $\beta_{j+1} = 0$. Matice S_j má vzájemně ortogonální sloupce a její první sloupec je roven $\beta_1^{-1} b$. Z tohoto vyplývá, že

$$S_j^T b = \begin{pmatrix} \beta_1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}.$$

V případě (3.7) vidíme, že soustava

$$(S_j^T A W_{j-1})(W_{j-1}^T x) \approx S_j^T b$$

není kompatibilní, neboť $S_j^T b \notin R(S_j^T A W_{j-1})$. Naopak pro situaci z (3.8) je soustava

$$(S_j^T A W_j)(W_j^T x) \approx S_j^T b$$

kompatibilní, jelikož $rank(S_j^T A W_j) = j$.

Dále se zaměříme pouze na nekompatibilní případ (3.7). Tato situace nastane právě tehdy, když byla původní úloha $Ax \approx b$ také nekompatibilní. Pokud doplníme matici S_j (resp. matici W_{j-1}) o sloupce $S_{j+1,m} = (s_{j+1}, \ldots, s_m)$ (resp. $W_{j,n} = (w_j, \ldots w_n)$) tak, aby byli výsledné matice $S_m = [S_j, S_{j+1,m}]$ a $W_n = [W_{j-1}, W_{j,n}]$ stále ortogonální, získáme tím transformaci úlohy (3.1) tvaru

$$(S_m^T A W_n)(W_n^T x) = \begin{bmatrix} L_{j,j-1} & 0 \\ 0 & A_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \approx S_m^T b = \begin{bmatrix} b_1 \\ 0 \end{bmatrix}. \tag{3.9}$$

Zde má vektor $b_1 \in \mathbb{R}^j$ pouze jeden nenulový prvek roven β_1 na první pozici. Získali jsme transformaci tvaru (2.4), kde $A_{11} = L_{j-1,j}$ je bidiagonální matice obsahující pouze nenulové prvky na hlavní diagonále a první poddiagonále. Navíc úloha

$$L_{j,j-1} x_1 \approx b_1$$

je, jak dále ukážeme, skutečně core problémem v úloze $Ax \approx b$.

Poznámka. V praktickém výpočtu není třeba konstruovat matice $S_{j+1,m}$ a $W_{j,n}$. Jakmile dojdeme do iterace, kdy α_j nebo β_{j+1} je 0, proces ukončíme a získáme tím $L_{j-1,j}$ a vektory x_1 a b_1, což je všechny data tvořící core problém.

K důkazu, že se opravdu jedná o core problém, budeme nejprve potřebovat pomocné lemma, které je možné nalézt v [15] Lemma 3.1. a Theorem 3.2.

Lemma 6. *Mějme matici $L_{j,j-1}$ a vektor b_1 z (3.9). Označme $L_{j,j-1} = U_L S_L V_L^T$ singulární rozklad matice $L_{j,j-1}$, $U_L = [u_L^1, \ldots, u_L^j]$ a $S_L = diag(s_L^1, \ldots, s_L^{j-1})$. Potom platí následující tvrzení:*

1. *Všech $j-1$ singulárních čísel s_L^i matice $L_{j,j-1}$ je nenulových a unikátních.*
2. *Pro všechna $i \in \{1, \ldots, j-1\}$ platí $(u_L^i)^T b_1 \neq 0$.*

Důkaz. (1): Matice $L_{j,j-1} \in \mathbb{R}^{j \times (j-1)}$ je bidiagonální a obsahuje pouze nenulové prvky na hlavní diagonále a první poddiagonále. Z tohoto okamžitě plyne, že $rank(L_{j,j-1}) = j-1$. To znamená, že má $j-1$ nenulových singulárních čísel. Navíc matice $L_{j,j-1}^T L_{j,j-1}$ vznikne z matice $[b_1, L_{j,j-1}]^T [b_1, L_{j,j-1}]$ odebráním prvního sloupce a prvního řádku. Obě matice jsou symetrické, tridiagonální a obsahují pouze nenulové prvky na první poddiagonále a první naddiagonále. Tedy z věty o prokládání vlastních čísel ([20], strana 300) získáme, že singulární čísla matice $L_{j,j-1}$ musí být unikátní.

(2): Vztah $(u_L^i)^T b_1 \neq 0$ platí právě tehdy, když je první složka vektoru u_L^i nenulová, neboť $b_1 = \beta_1 e_1$. Označme $V_L = [v_L^1, \ldots, v_L^{j-1}]$. Z vlastnosti pravých a levých singulárních vektorů získáme pro $i \in \{1, \ldots, j-1\}$ vztahy

$$\begin{aligned} L_{j,j-1} v_L^i &= s_L^i u_L^i, \\ (u_L^i)^T L_{j,j-1} &= s_L^i (v_L^i)^T. \end{aligned}$$

Rozepíšeme-li první vztah po složkách dostaneme rovnice

$$\begin{aligned} \alpha_1 (v_L^i)_1 &= s_i (u_L^i)_1, \\ \beta_2 (v_L^i)_1 + \alpha_2 (v_L^i)_2 &= s_i (u_L^i)_2, \\ &\vdots \\ \beta_j (v_L^i)_{j-1} &= s_i (u_L^i)_j. \end{aligned} \tag{3.10}$$

Obdobně z druhého vztahu dojdeme k rovnicím

$$\begin{aligned} \alpha_1(u_L^i)_1 + \beta_2(u_L^i)_2 &= s_i(v_L^i)_1, \\ &\vdots \\ \alpha_{j-1}(u_L^i)_{j-1} + \beta_j(u_L^i)_j &= s_i(v_L^i)_{j-1}. \end{aligned} \tag{3.11}$$

Nyní pokud $(u_L^i)_1 = 0$, pak z (3.10) je i $(v_L^i)_1 = 0$. A tedy opakovaným používáním vztahů z (3.10) a (3.11) dojdeme k tomu, že $u_L^i = 0$. Jinými slovy celý levý singulární vektor příslušný nenulovému singulárnímu číslu s_i je nulový, což je spor. Dokázali jsme, že $(u_L^i)_1 \neq 0$ a tedy i $(u_L^i)^T b_1 \neq 0$. □

S tímto Lemmatem jsme nyní schopni přejít k hlavní větě sekce. Dokazující, že konstrukce popsaná výše skutečně vede na core problém. Toto tvrzení je možné nalézt v [15], Věta 3.3.

Věta 7. *Uvažujme úlohu (1.1) pro $d = 1$. Nechť $b \notin R(A)$ a $b \not\perp R(A)$. Dále nechť j je index, kdy se zastaví iterace (3.2)-(3.3) aplikované na matici A s počátečními vektory*

$$s_1 = \beta_1^{-1} b, \quad w_0 = 0, \quad kde \quad \beta_1 = ||b||.$$

V tomto případě $\alpha_j = 0$. Potom ortogonální transformace soustavy (1.1) z (3.9) splňuje, že úloha

$$L_{j,j-1} x_1 \approx b_1$$

je core problémem v (1.1) pro případ $d = 1$.

Důkaz. Bod (1) z Definice 3, zjednodušený pro případ $d = 1$ v (2.4), je triviálně splněn pro matice $P = S_m$ a $Q = W_n$. Důkaz minimality bude mít následující kroky

1. Ukážeme, že jsou splněny předpoklady Věty 2 pro $p = j - 1$.
2. Z Věty 2 získáme, že úloha má minimální rozměry.

(1): Označme

$$\begin{aligned} L_{j,j-1} &= U_L S_L V_L^T, \\ A_{22} &= U_{22} S_{22} V_{22}^T, \end{aligned}$$

singulární rozklady matic $L_{j,j-1}$ a A_{22}. Potom

$$A = S_m \begin{bmatrix} U_L S_L V_L^T & 0 \\ 0 & U_{22} S_{22} V_{22}^T \end{bmatrix} W_n^T = S_m \begin{bmatrix} U_L & 0 \\ 0 & U_{22} \end{bmatrix} \begin{bmatrix} S_L & 0 \\ 0 & S_{22} \end{bmatrix} \begin{bmatrix} V_L^T & 0 \\ 0 & V_{22}^T \end{bmatrix} W_n^T.$$

Získali jsme tedy singulární rozklad matice A. Singulární čísla matic $L_{j,j-1}$ a A_{22} dohromady tvoří všechna singulární čísla matice A. Navíc z Lemma 6 víme, že matice $L_{j,j-1}$ má $j-1$ nenulových a unikátních singulárních čísel, a jim příslušné levé singulární vektory matice A jsou $S_j U_L$. Připomeňme, že $S_m = [S_j, S_{j+1,m}]$. Potom platí

$$(S_j U_L)^T b = U_L^T S_j^T b = U_L^T b_1.$$

Z Lemma 6 víme, že pro $i \in \{1, \ldots, j-1\}$ platí, že $(u_L^i)^T b_1 \neq 0$. Odtud plyne, že b má nenulovou projekci na alespoň $j-1$ levých singulárních prostorů matice A příslušných různým nenulovým singulárním číslům. Navíc pro všechny zbylé levé singulární vektory $S_{j+1,m}U_{22}$ platí

$$(S_{j+1,m}U_{22})^T b = U_{22}^T(S_{j+1,m}^T b) = 0,$$

neboť vektory obsažené v matici $S_{j+1,m}$ jsou všechny kolmé na $s_1 = \beta^{-1}b$. Celkem jsem tedy ukázali, že vektor b není kolmý na právě $j-1$ levých singulárních prostorů matice A příslušných různým nenulovým singulárním číslům. Tímto jsme ověřili předpoklady Věty 2 pro $p = j-1$.

(2): Z Věty 2 víme, že rozměry core problému v úloze (1.1) jsou, v našem případě, $A_{11} \in \mathbb{R}^{j\times(j-1)}$ a $b_1 \in R^j$. Matice $L_{j,j-1}$ a vektor b_1 z (3.9) tedy mají minimální rozměry a úloha

$$L_{j,j-1}x_1 \approx b_1$$

tvoří core problém v úloze $Ax \approx b$.

□

Soustavu v (3.9) získanou Golub-Kahanovou iterační bidiagonalizací lze ortogonální transformací převést do tvaru uvedeného ve Větě 2. Použijeme k tomu singulární rozklady matic $L_{j,j-1}$ a A_{22} z důkazu předchozí věty. Snadno nahlédneme, že platí

$$\begin{bmatrix} U_L^T & 0 \\ 0 & U_{22}^T \end{bmatrix} S_m^T A W_n \begin{bmatrix} V_L & 0 \\ 0 & V_{22} \end{bmatrix} = \begin{bmatrix} S_L & 0 \\ 0 & S_{22} \end{bmatrix},$$
$$\begin{bmatrix} U_L^T & 0 \\ 0 & U_{22}^T \end{bmatrix} S_m^T b = \begin{bmatrix} U_L^T b_1 \\ 0 \end{bmatrix}.$$

Soustava (3.9) po transformaci tedy vypadá následovně

$$\begin{bmatrix} S_L & 0 \\ 0 & S_{22} \end{bmatrix} \begin{bmatrix} V_L^T x_1 \\ V_{22}^T x_2 \end{bmatrix} \approx \begin{bmatrix} U_L^T b_1 \\ 0 \end{bmatrix}.$$

Z Lemmatu 6 a Věty 7 víme, že S_L obsahuje $j-1$ různých nenulových singulárních čísel matice A a matice S_{22} obsahuje všechna zbylá singulární čísla. Navíc víme, že vektor b není kolmý právě na $j-1$ levých singulárních prostorů matice A příslušných singulárním číslům obsaženým v matici S_L. Vektor $U_L^T b_1$ obsahuje pouze nenulové prvky. Získali jsme tedy úlohu ve tvaru popsaném ve Větě 2.

Poznámka. V případě, kdy je úloha (1.1) kompatibilní, nastane situace (3.8) při aplikaci Golub-Kahanovi iterační bidiagonalizace. V této situaci lze analogicky, jako v případě (3.7), dokázat, že úloha

$$L_j x_1 \approx b_1$$

je core problémem v úloze $Ax \approx b$.

3.3 Konstrukce pro úlohy s vícenásobnou pravou stranou

Zde zobecníme konstrukci popsanou v předchozí sekci na případ, kdy $d > 1$. Tento postup je popsán v [10]. Využijeme blokovou Golub-Kahanovu bidiagonalizaci. Prvním krokem bude, stejně jako v Sekci 2.3, zredukovat pravou stranu B

tak, aby neobsahovala lineárně závislé sloupce. Opět existuje více způsobů. My zvolíme, stejně jako v Sekci 2.3, postup založený na SVD rozkladu matice B. S využitím stejného značení je

$$B = U_B S_B V_B^T$$

singulární rozklad matice B. Matici S_B rozdělíme na bloky

$$S_B = \begin{bmatrix} S_{B1} & 0 \end{bmatrix}.$$

Nyní aplikujeme matici V_B zleva na úlohu (1.1) a při stejném rozdělení matice XV_B na bloky X_1 a X_0, jako v Sekci 2.3, získáme

$$AX_1 \approx U_B S_{B1} = B_1, \quad AX_0 \approx 0.$$

Dále tedy budeme pracovat s úlohou $AX_1 \approx B_1$.

Cílem je použít iterace (3.4) na matici A. K získání počátečních matic nejprve spočteme QR rozklad matice B_1,

$$B_1 = P_1 R_1.$$

Označme $\hat{d}$ počet sloupců matice B_1. Nyní můžeme matice P_1 a R_1 společně s volbou $Q_0 = 0 \in \mathbb{R}^{n \times \hat{d}}$ zvolit jako počáteční matice pro iterace (3.4). Tento proces běží, dokud nedojdeme do iterace j, kdy buď $\bar{Q}_j$ nebo $\bar{P}_{j+1}$ nemá plnou sloupcovou hodnost. Nyní rozebereme postupně tyto případy

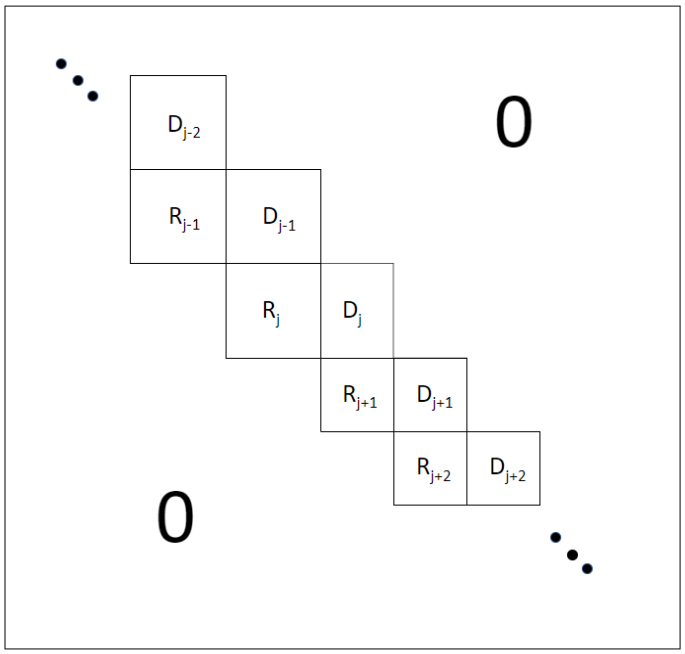
1. V iteraci j je hodnost matice $\bar{Q}_j < \hat{d}$.

2. V iteraci j je hodnost matice $\bar{P}_{j+1} < \hat{d}$.

Případ 1: Nechť v iteraci j má matice $\bar{Q}_j$ sloupcovou hodnost $rank(\bar{Q}_j) = \hat{d} - l$ pro nějaké $l \in \{1, \ldots, \hat{d}\}$. Potom, jak již víme, bude matice D_j^T z QR rozkladu matice $\bar{Q}_j$ obsahovat l nulových řádků, a proto tyto řádky odstraníme. Spolu s nimi odstraníme i příslušné sloupce matice Q_j. Označme $D_{j,l}^T$ matici D_j po odstranění nulových řádků a analogicky $Q_{j,l}$ matici Q_j po odstranění příslušných sloupců. Potom se tedy kroky v j-té iteraci změní na

$$\begin{aligned} \bar{Q}_j &= A^T P_j - Q_{j-1} R_j^T, \\ \bar{Q}_j &= Q_j D_j^T, \\ Q_j &= Q_{j,l}, \quad D_j^T = D_{j,l}^T, \\ \bar{P}_{j+1} &= AQ_j - P_j D_j, \\ \bar{P}_{j+1} &= P_{j+1} R_{j+1}. \end{aligned} \tag{3.12}$$

Tento proces způsobí horní deflaci. Snížení dimenze všech následujících matic spočtených v dalších iteracích, jak bylo popsáno v (3.5). Názorně je to ilustrováno na obrázku (3.3), kde vidíme zmenšující se bloky na diagonálách matice $\hat{L}_{k,k+1}$. Obrázek znázorňuje horní deflaci v j-té iteraci, při tomto jevu se zmenší počet nenulových diagonál matice $\hat{L}_{k,k+1}$ o l shora. Matice D_j má o l sloupců méně než předchozí matice D_i a R_{i+1} pro $i < j$ a všechny následující matice D_i a R_i pro $i > j$ mají i o l řádků méně.



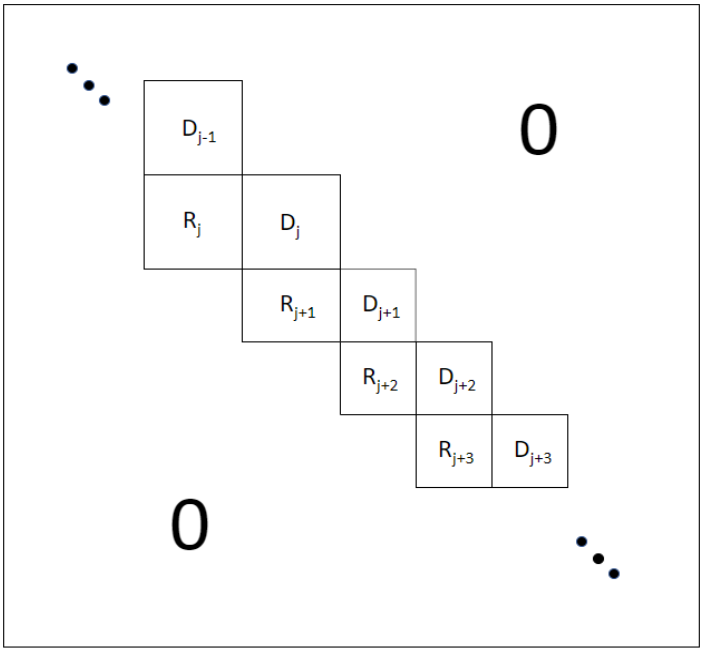
Obrázek 3.1: Horní deflace

Případ 2: Nyní předpokládáme, že v iteraci j má matice $\bar{P}_{j+1}$ sloupcovou hodnost $rank(\bar{P}_j) = \hat{d} - l$ pro nějaké $l \in \{1, \ldots, \hat{d}\}$. Matice R_{j+1} obsahuje l nulových řádků. Tyto řádky spolu s příslušnými sloupci matice P_{j+1} vynecháme. Označme $R_{j+1,l}$ matici R_{j+1} po vynechání všech l nulových řádků a $P_{j+1,l}$ matici P_{j+1} po vynechání příslušných l sloupců. Iterace j bude mít tedy tvar

$$\begin{aligned} \bar{Q}_j &= A^T P_j - Q_{j-1} R_j^T, \\ \bar{Q}_j &= Q_j D_j^T, \\ \bar{P}_{j+1} &= A Q_j - P_j D_j, \\ \bar{P}_{j+1} &= P_{j+1} R_{j+1}, \\ P_{j+1} &= P_{j+1,l}, \quad R_{j+1} = R_{j+1,l}. \end{aligned} \tag{3.13}$$

I zde se rozměry dalších matic spočtených iteracemi (3.4) zmenší, jak je popsáno v (3.6). Tento jev se nazývá dolní deflace. Opět to ilustrujeme zmenšováním bloků na diagonálách matice $\hat{L}_{k,k+1}$ na obrázku (3.3), na kterém vidíme dolní deflaci v j-té iteraci. V tomto případě se sníží počet nenulových diagonál matice $\hat{L}_{k,k+1}$ o l zespodu. Matice R_{j+1} má o l řádků méně než předchozí matice D_i a R_i pro $i < j + 1$ a všechny následující matice D_i a R_{i+1} pro $i > j$ mají i o l sloupců méně.

Nyní jsme v situaci, kdy v j-té iteraci nastala horní nebo dolní deflace a snížili se tím rozměry následujících bloků R_i a D_i o l sloupců a řádků. V iteracích (3.4), s případnými úpravami (3.12) nebo (3.13), pokud nastane další horní, respektive dolní deflace, pokračujeme tak dlouho, dokud nenastane tolik horních a dolních deflací, že další spočtená matice D_k^T nebo R_{k+1} je už pouze rozměrů 1×1 a rovná 0. Obrázek (3.3) toto ukazuje na případu, kdy iterace skončí iterací číslo 6, ve které $D_6^T = 0 \in \mathbb{R}$, přičemž v iteracích 2 a 5 nastane dolní deflace a v krocích 3 a 6 horní deflace. V tuto chvíli se iterace zastaví. Nechť se tak stane v iteraci k a



Obrázek 3.2: Dolní deflace

$D_k^T = 0 \in \mathbb{R}$. V tuto chvíli tedy máme spočtenou matici $\hat{L}_{k,k-1}$ a matice

$$\hat{P}_k = [P_1, \ldots, P_k],$$
$$\hat{Q}_{k-1} = [Q_1, \ldots, Q_{k-1}].$$

Navíc platí vztah

$$\hat{P}_k^T A \hat{Q}_{k-1} = \hat{L}_{k,k-1}.$$

Matice $\hat{P}_k$ obsahuje bloky, které jsou navzájem ortogonální a první blok P_1 pochází z QR rozkladu matice B_1. Odtud plyne, že

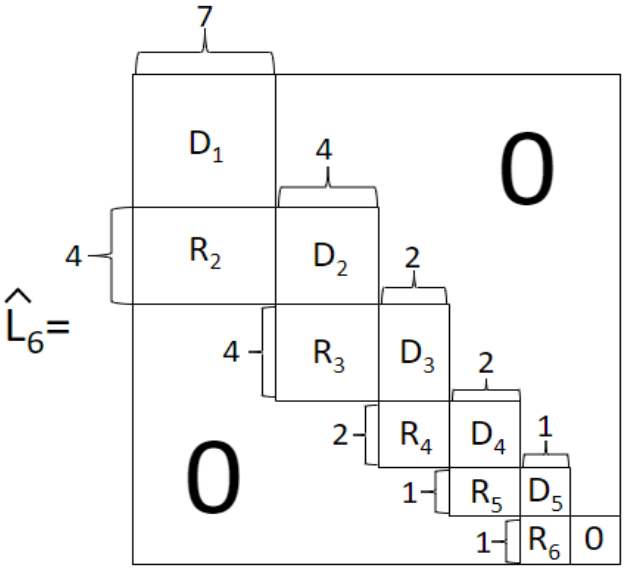
$$\hat{P}_k^T B_1 = (\hat{P}_k^T P_1) R_1 = \begin{bmatrix} R_1 \\ 0 \end{bmatrix}.$$

Pokud bychom analogicky k jednorozměrnému případu doplnily matice $\hat{P}_k$ a $\hat{Q}_{k-1}$ o bloky P_m a Q_n takové, že jsou ortogonální vůči všem ostatním blokům P_i, respektive Q_i a matice $\hat{Q}_n = [\hat{Q}_{k-1}, Q_n]$ a $\hat{P}_m = [\hat{P}_k, P_m]$ mají po řadě n a m sloupců, získali bychom

$$\hat{P}_m^T A \hat{Q}_n = \begin{bmatrix} L_{k,k-1} & 0 \\ 0 & A_{22} \end{bmatrix}, \quad \hat{P}_m^T B_1 = \begin{bmatrix} R_1 \\ 0 \end{bmatrix}.$$

Nyní spojíme vše dohromady. Nejprve jsme zredukovali pravou stranu, aby neobsahovala lineárně závislé sloupce aplikováním matice V_B zleva na úlohu (1.1),

$$A \begin{bmatrix} X_1 & X_0 \end{bmatrix} \approx \begin{bmatrix} B_1 & 0 \end{bmatrix}.$$



Obrázek 3.3: Průběh iterací s deflacemi

Poté jsme pomocí blokové Golub-Kahanovi bidiagonalizace s deflacemi zkonstruovali matice $\hat{P}_k$ a $\hat{Q}_{k-1}$ a doplnili je na matice $\hat{P}_m$ a $\hat{Q}_n$. Pomocí těchto matic pak získáme transformaci

$$\hat{P}_m^T A \hat{Q}_n (\hat{Q}_n^T \begin{bmatrix} X_1 & X_0 \end{bmatrix}) = \begin{bmatrix} \hat{L}_{k,k-1} & 0 \\ 0 & A_{22} \end{bmatrix} \hat{Q}_n^T \begin{bmatrix} X_1 & X_0 \end{bmatrix} \approx \hat{P}_m^T \begin{bmatrix} B_1 & 0 \end{bmatrix} = \begin{bmatrix} R_{1,0} & 0 \\ 0 & 0 \end{bmatrix}$$

s maticí $R_{1,0}$, což je R_1 doplněná o tolik nulových řádků, aby měla stejný počet řádků jako matice $\hat{L}_{k,k-1}$. Rozdělíme-li matici $\hat{Q}_n^T[X_1, X_0]$ na bloky odpovídající blokům pravé strany získáme

$$\begin{bmatrix} \hat{L}_{k,k-1} & 0 \\ 0 & A_{22} \end{bmatrix} \begin{bmatrix} X_{11} & X_{12} \\ X_{21} & X_{22} \end{bmatrix} \approx \begin{bmatrix} R_{1,0} & 0 \\ 0 & 0 \end{bmatrix}. \tag{3.14}$$

Navíc, jak ukážeme v další větě, úloha

$$\hat{L}_{k,k-1} X_{11} \approx R_{1,0}$$

je core problémem v úloze (1.1).

Poznámka. Analogicky k jednorozměrnému případu není nutné v praktickém výpočtu dopočítávat matice P_m a Q_n, jelikož nejsou třeba k výpočtu core problému.

Poznámka. Předpokládali jsme, že se iterace zastaví když $D_k^T = 0 \in \mathbb{R}$. Pokud by nastal druhý možný případ, tedy $R_{k+1} = 0 \in \mathbb{R}$, obdobně dojdeme ke stejné transformaci jako v (3.14) s maticí $\hat{L}_k$ namísto $\hat{L}_{k,k-1}$ a úloha

$$\hat{L}_k X_{11} \approx R_{1,0}$$

bude tvořit core problém.

Věta 8. *Uvažujme úlohu (1.1). Nechť k je index, ve kterém* $D_k^T = 0 \in \mathbb{R}$ *a tedy se zastaví iterace (3.4) aplikované na matici A s počátečními maticemi* $Q_0 = 0 \in \mathbb{R}^{n \times \hat{d}}$ *a* P_1,R_1 *z QR rozkladu matice B. Potom ortogonální transformace úlohy (1.1) z (3.14) splňuje, že*

$$\hat{L}_{k,k-1} X_{11} \approx R_{1,0}$$

je core problémem v (1.1).

Důkaz. První bod z Definice 3 je splněn triviálně pro $P = \hat{P}_m$, $Q = \hat{Q}_n$ a $R = V_B$.

K důkazu minimálních rozměrů použijeme Důsledek 5. Dokážeme, že úloha $\hat{L}_{k,k-1} X_{11} \approx R_{1,0}$ splňuje body 1.-3. Body 1.-2. jsou triviálně splněny z vlastností matic $\hat{L}_{k,k-1}$ a $R_{1,0}$. Důkaz, že je splněn i bod 3., je složitější a nebudeme ho zde uvádět. Lze ho nalézt v [10] Sekce 4.1 a 4.2.

□

4. Numerické experimenty

Tato kapitola je věnována testování klasického TLS algoritmu, popsaného v Kapitole 1, Sekci 1.3, a algoritmu pro výpočet core problému založeného na iterační konstrukci popsané v Kapitole 3, Sekci 3.3. Prozkoumáme chování algoritmů v závislosti na velikosti úlohy a velikosti samotného core problému. Dále také srovnáme aproximace řešení úlohy (1.1) získané přímou aplikací klasického TLS algoritmu s aproximacemi obdrženými použitím klasického TLS algoritmu na core problém v (1.1) s následnou zpětnou transformací. Všechny algoritmy byly implementovány v prostředí Matlab R2020b.

4.1 Popis implementace

V této sekci popíšeme naši konkrétní implementaci klasického TLS algoritmu a iterační konstrukci core problému. Dále vysvětlíme, jakým způsobem bylo provedeno generování testovacích úloh.

4.1.1 Klasický TLS algoritmus

Tento algoritmus je implementován podle popisu v Kapitole 1, Sekci 1.3. Vstupními parametry jsou matice A a B z úlohy (1.1) a kladné reálné číslo tol, které slouží k výběru singulárního prostoru v kroku 2. V naší implementaci používáme druhou variantu výběru singulárního podprostoru popsanou v kroku 2. Výstupem je matice X, řešení úlohy $AX \approx B$. Toto řešení je, v závislosti na klasifikaci úlohy, buďto řešení ve smyslu úplných nejmenších čtverců nebo negenerické řešení. Druhý výstupní parametr $flag$ informuje o tom, jaké řešení jsme získali. Pseudo kód je popsán v Algoritmu 1.

4.1.2 Iterační konstrukce core problému

Algoritmus je založen na konstrukci popsané v Kapitole 3, Sekci 3.3. Na vstupu jsou zadány matice A a B, dále kladný parametr tol sloužící k odhalovaní deflací. Nejprve jsou pomocí singulárního rozkladu odstraněny lineárně závislé sloupce matice B. Dále se jedná o přímou implementaci iterací (3.12) a (3.13), přičemž deflace jsou odhalovány na základě hledání "téměř"nulových řádků v maticích D_j a R_j pomocí kritéria

$$\max_i |d_{ki}| < tol.$$

Projdeme všechny řádky matice D_j, respektive R_j, a pokud v nějakém řádku mají všechny prvky velikost menší než tol, považujeme jej za nulový a odstraníme. Společně s ním odstraníme také příslušný sloupec matice Q_j, respektive P_j. Iterace probíhají tak dlouho, dokud není nějaká z matic D_j nebo R_j rovna $0 \in \mathbb{R}$. Výstupem algoritmu je matice L a matice $B1$ tvořící core problém v úloze $AX \approx B$ a ortogonální matice $\hat{Q}$ a V_B potřebné k zpětné transformaci řešení X. Níže v Algoritmu 2 je uveden příslušný pseudo kód.

Algorithm 1 Klasický TLS algoritmus

Vstup: $A \in \mathbb{R}^{m\times n}, B \in \mathbb{R}^{m\times d}, tol > 0$
Inicializace:
$flag \leftarrow 0$
$k_1 \leftarrow \min\{m,n+d\}$ {index začátku singulárního podprostoru}
$k_2 \leftarrow d$ {index určující velikost singulárního podprostoru}
Konstrukce singulárního podprostoru:
$[B,A] = USV^T$ {singulární rozklad matice $[B,A]$}
$k_2 \leftarrow k_2 + l$ {přičtení levé násobnosti singulárního čísla σ_{n+1} určené kritériem (1.7)}
$V_{min} \leftarrow [v_{k_1-k_2+1}, \ldots, v_{k_1}]$ {volba singulárního podprostoru}
$V_{min} \leftarrow HV_{min}$ {aplikace Householderových reflexí}
Negenerický případ:
while $rank(V_{min}^{12}) < d$ **do**
$flag \leftarrow flag + 1$
$k_1 \leftarrow k_1 - k_2$
$k_2 \leftarrow d + l$ {přičtení levé násobnosti singulárního čísla $\sigma_{k_1-k_2+1}$ určené kritériem (1.7)}
$V_{min} \leftarrow [v_{k_1-k_2+1}, \ldots, v_{k_1}]$ {volba nového singulárního podprostoru}
end while
Výpočet řešení:
přímou eliminací vyřešíme $XV_{min}^{12} = -V_{min}^{22}$
Výstup: $X, flag$

4.1.3 Generování testovacích úloh podle klasifikace TLS řešitelnosti

Nyní popíšeme algoritmus sloužící ke generování testovacích úloh spadajících do konkrétní skupiny F_1,F_2,F_3 nebo S, popsané v Kapitole 1, Sekci (1.2). Vstupem programu jsou rozměry úlohy n, m a d. Dále zadáváme číslo určující do jaké skupiny bude úloha spadat a vektor obsahující singulární čísla matice $[B,A]$. Principem je vygenerování matic U, S a V, přičemž V splňuje podmínky pro požadovanou skupinu. Matice U je náhodně vygenerována pomocí funkcí *randn* a *qr* následovně

$$U = randn(m,m)$$
$$[U, \sim] = qr(U).$$

Příkaz $randn(m,m)$ vygeneruje náhodnou $m \times m$ matici s normálním rozdělením prvků a funkce *qr* spočítá QR rozklad matice U. Za U poté dosadíme faktor Q. Matici S získáme jednoduše doplněním správného počtu nulových řádků nebo sloupců k diagonální matici obsahující zvolená singulární čísla na diagonále. Matici V nejprve vygenerujeme stejným postupem jako U, tedy

$$V = randn(n+d,n+d)$$
$$[V, \sim] = qr(V).$$

K dosažení příslušných hodností bloků matice V použijeme Householderovy reflexe, které vynulují potřebný počet sloupců v jednotlivých blocích. Dále uvedeme konstrukci pro skupinu F_1 na konkrétním příkladu.

Algorithm 2 Iterační konstrukce core problému

Vstup: $A \in \mathbb{R}^{m\times n}, B \in \mathbb{R}^{m\times d}, tol > 0.$
Odstranění lineárně závislých sloupců B**:**
$B = [U_B S_{B1}, 0]V_B^T$ {singulární rozklad matice B}
$B1 \leftarrow U_B S_{B1}$
Inicializace:
$B1 = PR$ {QR rozklad matice $B1$}
$B1 \leftarrow R$
$Q \leftarrow 0 \in \mathbb{R}^{n\times \hat{d}}$
$L \leftarrow [\,]$
$\hat{Q} \leftarrow [\,]$
$band \leftarrow \hat{d}$ {proměnná pro počítání deflací}
Hlavní iterace:
while $band > 0$ **do**
$Q \leftarrow A^T P - QR^T$
$[Q,D] \leftarrow$ QR rozklad matice Q
kontrola horní deflace a zredukování matic Q a D
$l \leftarrow$ velikost horní deflace
update matice L o blok D^T
$band \leftarrow band - l$
$\hat{Q} \leftarrow [\hat{Q}, Q]$
$P \leftarrow AQ - PD^T$
$[P,R] \leftarrow$ QR rozklad matice P
kontrola dolní deflace a zredukování matic P a R
$l \leftarrow$ velikost horní deflace
update matice L o blok R
$band \leftarrow band - l$
end while
Doplnění pravé strany o správný počet nulových řádků:
$B1 \leftarrow \begin{bmatrix} B1 \\ 0 \end{bmatrix}$
Výstup: L,$B1$,$\hat{Q}$,V_B

Příklad. Mějme vygenerovanou náhodnou ortogonální matici $V \in \mathbb{R}^{6\times 6}$ a nechť $d = 2$, $r = 1$, $l = 2$ a $n = 4$. Chceme docílit splnění podmínek skupiny F_1. Tyto podmínky jsou

$$rank(V_{12}) = r = 1 \quad \text{a} \quad rank(V_{13}) = d - r = 1.$$

Snížíme proto hodnost matice V_{12} na 1 tím, že vynulujeme první dva její sloupce. Zkonstruujeme Householderovu reflexy H splňující

$$HV = \begin{pmatrix} \mathrm{x} & \mathrm{x} & 0 & 0 & \mathrm{x} & \mathrm{x} \\ \mathrm{x} & \mathrm{x} & 0 & 0 & \mathrm{x} & \mathrm{x} \\ \mathrm{x} & \mathrm{x} & 0 & \mathrm{x} & \mathrm{x} & \mathrm{x} \\ \mathrm{x} & \mathrm{x} & \mathrm{x} & \mathrm{x} & \mathrm{x} & \mathrm{x} \\ \mathrm{x} & \mathrm{x} & \mathrm{x} & \mathrm{x} & \mathrm{x} & \mathrm{x} \\ \mathrm{x} & \mathrm{x} & \mathrm{x} & \mathrm{x} & \mathrm{x} & \mathrm{x} \end{pmatrix},$$

kde x značí obecný nenulový prvek. Matice V_{12} je nyní tvaru

$$V_{12} = \begin{pmatrix} 0 & 0 & \mathrm{x} \\ 0 & 0 & \mathrm{x} \end{pmatrix},$$

a má požadovanou hodnost 1. U matice V_{13} není nutné dělat žádné úpravy, neboť chceme, aby měla plnou sloupcovou hodnost a v náhodně vygenerované ortogonální matici je tato podmínka typicky splněna.

Podmínky skupiny F_2 jsou splněny automaticky, neboť příslušné bloky mají plnou hodnost. V takovém případě není třeba žádných Householderových reflexí. Pro skupiny F_3 a S je postup analogický. V případě F_3 docílíme splnění podmínky $rank(V_{13}) < d - r$ tím, že vynulujeme první sloupec a snížíme tím hodnost na $d - r - 1$. Podmínka $rank(V_{12}) > r$ je splněna automaticky, neboť v náhodné ortogonální matici bude mít tento blok plnou sloupcovou hodnost $l+r$. Ve skupině S požadujeme, aby $rank([V_{12},V_{13}]) < d$. Toho docílíme vynulováním prvních $l+1$ sloupců tohoto bloku. Pro názornost uvedeme postup na příkladu.

Příklad. Mějme vygenerovanou náhodnou ortogonální matici $V \in \mathbb{R}^{6\times 6}$ a nechť $d = 2$, $r = 1$, $l = 2$ a $n = 4$. Chceme docílit splnění podmínky skupiny S. Tato podmínka je

$$rank([V_{12},V_{13}]) < d = 2.$$

Snížíme proto hodnost matice $[V_{12},V_{13}]$ na $d-1$ tím, že vynulujeme prvních $l+1$ sloupců. Zkonstruujeme Householderovu reflexy H splňující

$$HV = \begin{pmatrix} \mathrm{x} & \mathrm{x} & 0 & 0 & 0 & \mathrm{x} \\ \mathrm{x} & \mathrm{x} & 0 & 0 & 0 & \mathrm{x} \\ \mathrm{x} & \mathrm{x} & 0 & 0 & \mathrm{x} & \mathrm{x} \\ \mathrm{x} & \mathrm{x} & 0 & \mathrm{x} & \mathrm{x} & \mathrm{x} \\ \mathrm{x} & \mathrm{x} & \mathrm{x} & \mathrm{x} & \mathrm{x} & \mathrm{x} \\ \mathrm{x} & \mathrm{x} & \mathrm{x} & \mathrm{x} & \mathrm{x} & \mathrm{x} \end{pmatrix},$$

kde x značí obecný nenulový prvek. Matice V_{12} je nyní tvaru

$$[V_{12},V_{13}] = \begin{pmatrix} 0 & 0 & 0 & \mathrm{x} \\ 0 & 0 & 0 & \mathrm{x} \end{pmatrix},$$

a má požadovanou hodnost $d - 1$.

4.1.4 Generování testovacích úloh dle velikosti core problému

Zde ukážeme, jakým způsobem generujeme úlohy, u kterých pro testovací účely potřebujeme vědět, jak přesně vypadá core problém a jaké má rozměry. Postup je založený na explicitní konstrukci matice $L_{k,k-1}$. Prvních j bloků D_i a R_i jsou matice 2×2. V bloku D_{j+1} nastane horní deflace velikosti 1 a poté následuje l bloků R_i a D_i, které jsou tvořeny reálnými čísly. V bloku R_{j+l+1} nastane poslední deflace. Matice A_{22} je zvolena jako náhodná matice 2×2. Pravá strana B je náhodná matice se dvěma sloupci. Matice $\hat{Q}$ je zvolená jako náhodná ortogonální matice a matice $\hat{P}$ má první dva sloupce stejné jako Q faktor QR

rozkladu matice B a zbytek je doplněn náhodně. Celá matice je zortogonalizována. Náhodné matice jsou generovány pomocí funkce *randn* a pokud chceme, aby byly ortogonální, spočítáme QR rozklad a použijeme Q faktor. Bloky D_i a R_i jsou zvoleny pro všechna $i \leq j$ jako

$$D_i^T = \begin{bmatrix} 6 & 0 \\ 2 & 3 \end{bmatrix}, \quad R_i = \begin{bmatrix} 1 & 1 \\ 0 & 4 \end{bmatrix}.$$

Dále blok $D_{j+1}^T = [3,2]^T$ (po deflaci). Zbylé bloky jsou, pro $i > j+1$, $D_i = 2$ a $R_i = 1$.

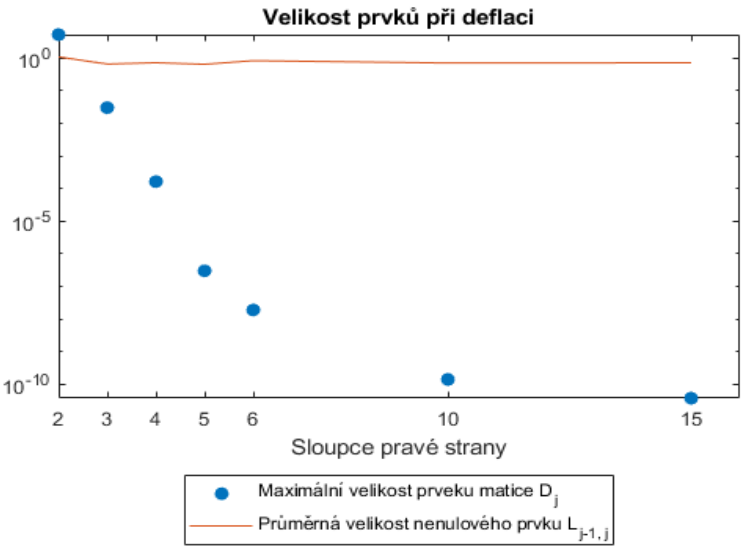
Vstupními parametry generátoru testovacích úloh jsou čísla j a l. Výstupem jsou matice A, B a také matice L_{test}, což je sestavená matice

$$L_{test} = \begin{bmatrix} L_{k,k-1} & 0 \\ 0 & A_{22} \end{bmatrix}.$$

4.2 Chování algoritmu pro iterační výpočet core problému

První experimenty, které provedeme, budou zaměřené na chovaní iteračního algoritmu počítajícího core problém. Podíváme se, jak velikost úlohy, počet pravých stran a samotná velikost core problému ovlivňují přesnost výpočtu. Budeme sledovat velikost prvků v maticích D_j a R_j v iteracích, kdy v přesné aritmetice nastávají deflace. Dále budeme sledovat ztrátu ortogonality matic $\hat{P}_j$ a $\hat{Q}_j$ a také chybu projekce $\hat{P}_{j+1}^T A \hat{Q}_j - L_{j+1,j}$.

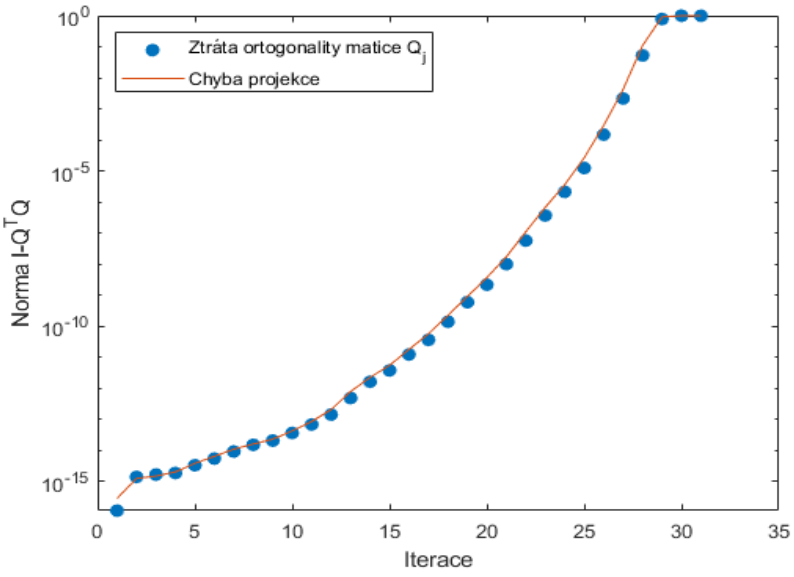
4.2.1 Problémy s proměnnou velikostí pravé strany



Obrázek 4.1: Velikost prvků matice D_j v poslední iteraci pro úlohy s fixní maticí A a různým počtem sloupců pravé strany B.

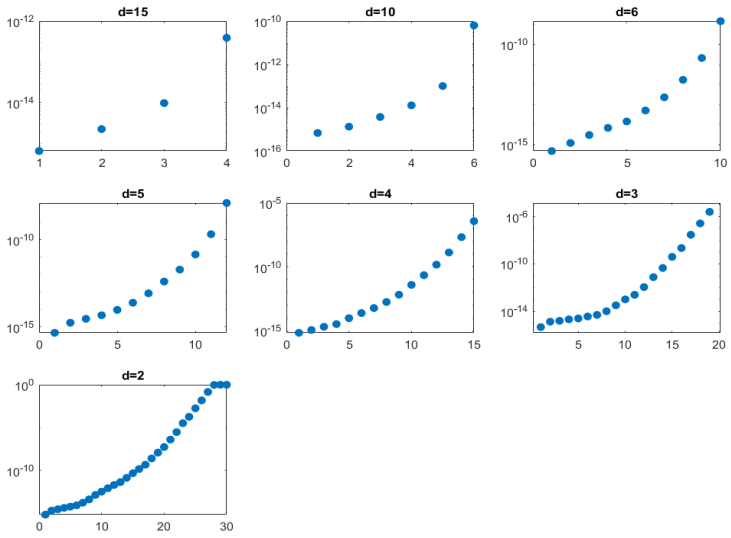
V tomto testu vygenerujeme matice A a B jako náhodné matice pomocí Matlabovské funkce *randn*. Matice A bude mít fixní rozměry 150×60. Taková matice je dobře podmíněná, v našem případě číslo podmíněnosti vyšlo 4.4846. Pro obdélníkové matice s plnou hodností číslem podmíněnosti myslíme podíl největšího a nejmenšího singulárního čísla. Budeme měnit počet sloupců pravé strany, tedy počet pozorování. Použité hodnoty jsou $d \in \{15{,}10{,}6{,}5{,}4{,}3{,}2\}$. V případě, že úloha $AX \approx B$ je sestavena z náhodných matic, má matice A vždy navzájem různá nenulová singulární čísla. Navíc sloupce matice B nejsou kolmé na žádný z příslušných levých singulárních vektorů matice A. Matice core problému má tedy stejná singulární čísla jako matice A a shoduje se v počtu sloupců. Zmenšující se počet sloupců pravé strany tedy způsobí, že bude třeba více iterací pro výpočet core problému s maticí, která má vždy 60 sloupců. My se zaměříme na pozorování velikosti prvků v matici D_j z poslední iterace, která by měla být teoreticky nulová. Dále také sledujeme ztrátu ortogonality mezi sloupci matice $\hat{Q}_j$.

Na grafu (4.1) znázorňují modré tečky maximální velikost hodnot v dolní trojúhelníkové matici D_j z poslední iterace pro úlohy s fixní maticí A a různým počtem sloupců pravé strany B. V tuto chvíli by měla být matice D_j vždy nulová. Pro porovnání červená čára značí průměrnou velikost nenulové hodnoty v matici $L_{j,j-1}$, která by měla tvořit core problém. S menším počtem sloupců pravé strany rostou hodnoty v matici D_j. Při dvou pravých stranách vidíme, že velikost prvků matice D_j je větší než průměrná velikost prvku v matici $L_{j,j-1}$ a tedy tuto deflaci není možné rozeznat ani volbou většího parametru *tol*. Při tomto počtu pravých stran bylo k výpočtu nutno provést 30 iterací. Vidíme, že průměrná velikost prvků matice $L_{j-1,j}$ zůstává stejná.



Obrázek 4.2: Ztráta ortogonality mezi sloupci matice $\hat{Q}_j$ a chyba projekce v případě $d = 2$ pro dobře podmíněnou matici A.

Graf (4.2) ukazuje na příkladu nejmenšího počtu pravých stran, tedy 2, jak s počtem iterací dochází ke ztrátě ortogonality v matici $\hat{Q}_j$ a zvětšování chyby



Obrázek 4.3: Ztráta ortogonality mezi sloupci matice $\hat{Q}_j$ v závislosti na počtu iterací pro různé počty pravých stran d.

projekce znázorněné červenou čarou. Tyto veličiny jsou měřeny pomocí norem

$$||I - \hat{Q}_j^T \hat{Q}_j|| \quad \text{a} \quad ||\hat{P}_{j+1}^T A \hat{Q}_j - L_{j+1,j}||.$$

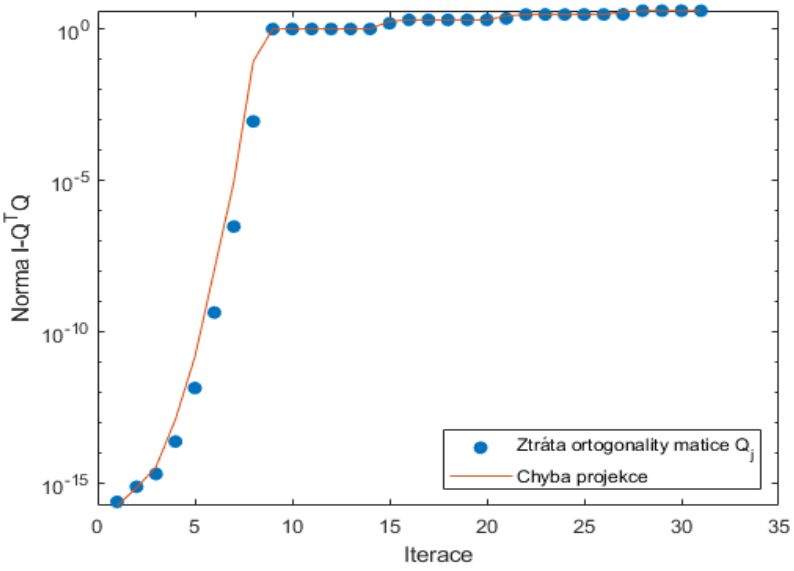
Obě tyto veličiny rostou stejně rychle. Okolo iterace 30 se ortogonalita ztratí úplně.

Následující graf (4.3) ukazuje ztrátu ortogonality ve výpočtech s různým počtem pravých stran. Všimněme si, že v případech, kdy d je velké, je ortogonalita zachována poměrně dobře po celou dobu výpočtu. Počet iterací potřebných k výpočtu spolu s velikostí bloků matic $\hat{P}_j$ a $\hat{Q}_j$, které jsou v našem případě určené právě počtem pravých stran, mají vliv na přesnost výpočtu.

Předchozí výsledky jsem získali pro dobře podmíněnou matici A s plnou hodností. Pro srovnání se proto podíváme na ztrátu ortogonality a chybu projekce pro špatně podmíněnou matici A. Zvolili jsme matici A, jejíž největší singulární číslo je 10 a nejmenší 10^{-6}. Číslo podmíněnosti tedy vyjde 10^7. Podíváme se, jaký dopad to bude mít na rychlost ztráty ortogonality. Na grafu (4.4) vidíme, že ve srovnání s předchozím případem zachyceném na grafu (4.3) se ortogonalita i chyba projekce ztrácí podstatně rychleji, konkrétně do 10 iterací.

Výsledkem těchto experimentů jsou následující závěry.

- Ukázali jsme, že existují i jiné faktory než je velikost core problému, které určují přesnost výpočtu. Záleží také na počtu potřebných iterací. Počet iterací ovlivňuje velikost pravé strany a také vztah matic A a B určující, kdy nastanou deflace. Poznamenejme, že v obecném případě může počet pravých stran ovlivnit počet sloupců matice core problému. Větší počet sloupců pravé strany může totiž způsobit, že nějaké vícenásobné singulární číslo matice A bude v core problému obsaženo vícekrát. V našem případě tato situace nenastala, neboť matice A měla pouze jednonásobná singulární čísla.

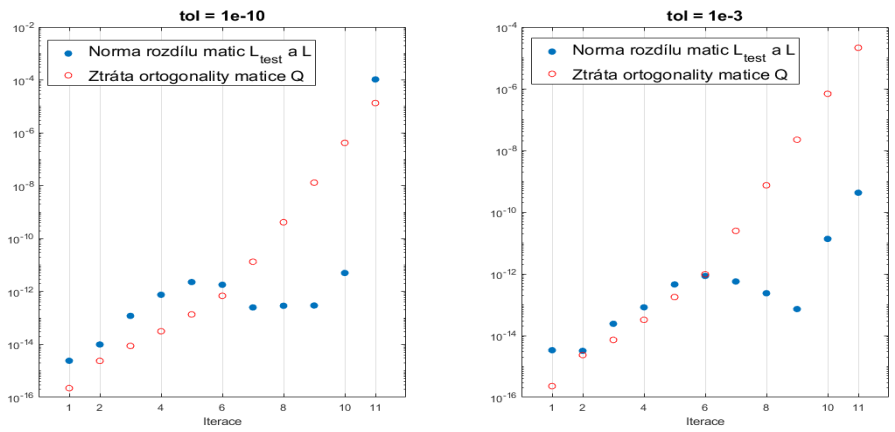


Obrázek 4.4: Ztráta ortogonality a chyba projekce v závislosti na číslu iterace v případě $d = 2$ pro špatně podmíněnou matici A.

- QR rozklady počítané v každé iteraci umožnujě udržovat ortogonalitu mezi sloupci matic $\hat{Q}$ a $\hat{P}$ spočtenými ve stejném bloku. Při větším počtu pravých stran se nám lépe zachová ortogonalita, neboť v každém kroku počítáme větší bloky matic $\hat{Q}$ a $\hat{P}$.
- Ztráta ortogonality a chyba projekce narůstají řádově stejným tempem. Ortogonalita se ztrácí téměř lineárně v logaritmickém měřítku a v rozmezí mezi 25. a 30. iterací je ve výše uvedeném experimentu ztracena kompletně i pro dobře podmíněnou matici. Pokud je matice špatně podmíněná, ztráta ortogonality je podstatně rychlejší. V našem testu přibližně třikrát.
- Deflace, které by měli nastat ve vyšších iteracích není možné naším kritériem odhalit, neboť prvky, které by měli být nulové, mají velikost na úrovní velikosti ostatních prvků matice $L_{j-1,j}$. Děje se tak vlivem diskutovaných chyb způsobených ztrátou ortogonality.

4.2.2 Problémy s proměnnou velikostí core problému

Nyní budeme zkoumat chování iteračního algoritmu pro výpočet core problému při aplikaci na úlohy s různou velikostí core problému. Cílem je sledovat, jak velikost core problému a číslo iterace, ve které nastává deflace, ovlivní spolehlivost výpočtu. Testovací úlohy jsou generovány postupem uvedeným v Podsekci (4.1.4). V prvním případě zvolíme $j = l = 5$. Rozměry matice core problému tedy budou 17×16. První deflace by měla nastat v iteraci číslo 6 a druhá a poslední v iteraci 11. Počet sloupců pravé strany je 2. Na grafech (4.5) vidíme pro dvě zvolené hodnoty parametru *tol* průběh výpočtu. Modré tečky značí normu rozdílu mezi bloky matice L spočtenými v i-té iteraci a odpovídajícími bloky v přesné



Obrázek 4.5: Přesnost spočtených bloků matice L a ztráta ortogonality v matici $\hat{Q}$ pro 11 iterací a dvě různé volby tol.

matici L_{test}. V prvních j iteracích se jedná vždy o maticovou normu rozdílu

$$\begin{bmatrix} D_i^T \\ R_{i+1} \end{bmatrix} - \begin{bmatrix} 6 & 0 \\ 2 & 3 \\ 1 & 1 \\ 0 & 4 \end{bmatrix}$$

a ve zbývajících iteracích o normy rozdílů

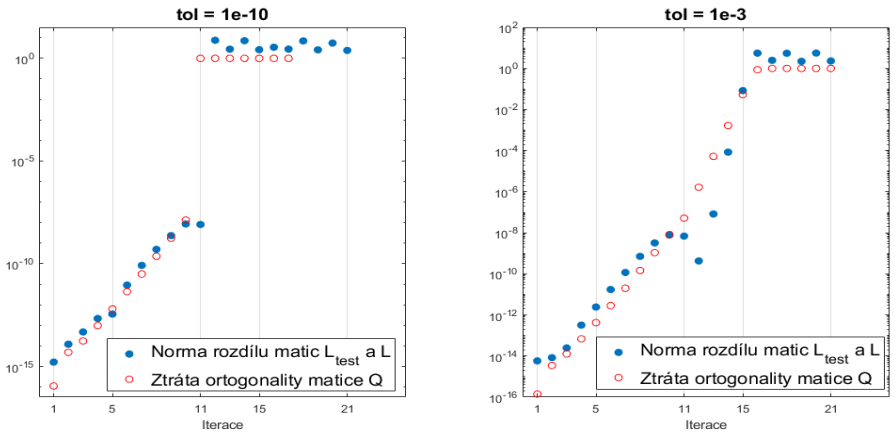
$$\begin{bmatrix} D_i^T \\ R_{i+1} \end{bmatrix} - \begin{bmatrix} 2 \\ 1 \end{bmatrix}$$

S jediným rozdílem v iteracích $j+1$ a $j+l+1$, kde nastanou deflace. V těchto iteracích se jedná po řadě o normy rozdílů

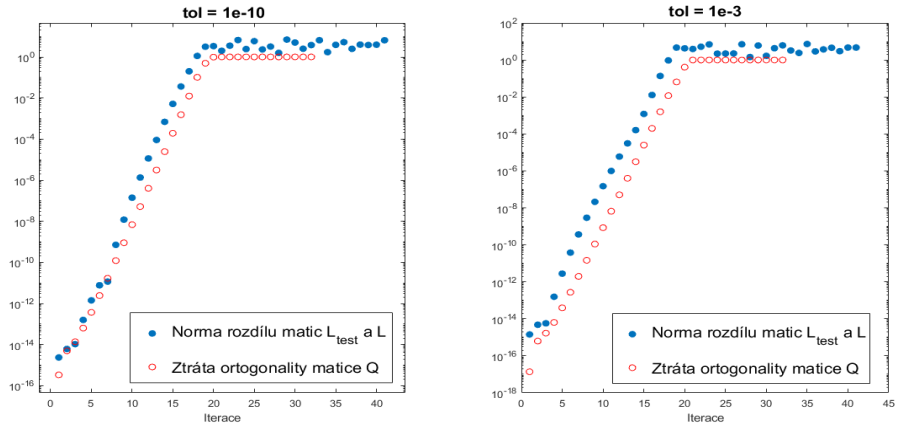
$$\begin{bmatrix} D_{j+1}^T \\ R_{j+2} \end{bmatrix} - \begin{bmatrix} 3 \\ 2 \\ 1 \end{bmatrix}, \quad \begin{bmatrix} D_{j+l+1}^T \\ R_{j+l+2} \end{bmatrix} - \begin{bmatrix} 2 \\ 0 \end{bmatrix}.$$

Pro obě tolerance se podařilo rozpoznat první deflaci v šesté iteraci. V případě, kdy $tol = 10^{-10}$, se nepodařilo zachytit druhou deflaci v iteraci 11, což způsobuje velké navýšení chyby v poslední iteraci. Pro vyšší toleranci se podařilo rozpoznat i druhou deflaci. Chyba v poslední iteraci proto zůstává podstatně menší. Nehledě na toleranci a případné odhalení nebo neodhalení deflace, ztráta ortogonality narůstá stále stejně.

Nyní provedeme stejný test pro úlohu s větším rozměrem core problému. Zvolíme $j = l = 10$. Rozměry matice core problému budou 32×31. První deflace by měla nastat v iteraci 11 a druhá v 21. Výsledky jsou zachyceny na grafu (4.6). V případě nižší tolerance nebyla zachycena první deflace a zbytek iterací je následně zatížen nepřesnostmi ve výpočtu. Deflace nenastává ani se zpožděním, rozměry počítaných matic se nesnižují. V druhém případě, kdy $tol = 10^{-3}$, je odhalena první deflace. Chyba nicméně narůstá velmi rychle i poté a druhá deflace odhalena není. Opět nedojde k deflaci ani opožděně a výpočet se nezastaví.



Obrázek 4.6: Přesnost spočtených bloků matice L a ztráta ortogonality v matici $\hat{Q}$ pro 21 iterací a dvě různé volby *tol*.



Obrázek 4.7: Přesnost spočtených bloků matice L a ztráta ortogonality v matici $\hat{Q}$ pro 41 iterací a dvě různé volby *tol*.

V posledním pokusu byly zvoleny parametry $j = l = 20$. Matice core problému má pak dimenze 62×61. Výsledky jsou zakresleny do grafu (4.7). Zde pozorujeme, pro obě dvě tolerance, že chyba výpočtu i ztráta ortogonality naroste před první deflací (iterace číslo 21) na úroveň 10^0. Odhalení deflace je tak nemožné nehledě na zvolenou toleranci *tol*.

Závěr z těchto testů shrneme v následujících bodech.

- Ortogonalita sloupců počítaných matic se opět ztrácí lineárním tempem v logaritmickém měřítku.
- Odhalení deflace překvapivě nemá pozitivní vliv na ztrátu ortogonality.
- Přesnost spočtených bloků matice L klesá rychlostí odpovídající rychlosti ztrátě ortogonality. Odhalení deflace však v tomto případě způsobí dočasné zpřesnění pro následující iterace. Následně chyba začne rychle opět narůstat až na úroveň ztráty ortogonality.

- Deflace v nižších iteracích (v našem případě < 15) nastávají vždy ve správné iteraci a jejich odhalení závisí na velikosti parametru *tol*. Jak bylo popsáno v předchozí sekci, deflace, která by měla nastat v příliš vysoké iteraci (v našem případě $>= 20$), nelze naším kritériem odhalit. Žádné zpoždění deflací nebylo pozorováno.

4.3 Srovnání TLS řešení originální úlohy a core problému

V této sekci se zaměříme na srovnání dvou aproximačních řešeních. První získáme aplikací klasického TLS algoritmu na úlohu (1.1) a budeme ho značit X_{TLS}. Druhé obdržíme použitím klasického TLS algoritmu na core problém v úloze (1.1) s následnou zpětnou transformací. Toto řešení označíme jako X_{core}. Core problém je počítán Algoritmem 2 a tolerancí $tol = 10^{-3}$. Testovací úlohy jsou generovány podle skupiny TLS řešitelnosti popsané v Kapitole 1, Sekci 1.2. Způsob generování takových úloh byl popsán v Sekci 4.1.3. Budeme sledovat normu residua u obou získaných řešení a také relativní normu rozdílu těchto dvou řešení. Zavedeme pro tyto normy následující značení

$$\begin{aligned} r_{TLS} &= ||AX_{TLS} - B||, \\ r_{core} &= ||AX_{core} - B||, \\ f &= \frac{||X_{TLS} - X_{core}||}{||X_{TLS}||}. \end{aligned}$$

Rozměry úloh zvolíme jako $m = 15$, $n = 7$ a $d = 3$. Toto jsou, podle výsledků minulé sekce, dostatečně malé rozměry, aby byl výpočet core problému stále přesný. V první sadě úloh zvolíme následující singulární čísla matice $[B,A]$:

$$S = diag(100,60,40,20,10,5,2,2,2,1).$$

V tomto případě je matice $[B,A]$ dobře podmíněná. Z každé skupiny S, F_1, F_2 a F_3 jsem vygenerovali 1000 úloh, které mají rozměry a singulární čísla uvedená výše. V tabulce níže jsou pro každou skupinu uvedeny výsledky získané zprůměrováním hodnot ze všech 1000 úloh v dané skupině.

Skupina:	S	F_1	F_2	F_3
$\bar{r}_{TLS}$	124.4600	1.3714	0.1679	3.0404
$\bar{r}_{core}$	1.2233	1.3714	0.1679	0.7913
$\bar{f}$	1.0838	3.3809×10^{-8}	1.5257×10^{-11}	0.0010

Získaná řešení jsou velmi blízká ve skupinách F_1 a F_2. To jsou skupiny, kde existuje řešení ve smyslu úplných nejmenších čtverců. Naopak ve skupinách S a F_3, kde řešení neexistuje, pozorujeme větší rozdíl v získaných maticích. V obou případech dostaneme nejhorší výsledky, co se residua týče, pro skupinu S. Avšak v druhé variantě výpočtu (přes core redukci) pozorujeme podstatně nižší normu

residua pro spočtenou aproximaci negenerického řešení. Stejné zlepšení nastává i pro skupinu F_3.

V druhé sadě testů jsme ponechali stejné rozměry úloh. Jediné, co se mění, jsou singulární čísla matice $[B,A]$, která nyní volíme

$$S = diag(10,6,4,2,1,10^{-3},10^{-5},10^{-5},10^{-5},10^{-6}).$$

Matice $[B,A]$ je v tomto případě špatně podmíněná. Dále následuje tabulka s analogickými výsledky jako v prvním případě.

Skupina:	S	F_1	F_2	F_3
$\bar{r}_{TLS}$	9.1811	5.1799×10^{-5}	8.3076×10^{-6}	1.1967×10^{-4}
$\bar{r}_{core}$	0.0026	4.0168×10^{-4}	2.7742×10^{-4}	4.3385×10^{-4}
$\bar{f}$	1.0710	0.7887	0.7071	0.7850

Zde získáme rozdílné aproximace řešení ve všech skupinách. Ovšem ve skupinách F_1 a F_2, kde řešení ve smyslu úplných nejmenších čtverců existuje, vidíme, že oba postupy vedou na řešení s malou normou residua. Druhý přístup (přes core redukci) má normu residua nepatrně vyšší. Ve skupině F_3 žádný podstatný rozdíl v normách residua není. Stejně jako v předchozím případě vidíme nižší normu residua pro druhou metodu při použití na aproximační úlohy ve skupině S.

Výsledky shrneme v následujících bodech.

- Pro dobře podmíněnou matici jsou si získaná řešení blízko v případech, kdy existuje řešení ve smyslu úplných nejmenších čtverců. Pokud řešení neexistuje, potom mohou být výsledky významně odlišné. V tuto chvíli totiž počítáme nějaké negenerické řešení a proto není překvapivé, že se výsledky budou lišit.

- Pro špatně podmíněnou matici se spočtené aproximace zásadně liší ve všech případech. Ve skupinách F_1 a F_2, kde řešení ve smyslu úplných nejmenších čtverců existuje, mají obě aproximace srovnatelnou normu residua.

- Pro úlohy ve skupině S získáme výpočtem přes core redukci aproximaci s výrazně menším residuem. To může být způsoben tím, že v našem případě patřil core problém do vždy skupiny F_1 nebo F_2 a tedy nebylo nutné počítat jeho negenerické řešení. Core problém může obecně, jak bylo dokázáno v [12], spadat do libovolné skupiny TLS řešitelnosti. V našich testech jsme ovšem na jiné případy nenarazily.

Závěr

V této práci jsme na základě použité literatury popsali problém úplných nejmenších čtverců pro řešení aproximační úlohy $AX \approx B$. Shrnuli jsme klasifikaci řešitelnosti ve smyslu úplných nejmenších čtverců a popsali jsme klasický TLS algoritmus sloužící k výpočtu aproximace řešení. Dále jsme podrobně vysvětlili core redukci, která má za cíl, pomocí ortogonálních transformací, zredukovat dimenze úlohy tím, že odstraní z dat A a B přebytečnou informaci. Výsledná zredukovaná úloha se nazývá core problém. Pro konstrukci core problému jsme uvedli dva postupy. Přímou metodu založenou na singulárním rozkladu a iterační konstrukci využívající zobecněnou Golub-Kahanovu bidiagonalizaci. Také jsme, na základě těchto konstrukcí, vyvodili některé vlastnosti a charakterizace core problému.

Pozornost byla dále věnována numerickým experimentům v prostředí Matlab. V těchto experimentech jsme se zaměřili na klasický TLS algoritmus a implementaci iterační konstrukce core problému. Nejprve jsme na příkladu náhodných matic A a B, které jsou typicky dobře podmíněné, ukázali, že ztráta ortogonality při výpočtu core problému závisí z velké míry na počtu potřebných iterací k výpočtu a velikosti pravé strany. Čím větší bloky matic $\hat{Q}$ a $\hat{P}$ v každé iteraci počítáme, tím lépe se zachová ortogonalita. Velikost počítaných bloků ovšem nezávisí pouze na počtu pravých stran, ale také na vztahu matic A a B určujícím, kdy a nastanou deflace a jak dobře bude možné je numericky detekovat. Pro srovnání jsme na příkladu špatně podmíněné matice A ukázali, že ortogonalita se ztrácí podstatně rychleji, než v případě s dobře podmíněnou maticí A. V druhém pokusu jsme si nejprve sami sestavili matici L tvořící core problém. Potom jsme zpětně zkonstruovali úlohu. Zkoumali jsme, jak přesný výsledek získáme naším iteračním algoritmem. Ukázali jsme, že výsledky jsou relativně spolehlivé do 15. iterace. V pozdějších iterací dochází k úplné ztrátě ortogonality matice $\hat{Q}$ a nelze již správně detekovat deflace a to ani volbou vyššího parametru tol. Poměrně překvapivě jsme nepozorovali výskyt zpožděných deflací. Všechny výpočty byli provedeny bez reortogonalizace. V posledním experimentu jsme srovnali aproximace řešení získaných dvěma postupy. První řešení získáme pomocí aplikace klasického TLS algoritmu na úlohu $AX \approx B$. Druhé řešení obdržíme použitím klasického TLS algoritmu na core problém v této úloze s následnou zpětnou transformací. Úlohy pro tyto testy byly generovány tak, abychom pokryli všechny skupiny TLS řešitelnosti. Pro dobře podmíněné matice si řešení spočtená oběma postupy byla blízká ve skupinách, kde řešení ve smyslu úplných nejmenších čtverců existuje. V ostatních případech se řešení lišila. Pro špatně podmíněné matice byla řešení rozdílná ve všech skupinách.

Poslední provedený experiment by bylo možné dále rozšířit. Z [12] víme, že core problém může patřit do libovolné skupiny TLS řešitelnosti. My jsme ukázali, že pro úlohy ze skupin S a F_3, kde řešení ve smyslu úplných nejmenších čtverců neexistuje, se aproximace řešení spočítané v prostředí Matlab významně liší. Ve skupině S jsme pozorovali snížení normy residua u metody přes core redukci. V našich testech ovšem core problém vždy spadl do skupiny F_1 nebo F_2, kde

řešení ve smyslu úplných nejmenších čtverců existuje. Pokud bychom dokázali vygenerovat úlohy ze skupiny S a F_3 a zároveň kontrolovat do jaké skupiny bude patřit příslušný core problém, bylo by možné experimentálně ukázat, jestli tyto výsledky budou platit i pro core problémy, které budou ze skupin S a F_3. Získané výsledky je třeba dále hlouběji analyzovat, což však již přesahovalo rozsah zadání diplomové práce.

Literatura

[1] A. Björck, *A Band-Lanczos Generalization of Bidiagonal Decomposition*, Presentation, Conference in Honor of G. Dahlquist, Stockholm, Sweden, 2006

[2] C. Eckhart, G. Young, *The approximation of one matrix by another of lower rank*, Psychometrika, 1 (1936), str. 211-218

[3] R. D. Fierro, G. H. Golub, P. C. Hansen a O'Leary, *Regularization by truncated total least squares*, SIAM J. Sci. Comput., 18 (1997), str. 1223-1241

[4] G. H. Golub a W. Kahan, *Calculating the singular values and pseudo-inverse of a matrix*, SIAM J. Numer. Anal., 2 (1965), str. 205-224

[5] G. H. Golub, C. F. Van Loan, *An analysis of the total least squares problem*, SIAM J. Numer. Anal., 17 (1980), str. 883-893

[6] G. H. Golub, C. F. Van Loan, *Matrix Computations*, The Johns Hopkins University Press, Baltimore and London, 1996

[7] M. H. Gutknecht, *Block Krylov space methods for linear systems with multiple right-hand sides: An Introduction*, Seminar for Applied Mathematics, Zurich (2006)

[8] M. H. Gutknecht, T. Schmelzer, *The block grade of a block Krylov space*, Linear algebra and its applications, 430 (2009), str. 174-185

[9] I. Hnětynková, M. Plešinger, D. M. Sima, Z. Strakoš a S. Van Huffel, *The total least squares problem in $AX \approx B$: A new classification with the relationship to the classical works*, SIAM J. Matrix Ana., 32 (2011), str. 748-770

[10] I. Hnětynková, M. Plešinger a Z. Strakoš, *Band Generalization of the Golub-Kahan Bidiagonalization, Generalized Jacobi Matrices, and the Core problem*, SIAM J. Matrix. Anal., 36 (2015), str. 417-434

[11] I. Hnětynková, M. Plešinger a Z. Strakoš, *The core problem within a linear approximation problem $AX \approx B$ with multiple right hand sides*, SIAM J. Matrix Anal., 34 (2013), str. 917-931

[12] I. Hnětynková, M. Plešinger a D. M. Sima, *Solvability of the core problem with multiple right hand sides in the TLS sense*, SIAM J. Matrix Anal., 37 (2016), str. 861-876

[13] C. Lanczos, *An iteration method for the solution of the eigenvalue problem of linear differential and itegral operators*, J. Res. Nat. Bur. Standarts, 45 (1950), str. 255-282

[14] J. Liesen a Z.Strakoš, *Krylov subspace methods: Principles and analysis*, Oxford University Press, 2012

[15] C. C. Paige a Z. Strakoš, *Core problems in linear algebraic systems*, SIAM J. Matrix Anal., 27 (2006), str. 861-875

[16] J. D. Tebbens, I. Hnětynková, M. Plešinger, Z. Strakoš a P.Tichý *Analýza metod pro maticové výpočty: Základní metody*, Matfyzpress, 2012

[17] S. Van Huffel, J. Vandewalle, *The Total Least Squares Problem: Computation Aspects and Analysis*, Frontiers in applied mathematics, SIAM, Philadelphia 1991

[18] S. Van Huffel, *Documented Fortran 77 programs of the extended classical total least squares algorithm, the partial singular value decomposition algorithm and the partial total least squares algorithm*, Internal. Report ESAT-KUL 88/1, ESAT Lab., Debt. of Electrical Engrg., Katholieke Universiteit Leuven, Leuven, Belgium, 1988

[19] S. Van Huffel, *The extended classical total least squares algorithm*, J. Comput. Appl. Math., 25 (1989), str. 111-119

[20] J. H. Wilkinson, *The Algebraic Eigenvalue Problem*, Clarendon Press, Oxford (1965)