Bakalářská práce

České
vysoké
učení
technické
v Praze

F1

Fakulta stavební
Katedra mechaniky

# Duální metody dělení oblasti pro kontaktní úlohy

Jakub Mareš

Školitel: Ing. Martin Doškář, Ph.D
Květen 2022

# ZADÁNÍ BAKALÁŘSKÉ PRÁCE

## I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Mareš** Jméno: **Jakub** Osobní číslo: **486098**

Fakulta/ústav: **Fakulta stavební**

Zadávající katedra/ústav: **Katedra mechaniky**

Studijní program: **Stavební inženýrství**

Studijní obor: **Konstrukce pozemních staveb**

## II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce:

**Duální metody dělení oblasti pro kontaktní úlohy**

Název bakalářské práce anglicky:

**Dual domain decomposition methods for contact problems**

Pokyny pro vypracování:

Seznam doporučené literatury:

Jméno a pracoviště vedoucí(ho) bakalářské práce:

**Ing. Martin Doškář, Ph.D. katedra mechaniky FSv**

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) bakalářské práce:

Datum zadání bakalářské práce: **18.02.2022** Termín odevzdání bakalářské práce: **15.05.2022**

Platnost zadání bakalářské práce: _____

| Ing. Martin Doškář, Ph.D.<br>podpis vedoucí(ho) práce | podpis vedoucí(ho) ústavu/katedry | prof. Ing. Jiří Máca, CSc.<br>podpis děkana(ky) |
|---|---|---|

## III. PŘEVZETÍ ZADÁNÍ

Student bere na vědomí, že je povinen vypracovat bakalářskou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v bakalářské práci.

| Datum převzetí zadání | Podpis studenta |
|---|---|


© ČVUT v Praze, Design: ČVUT v Praze, VIC

# Poděkování

V první řadě bych chtěl poděkovat Ing. Martinu Doškářovi, Ph.D. za to, že mne spolu s Prof. Ing. Janem Zemanem, Ph.D. uvedli do oblasti numerických řešení úloh a podporovali ve mně nadšení pro řešení problémů metod z rodiny FETI a pro kontaktní úlohy. Mé vlezlé dotazy mu jistě často vytvořily na čele vrásky a já jsem mu vděčný za jeho věcné a mnohdy zcela, a to i po grafické stránce, vyčerpávající odpovědi.

Dále bych chtěl poděkovat doc. RNDr. Ivaně Pultarové, Ph.D. za její pomoc a věcné připomínky z oblasti matematiky a za její trpělivost a obětavou práci při vysvětlování nejasností z různých oblastí matematiky.

V neposlední řadě bych chtěl poděkovat své rodině a přátelům, kteří mi byli při práci oporou a bez jejichž přispění by tato práce nikdy nemohla vzniknout.

Má vděčnost patří i Grantové agentuře České republiky, za jejíhož finančního přispění v rámci projektu č. 19-26143X tvorba této práce probíhala. Tato skutečnost mi byla motivací a umožnila mi věnovat se přípravě této práce bez nutnosti různých zdržení.


# Prohlášení

Prohlašuji, že jsem bakalářskou práci na téma „Duální metody dělení oblasti pro kontaktní úlohy“ vypracoval samostatně a veškerou použitou literaturu a další prameny jsem řádně označil a uvedl v přiloženém seznamu.

Ve Stěžírkách, 15. května 2022

Podpis .....

# Abstrakt

Předmětem této práce je studium a implementace metod pro řešení úloh kvadratického programování s omezujícími podmínkami ve formě lineárních rovnic a nerovnic a s využitím postupů dělení oblasti.

Práce se zabývá duální metodou Finite Element Tearing and Interconnecting (FETI), metodou aktivní množiny a jejich kombinací v rámci metody metody FETI-C.

Metoda FETI-C je ilustrována na dvourozměrné úloze kontaktu sady prutů s rozdílnou tuhostí a zatížením.



**Klíčová slova:** doménová dekompozice, iterativní metody, metoda největšího spádu, metoda sdružených gradientů, kontaktní úlohy, metoda aktivní množiny, FETI, FETI-C



**Školitel:** Ing. Martin Doškář, Ph.D
Katedra mechaniky
Fakulta stavební



# Abstract

The purpose of this thesis is to study and implement methods for solving the problems of quadratic programming with boundary conditions in the form of linear equalities and inequalities with emphasis on the principles of domain decomposition.

In particular, this thesis focuses on the Finite Element Tearing and Interconnecting (FETI) method, the active set method, and also their combination within one method, FETI-C.

The FETI-C method is illustrated with a 2D problem dealing with the contact of a set of struts with varying stiffness and loading.



**Keywords:** domain decomposition, iterative methods, method of steepest descent, conjugate gradient method, contact problems, active set method, FETI, FETI-C


**Title translation:** Dual domain decomposition methods for contact problems

# Obsah

**1 Úvod** **1**
1.1 Cíle práce a motivace . . . . . 2
**2 Úvod do numerických metod** **3**
2.1 Vlastní čísla a vlastní vektory . . . 3
2.2 Soustavy lineárních rovnic a jejich kvadratická forma . . . . . 4
2.2.1 Vliv vlastních čísel na tvar kvadratické formy . . . . . 5
**3 Metoda největšího spádu** **7**
**4 Metoda sdružených gradientů** **11**
**5 Metoda Lagrangeových multiplikátorů** **19**
**6 Duální metody výpočtu** **23**
6.1 Duální metoda dělení oblasti . . . 23
6.2 FETI metoda . . . . . 26
6.2.1 Základní metoda FETI . . . . . 26
6.2.2 Řešení metody FETI pomocí metody PCPG . . . . . 32
**7 Kontaktní úlohy** **35**
7.1 Metoda aktivní množiny . . . . . 37
**8 Metoda FETI-C** **43**
**9 Implementace metody aktivní množiny a metody FETI-C** **51**
**10 Závěr** **55**
**Literatura** **57**

# Obrázky

2.1 Příklad tvarů kvadratických funkcí pro $\mathbf{K} \in \mathbb{R}^{2\times 2}$ . . . . . 6

3.1 Zobrazení rezidua na kvadratickou formu . . . . . 8
3.2 Příklad průběhu metody největšího spádu, $\mathbf{K} \in \mathbb{R}^{2\times 2}$ . . . . . 10
3.3 Pohledy na příklad průběhu metody největšího klesání . . . . . 10

4.1 Konjugované vektory, znázornění kolmosti $\boldsymbol{d}_i^T$ na $\mathbf{K}\boldsymbol{d}_j$ . . . . . 12
4.2 Konjugované vektory, kolmost $\boldsymbol{d}_i$ a $\boldsymbol{d}_j$ na konjugovaném prostoru . . . . . 13
4.3 Příklad průběhu metody sdružených gradientů, $\mathbf{K} \in \mathbb{R}^{2\times 2}$ . . 18
4.4 Pohled na průběh směrových vektorů při metodě CG . . . . . 18

5.1 Znázornění extrémů vázaných podmínkami . . . . . 22

6.1 Znázornění původní úlohy . . . . . 24
6.2 Úloha rozdělená na domény a připravená pro duální řešení . . . . . 25
6.3 Znázornění $\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda})$ a jeho rozklad na $f(\boldsymbol{x})$ a $\boldsymbol{\lambda}^T \boldsymbol{g}(\boldsymbol{x}^T)$ . . . . . 28
6.4 Rozklad na $f(\boldsymbol{x})$ a $\boldsymbol{\lambda}^T \boldsymbol{g}(\boldsymbol{x})^T$ . . . . 28

7.1 Vyobrazení kontaktní úlohy . . . . 36

9.1 Zadání testovací úlohy . . . . . 51
9.2 Řešení testovací úlohy . . . . . 52
9.3 Porovnání vývoje chyby a velikosti vyhledávacího kroku metoda FETI-C . . . . . 52
9.4 Porovnání vývoje množství aktivních podmínek při metodě aktivního setu a metodě FETI-C . . 53

10.1 Příklad úlohy s různým rozložením tuhostí materiálu . . . . . 55

# Kapitola 1

# Úvod

Tato bakalářská práce se bude v nejširším slova smyslu zabývat numerickými metodami. Numerické metody se zabývají způsoby jak určit výsledek zadaného problému s co nejmenší spotřebou času a energie. Zároveň umožňují snížení přesnosti řešení za účelem zvýšení rychlosti výpočtu, takže jejich výstupem není dokonale přesný výsledek, ale řešení velice blízké skutečné hodnotě.

V inženýrské praxi ve většině případů nejsme schopni nalézt vstupní hodnoty pro jakýkoliv problém s dostatečnou přesností a mírou jistoty, aby se nám vyplatilo přesně řešit velké soustavy rovnic.

Proto mají numerické výpočty v praxi své místo. Jejich význam s rozvojem technologií stále stoupá a my se s nimi setkáváme stále častěji.

Numerické řešiče se často využívají pro řešení rozmanitých problémů hledání extrémů funkce.

Na prvních několika stránkách se tedy budeme zabývat základy numerických metod. Ukážeme, jak převést soustavy lineárních rovnic na kvadratickou formu a jak nalézt její optimum. Předvedeme základní tvary kvadratických forem a základní gradientní metody.

Gradientní metody jsou iterativní numerické řešiče, které využívají geometrických vlastností úlohy a hledají optima předložené funkce pomocí jejího gradientu. Z těchto metod se budeme především věnovat metodě největšího spádu a metodě sdružených gradientů.

Tím vším si připravíme půdu pro metody z rodiny FETI. Tyto metody využívají duality funkcí a tedy se řadí mezi duální metody. Součástí metody FETI je dělení na oblasti a je pro ni možné využít řešení gradientními řešiči.

Následně tyto metody zobecníme pro řešení kontaktních úloh. Kontaktní úlohy představují problém, jehož součástí je dotek dvou prvků nebo rozvoj trhlin a podobně. V takových místech se přenášejí síly jen potom, nebo do té doby, než daná konstrukce dosáhne určitých deformací.

V rámci práce budou skalární veličiny značeny pomocí kurzívy, $a$, vektory budou psány kurzívou a tučně, např. $\boldsymbol{x}$, a matice budou zapisovány pomocí tučného řezu písma, $\mathbf{A}$. Intervaly jsou značeny kurzívou a lomenými, či kulatými závorkami $\langle a;b\rangle$,$(a;b)$. Hranaté závorky budou vyhrazeny pro značení souřadnic nebo rozepsaný maticový zápis a složené závorky budou značit množinu. Odkazy na obrázky a definice nebudou rozlišeny, odkazy na rovnice budou uvedeny v kulatých závorkách.

## 1.1 Cíle práce a motivace

Jak už bylo řečeno výše, cílem práce je ukázat různé způsoby řešení soustav lineárních rovnic pomocí numerických metod, představit základní gradientní metody a vysvětlit jejich geometrické vlastnosti. Výsledkem práce bude představení metody FETI a metody aktivní množiny a následně jejich propojení do metody FETI-C.

Současně s tvorbou textové části práce probíhá i implementace všech uvedených metod pro jednoduché příklady. Cílem je pochopit a vyzkoušet si jednotlivé metody a naučit se řešení kontaktních úloh.

Tyto znalosti a zkušenosti budou podkladem pro další studium a implementační snažení v oblasti delaminačních poškození materiálu.

# Kapitola 2

# Úvod do numerických metod

## 2.1 Vlastní čísla a vlastní vektory

Vlastní čísla a vektory jsou důležitým ukazatelem v teorii numerických metod. Určují, které metody je vhodné použít, a dávají nám informace o vlastnostech úlohy. Jejich výpočet je však zdlouhavý, a proto není vhodné je využívat přímo pro výpočet.

Vlastní čísla jsou tedy jakýmsi analytickým aparátem, který nebude využit v následujících metodách, ale jehož znalost je nutná pro pochopení principů uvedených metod.

**Definice 2.1.** ***O vlastních vektorech*** *[She94]*

*Nechť existuje čtvercová matice* $\mathbf{K} \in \mathbb{R}^{n \times n}$ *, kde* $n \in \mathbb{N}$*. Pak nenulový vektor* $\boldsymbol{v}$ *je vlastním vektorem matice* $\mathbf{K}$*, pokud platí:*

$$\mathbf{K}\boldsymbol{v} = \lambda \boldsymbol{v}, \tag{2.1}$$

*kde* $\lambda$ *je označováno jako vlastní číslo matice.*

Při aplikaci matice na libovolný vektor dochází k pootočení a změně délky vektoru. Z rovnice (2.1) vyplývá, že aplikujeme-li matici na její vlastní vektor, tento vektor se nepootočí, ale pouze změní svou délku.

Iterativní metody bývají často založeny na opakovaném součinu matice s vektorem. Jedná-li se o vlastní vektor, pak se tento vektor buď postupně zmenšuje k nule nebo zvětšuje až k nekonečnu.

**Věta 2.2.** *[She94]*

*Je-li* $|\lambda| < 1$ *pak se při opakované aplikaci* $\mathbf{K}$ *vlastní vektor* $\boldsymbol{v}$ *odpovídající vlastnímu číslu* $\lambda$ *zmenšuje až k nule.*

*Je-li naopak* $|\lambda| > 1$ *pak se při opakované aplikaci* $\mathbf{K}$ *vlastní vektor* $\boldsymbol{v}$ *odpovídající* $\lambda$ *zvětšuje až k nekonečnu.*

Pro obecný vektor, lze součin matice s vektorem zapsat jako součet jednotlivých součinů vlastních vektorů s maticí $\mathbf{K}$, neboli jako lineární kombinace vlastních vektorů. [She94]

$$\mathbf{K}\boldsymbol{x} = \mathbf{K}\boldsymbol{v}_1 + \mathbf{K}\boldsymbol{v}_2 + \cdots + \mathbf{K}\boldsymbol{v}_i = \lambda_1 \boldsymbol{v}_1 + \lambda_2 \boldsymbol{v}_2 + \cdots + \lambda_n \boldsymbol{v}_n \tag{2.2}$$

Všimněme si, že vlastních čísel je přesně tolik, kolik má čtvercová matice $\mathbf{K}$ řádků. V případě, že matice je singulární, budou některá vlastní čísla nulová.

Lze dokázat, že pokud vektor $\boldsymbol{x}$ při rozložení na součet vlastních vektorů dle (2.2) obsahuje alespoň jeden vlastní vektor, jemuž přísluší vlastní číslo matice $\mathbf{K}$, které je vetší než jedna, pak opakovaná aplikace matice $\mathbf{K}$ na vektor $\boldsymbol{x}$ bude směřovat k nekonečnu. Zároveň, pokud bude velikost všech vlastních čísel menší než jedna, pak bude opakovaná aplikace $\mathbf{K}$ na $\boldsymbol{x}$ směřovat k nule.

Z tohoto důvodu byl zadefinován spektrální poloměr matice.

$$\rho(\mathbf{K}) = \max |\lambda_i|, \tag{2.3}$$

kde $\lambda_i$ je vlastní číslo matice $\mathbf{K}$, $i \in \{1, 2, \dots n\}$.

## 2.2 Soustavy lineárních rovnic a jejich kvadratická forma

Soustavy lineárních rovnic obecně označují množinu jedné, či více rovnic, alespoň se dvěma proměnnými. Tyto soustavy rovnic se běžně zapisují v maticovém tvaru, neboť takový zápis má výhodu nejen v jednoduchosti, ale i v tom, že pro jednotlivé matice můžeme uvažovat užitečné vlastnosti, které usnadňují jejich řešení.

V našem případě se zaměříme na řešení reálných úloh mechaniky, pro které obecně platí, že matice tuhosti, která definuje soustavu lineárních rovnic, je čtvercová, symetrická a pozitivně definitní.

**Definice 2.3.** *Nechť matice $\mathbf{K} \in \mathbb{R}^{n \times n}$ a $\boldsymbol{u} \in \mathbb{R}^n$ a $\boldsymbol{f} \in \mathbb{R}^n$ jsou sloupcové vektory, kde $n \in \mathbb{N}$. Pak námi řešená soustava lineárních rovnic má tvar*

$$\mathbf{K}\boldsymbol{u} = \boldsymbol{f}. \tag{2.4}$$

Tento zápis je vhodný pro přímá řešení (jako je Gaussova eliminace), ale ta se stávají neefektivními pokud $\mathbf{K}$ nabývá na velikosti. Proto se snažíme převést lineární problém na problém kvadratický, pro který by bylo možno využít minimalizační řešiče. Zadefinujeme tedy takzvanou kvadratickou formu výrazu (2.4)

$$f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^T\mathbf{K}\boldsymbol{x} - \boldsymbol{x}^T\boldsymbol{f}, \tag{2.5}$$

která platí pro libovolný vektor $\boldsymbol{x} \in \mathbb{R}^n$

Lze jednoduše dokázat, že pokud je $\mathbf{K}$ symetrická a pozitivně definitní matice, pak skutečné řešení $\boldsymbol{u}$ rovnice (2.4) minimalizuje tuto kvadratickou formu.

$$\nabla f(\boldsymbol{x}) = \frac{1}{2}\mathbf{K}^T\boldsymbol{x} + \frac{1}{2}\mathbf{K}\boldsymbol{x} - \boldsymbol{f}. \tag{2.6}$$

Pokud je $\mathbf{K}$ symetrická můžeme gradient $f(\boldsymbol{x})$ zjednodušit na [She94]

$$\nabla f(\boldsymbol{x}) = \mathbf{K}\boldsymbol{x} - \boldsymbol{f}. \tag{2.7}$$

V případě, kdy je splněno, že matice $\mathbf{K}$ je pozitivně definitní, je zaručeno, že existuje minimum funkce $f(\boldsymbol{x})$. Toto minimum se bude nacházet v bodě,

pro který platí $\nabla f(\boldsymbol{x}) = 0$. Nemusíme dokazovat, že tento výraz odpovídá rovnici (2.4) a tedy minimum funkce $f(\boldsymbol{x})$ odpovídá námi hledanému řešení $\boldsymbol{u}$.

$$\boldsymbol{u} = \arg\min_{\boldsymbol{x}} \left( \frac{1}{2} \boldsymbol{x}^T \mathbf{K} \boldsymbol{x} - \boldsymbol{x}^T \boldsymbol{f} \right) \tag{2.8}$$

### 2.2.1 Vliv vlastních čísel na tvar kvadratické formy

Pro pochopení, proč je řešením rovnice (2.4) právě minimalizace, a ne třeba maximalizace kvadratické formy (2.5), je nutné pochopit, jaký má kvadratická forma geometrický význam a jaký vliv mají na tvar kvadratické formy vlastní čísla matice $\mathbf{K}$

**Definice 2.4.** ***Pozitivně definitní matice***

*Nechť $\mathbf{K} \in \mathbb{R}^{n \times n}$ je čtvercová matice a $\boldsymbol{x} \in \mathbb{R}^n$ je libovolný nenulový sloupcový vektor, pak matice $\mathbf{K}$ je pozitivně definitní, pokud platí:*

$$\boldsymbol{x}^T \mathbf{K} \boldsymbol{x} > 0 \quad \forall \boldsymbol{x} \in \mathbb{R}^n, \|\boldsymbol{x}\| \neq 0 \tag{2.9}$$

Můžeme si všimnout, že tato definice je speciálním případem kvadratické formy. Ze skutečnosti, že libovolný vektor $\boldsymbol{x}$ se dá zapsat jako lineární kombinace vlastních vektorů a z definic 2.1 a 2.4 lze odvodit, že pokud je matice $\mathbf{K}$ pozitivně definitní musí mít všechna vlastní čísla kladná.

Dosadíme-li do definice 2.4 libovolný vlastní vektor dostáváme vztah

$$\boldsymbol{v}^T \mathbf{K} \boldsymbol{v} > 0. \tag{2.10}$$

Následně rozložíme levou stranu nerovnice podle definice o vlastních vektorech.

$$\boldsymbol{v}^T \mathbf{K} \boldsymbol{v} = \lambda \boldsymbol{v}^T \boldsymbol{v} \tag{2.11}$$

$$\lambda \boldsymbol{v}^T \boldsymbol{v} > 0 \tag{2.12}$$

$$\lambda > 0 \tag{2.13}$$

Je snadno vysledovatelné, že pokud se bude zvětšovat velikost vektoru $\boldsymbol{x}$, pak se musí zvětšovat i hodnota součinu (2.9) a to kvadraticky. Součin $\boldsymbol{x}^T \boldsymbol{f}$ pak znamená pouze lineární zvětšení či zmenšení hodnoty kvadratické formy. To znamená, že tvar kvadratické formy s pozitivně definitní symetrickou maticí bude konvexní paraboloid. Lze podobně odvodit i tvary pro negativně definitní, či indefinitní matice.

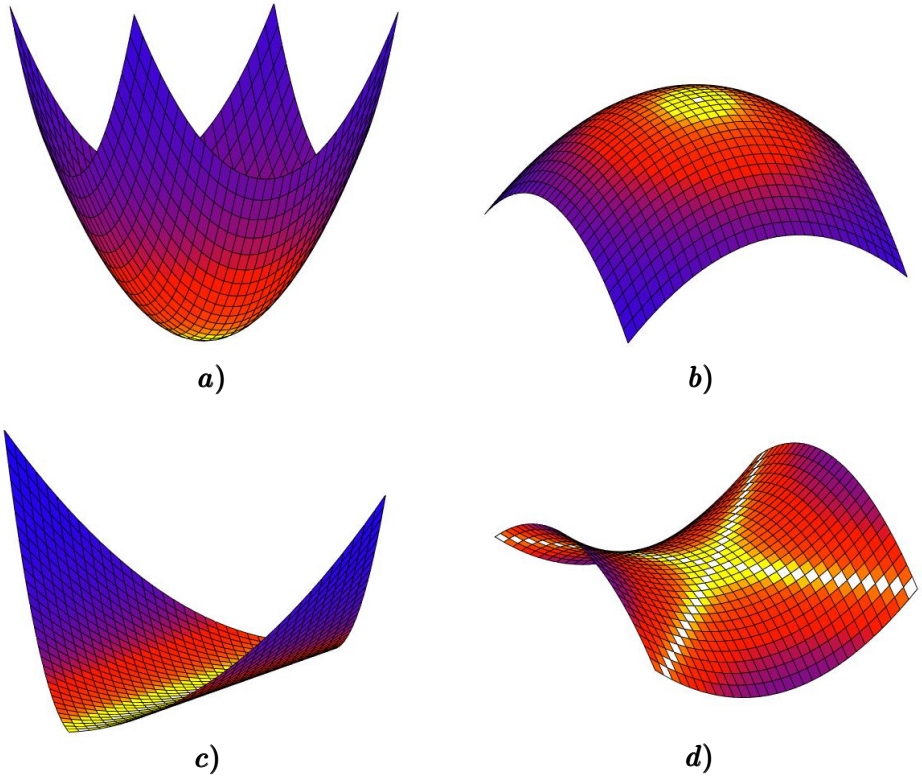
Čtvercová matice $\mathbf{K} \in \mathbb{R}^{n \times n}$ je

$$\text{negativně definitní, když} \quad \boldsymbol{x}^T \mathbf{K} \boldsymbol{x} < 0, \qquad \forall \boldsymbol{x} \in \mathbb{R}^n, \|\boldsymbol{x}\| \neq 0 \tag{2.14}$$

$$\text{negativně semidefinitní, když} \quad \boldsymbol{x}^T \mathbf{K} \boldsymbol{x} \leq 0, \qquad \forall \boldsymbol{x} \in \mathbb{R}^n, \|\boldsymbol{x}\| \neq 0 \tag{2.15}$$

$$\text{pozitivně semidefinitní, když} \quad \boldsymbol{x}^T \mathbf{K} \boldsymbol{x} \geq 0, \qquad \forall \boldsymbol{x} \in \mathbb{R}^n, \|\boldsymbol{x}\| \neq 0 \tag{2.16}$$

$$\begin{aligned} \text{indefinitní, když} \quad & \boldsymbol{x}^T \mathbf{K} \boldsymbol{x} > 0 \ \wedge \ \boldsymbol{y}^T \mathbf{K} \boldsymbol{y} < 0, \\ & \exists \boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n, \|\boldsymbol{x}\| \neq 0, \|\boldsymbol{y}\| \neq 0 \end{aligned} \tag{2.17}$$




**Obrázek 2.1:** Příklad tvarů kvadratických funkcí pro $\mathbf{K} \in \mathbb{R}^{2\times 2}$.
**a)**Pozitivně definitní matice, **b)**Negativně definitní matice,
**c)** Pozitivně semidefinitní (singulární) matice, **d)** Indefinitní matice.
Barevnost označuje odchylku plochy od optimálních hodnot.

Je tedy zřejmé, že chceme-li nalézt řešení rovnice (2.4) jako stacionární bod kvadratické formy (2.5), pak, pokud je $\mathbf{K}$ pozitivně definitní, se toto řešení nachází v minimu kvadratické formy. Pokud by $\mathbf{K}$ byla negativně definitní, pak by bylo řešení v maximu (2.5).

U indefinitní matice bychom prostou minimalizací řešení nedosáhli, protože se jedná o sedlobodovou úlohu. Z toho důvodu nelze běžné gradientní metody, o kterých se budeme bavit dále, využít pro řešení indefinitních úloh.

Tvary kvadratických forem z obrázku 2.1 lze zobecnit do libovolného počtu rozměrů a z nich vyplývající poznatky lze užít pro libovolně rozměrnou úlohu.

Znalost vlastních čísel a tedy i tvaru kvadratické formy nám tedy dává možnost předem posoudit, zda metoda, kterou budeme chtít použít, je vhodná a zda výsledné řešení je jednoznačné nebo nikoliv.

# Kapitola 3

# Metoda největšího spádu

Metoda největšího spádu, nebo-li „Method of the steepest descent“, bude prvním z několika iteračních postupů, kterými se budeme v tomto textu zabývat. Při tomto způsobu výpočtu není naším úkolem spočíst výsledek naprosto bezchybně, ale dovolujeme si drobnou nepřesnost. Tato skutečnost nám dovoluje nevyjadřovat výsledek přímo z rovnic, ale použít postup, který je svou podstatou opačný. Nejprve odhadneme výsledné řešení a poté ověříme, že toto řešení odpovídá zadaným rovnicím. Tento postup pak opakujeme, dokud nedosáhneme požadované odchylky od skutečného řešení.

V našem případě případě hledáme minimum na kvadratické ploše. Při aplikaci metody největšího spádu je tedy nejprve zvolen počáteční bod $\boldsymbol{x}_0$, ze kterého se pak iterativním způsobem postupuje k nejnižšímu místu na ploše. Směr největšího klesání je dán opačným směrem gradientu funkce $f(\boldsymbol{x})$ a je tedy snadno určitelný jako záporná rovnice (2.7).

$$-\nabla f(\boldsymbol{x}_i) = \boldsymbol{f} - \mathbf{K}\boldsymbol{x}_i \tag{3.1}$$

Máme bod $\boldsymbol{x}_i$ a směr, ve kterém můžeme postoupit, abychom zmenšili odchylku následujícího bodu od skutečného řešení. Tuto odchylku nelze přímo zjistit, protože neznáme vektor $\boldsymbol{u}$.

**Definice 3.1.** ***Odchylka od skutečného řešení*** *[She94]*

*Pokud $\boldsymbol{u} \in \mathbb{R}^n$ je vektorem skutečného řešení a $\boldsymbol{x}_i \in \mathbb{R}^n$ je vektorem řešení dosaženého v kroku $i \in \mathbb{N}$, pak vektor*

$$\boldsymbol{e}_i = \boldsymbol{x}_i - \boldsymbol{u} \tag{3.2}$$

*nazýváme chybou, nebo odchylkou od skutečného řešení.*

*Pokud vektor $\boldsymbol{f}$ a matice $\mathbf{K}$ odpovídají výše zadefinovanému problému (2.5), pak*

$$\boldsymbol{r}_i = \boldsymbol{f} - \mathbf{K}\boldsymbol{x}_i \tag{3.3}$$

*nazýváme reziduem.*

Reziduum zobrazuje, jak velký je rozdíl součinu $\mathbf{K}\boldsymbol{x}_i$ a skutečné hodnoty $\boldsymbol{f}$, neboli $\mathbf{K}\boldsymbol{u}$. Je zřejmé, že reziduum $\boldsymbol{r}_i$ odpovídá chybě $\boldsymbol{e}_i$ projektované přes matici $-\mathbf{K}$ do stejného prostoru, v jakém leží $\boldsymbol{f}$.

$$\boldsymbol{r}_i = \boldsymbol{f} - \mathbf{K}(\boldsymbol{e}_i + \boldsymbol{u}) = \boldsymbol{f} - \mathbf{K}\boldsymbol{u} - \mathbf{K}\boldsymbol{e}_i \tag{3.4}$$

$$\boldsymbol{r}_i = -\mathbf{K}\boldsymbol{e}_i \tag{3.5}$$

Je důležité si uvědomit, že

$$\boldsymbol{r}_i = -\nabla f(\boldsymbol{x}_i). \tag{3.6}$$

Tedy reziduum je zároveň směrem největšího klesání a tedy i směrem postupu do následné iterace vyjádřený explicitně.

Už tedy víme, jaký směrem se máme vydat, abychom se dostali do nejhlubšího místa plochy. Zbývá zjistit, jak dlouhý by daný krok měl být.

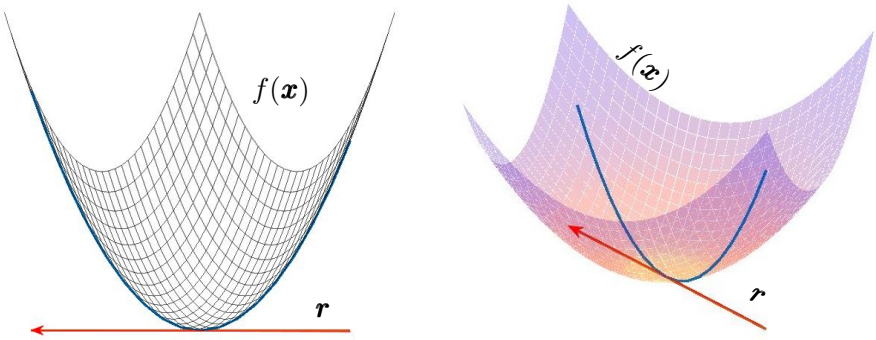
Zadefinujme tedy tzv. délku kroku $\alpha \in \mathbb{R}$, pro kterou platí [She94]

$$\boldsymbol{x}_{i+1} = \boldsymbol{x}_i + \alpha \boldsymbol{r}_i \tag{3.7}$$

pro všechna $i \in \mathbb{N}$

Zároveň jsme tak explicitně zapsali to, o čem jsme hovořili výše, že z bodu $\boldsymbol{x}_i$ do bodu $\boldsymbol{x}_{i+1}$ se dostaneme přičtením rezidua s optimální délkou.

Představujeme-li si námi řešenou kvadratickou formu jako paraboloid, pak reziduum $\boldsymbol{r}$ je vektor pohybující se na definičním oboru funkce $f(\boldsymbol{x})$. Optimální délka rezidua je pak taková, která nás přesune do bodu s nejnižší funkční hodnotou. Všechny funkční hodnoty příslušící danému směru mají tvar kolmého průmětu přímky dané směrem $\boldsymbol{r}$ a bodem $\boldsymbol{x}_i$ na plochu funkce $f(\boldsymbol{x})$. Takový průmět má tvar konvexní křivky.




a) b)

**Obrázek 3.1:** Zobrazení rezidua na kvadratickou formu
a) Nárysný pohled na průmět rezidua na plochu $f(\boldsymbol{x})$,
b) axonometrický pohled na průmět rezidua na plochu $f(\boldsymbol{x})$

Optimální směr rezidua tedy lze nalézt z minima této křivky, zjištěné z parciální derivace podle $\alpha$ rovné nule. [She94]

$$\frac{d}{d\alpha} f\left(\boldsymbol{x}_{i+1}(\alpha)\right) = 0 \tag{3.8}$$

A dále pomocí řetízkového pravidla

$$\nabla f(\boldsymbol{x}_{i+1})^T \frac{d}{d\alpha_i} \boldsymbol{x}_{i+1}(\alpha) = 0 \tag{3.9}$$

$$\nabla f(\boldsymbol{x}_{i+1})^T \frac{d}{d\alpha_i} (\boldsymbol{x}_i + \alpha_i \boldsymbol{r}_i) = 0 \tag{3.10}$$

$$\nabla f(\boldsymbol{x}_{i+1})^T \boldsymbol{r}_i = 0. \tag{3.11}$$

Rovnice (3.11) doslovně znamená, že optimální $\alpha$ je takové, pro které je $\nabla f(\boldsymbol{x}_{i+1})$ kolmé na $\boldsymbol{r}_i$.

Spojíme-li rovnice (3.6) a (3.11) pak [She94]

$$\boldsymbol{r}_{i+1}^T \boldsymbol{r}_i = 0 \tag{3.12}$$

a využijeme-li rovnici 3.3 dostáváme postup

$$(\boldsymbol{f} - \mathbf{K}\boldsymbol{x}_{i+1})^T \boldsymbol{r}_i = 0 \tag{3.13}$$

$$[\boldsymbol{f} - \mathbf{K}(\boldsymbol{x}_i + \alpha_i \boldsymbol{r}_i)]^T \boldsymbol{r}_i = 0 \tag{3.14}$$

$$(\boldsymbol{f} - \mathbf{K}\boldsymbol{x}_i)^T \boldsymbol{r}_i - (\alpha_i \mathbf{K}\boldsymbol{r}_i)^T \boldsymbol{r}_i = 0 \tag{3.15}$$

$$\underbrace{(\boldsymbol{f} - \mathbf{K}\boldsymbol{x}_i)^T}_{\boldsymbol{r}_i^T} \boldsymbol{r}_i = (\alpha_i \underbrace{\mathbf{K}}_{\mathbf{K}=\mathbf{K}^T} \boldsymbol{r}_i)^T \boldsymbol{r}_i \tag{3.16}$$

$$\boldsymbol{r}_i^T \boldsymbol{r}_i = \alpha_i \boldsymbol{r}_i^T \mathbf{K} \boldsymbol{r}_i \tag{3.17}$$

$$\alpha_i = \frac{\boldsymbol{r}_i^T \boldsymbol{r}_i}{\boldsymbol{r}_i^T \mathbf{K} \boldsymbol{r}_i}. \tag{3.18}$$

**Definice 3.2.** ***Délka kroku***

*Je-li $\boldsymbol{r}_i$ reziduum zadefinované rovnicí (3.3) a matice $\mathbf{K}$ je symetrická a pozitivně definitní, pak*

$$\alpha_i = \frac{\boldsymbol{r}_i^T \boldsymbol{r}_i}{\boldsymbol{r}_i^T \mathbf{K} \boldsymbol{r}_i} \tag{3.19}$$

*nazýváme délkou kroku největšího klesání, pro libovolné $i \in \mathbb{N}$.*

Spojíme-li tedy jednotlivé rovnice, dostaneme kompletní algoritmus pro výpočet metodou největšího spádu.

**Definice 3.3.** ***Metoda největšího spádu*** *[She94]*

*Nechť $\mathbf{K} \in \mathbb{R}^{n \times n}$ a je symetrická a pozitivně definitní a vektor $\boldsymbol{f} \in \mathbb{R}^n$. Pak problém zapsaný rovnicí $\mathbf{K}\boldsymbol{u} = \boldsymbol{f}$, lze pro libovolně zvolený počáteční bod $\boldsymbol{x}_0$ řešit jako:*

$$\boldsymbol{x}_0 \in \mathbb{R}^n \tag{3.20}$$

***while*** $\|\boldsymbol{r}_i\| > \epsilon$

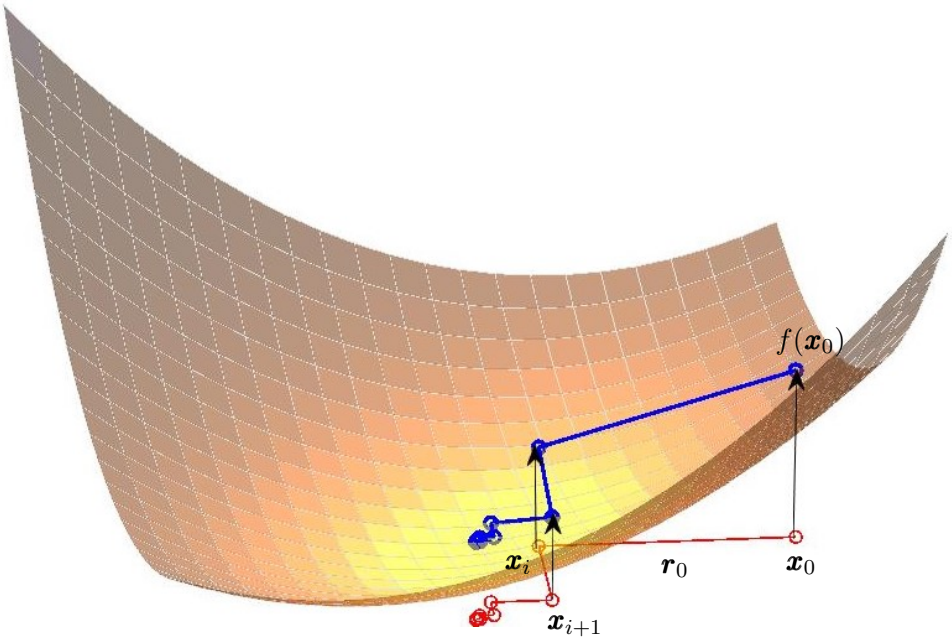
$$\boldsymbol{r}_i = \boldsymbol{f} - \mathbf{K}\boldsymbol{x}_i \tag{3.21}$$

$$\alpha_i = \frac{\boldsymbol{r}_i^T \boldsymbol{r}_i}{\boldsymbol{r}_i^T \mathbf{K} \boldsymbol{r}_i} \tag{3.22}$$

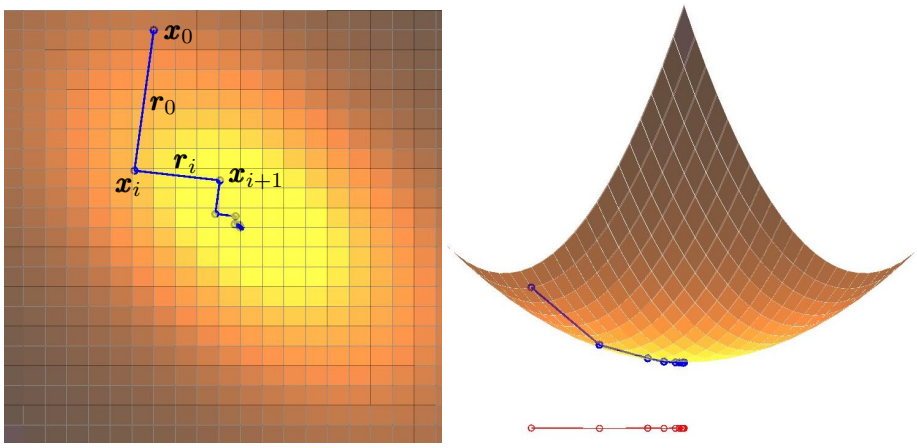
$$\boldsymbol{x}_{i+1} = \boldsymbol{x}_i + \alpha_i \boldsymbol{r}_i \tag{3.23}$$

***end***

*pro $i \in \mathbb{N}$. Konstanta $\epsilon$ značí námi zvolenou toleranci chyby v reziduu $\boldsymbol{r}_i$.*




**Obrázek 3.2:** Příklad průběhu metody největšího spádu, $\mathbf{K} \in \mathbb{R}^{2\times 2}$




a) b)

**Obrázek 3.3:** Pohledy na příklad průběhu metody největšího klesání

a) Průběh $\boldsymbol{r}$ při metodě největšího spádu,
b) Pohled na změnu $f(x)$ při metodě největšího spádu

# Kapitola 4

# Metoda sdružených gradientů

Metoda největšího spádu má svou výhodu ve své jednoduchosti. Pokud se nám navíc povede zvolit $\boldsymbol{x}_0$ tak, že vektor $\boldsymbol{r}_0$ je roven vlastnímu vektoru, pak dojdeme ke správnému řešení hned po první iteraci.

Tato metoda má ale i své nevýhody. Používá vždy směr z předchozího kroku, který je ve směru největšího spádu. Jak můžete vidět z obrázku 3.3, tento směr nejde k řešení přímou cestou, ale její průběh má tvar mnohočetné lomenice. Pro rozšíření informací ke konvergenci metody největšího klesání viz [She94].

Nebylo by tedy možné najít směr, který by byl lepší?

Víme například, že každé dva po sobě jdoucí směry $\boldsymbol{r}$ v metodě největšího spádu jsou na sebe kolmé. Co kdyby například každý krok byl právě tak dlouhý, aby vektor výsledné chyby $\boldsymbol{e}_{i+1}$ byl kolmý na daný směr? To by v podstatě znamenalo, že v každém kroku budeme schopni zarovnat naše řešení k jedné souřadnici skutečného řešení.

Zkusme tedy zadefinovat novou rovnici pro hledání následujícího bodu iterace.[She94]

$$\boldsymbol{x}_{i+1} = \boldsymbol{x}_i + \alpha_i \boldsymbol{d}_i, \tag{4.1}$$

kde $\boldsymbol{d}_0, \boldsymbol{d}_1 \dots \boldsymbol{d}_{m-1}$ je řada na sebe kolmých vyhledávacích směrů pro $\boldsymbol{d}_i \in \mathbb{R}^n, \forall i \in \{1, 2, \dots, m-1\}, m \in \mathbb{N}$. Pro každé $\boldsymbol{d}_i$ navíc platí

$$\boldsymbol{d}_i^T \boldsymbol{e}_{i+1} = 0 \tag{4.2}$$

tedy, že každý vyhledávací směr bude kolmý na následnou odchylku od skutečného řešení.

Zkusme tedy odvodit délku kroku $\alpha_i$. Nemusíme dokazovat, že přičtemi-li změnu bodu $\boldsymbol{x}$, neboli $\alpha \boldsymbol{d}$, k chybě, dostaneme stejným způsobem jako jsme dostali nové $\boldsymbol{x}$ i novou chybu $\boldsymbol{e}$. [She94]

$$\boldsymbol{d}_i^T (\boldsymbol{e}_i + \alpha_i \boldsymbol{d}_i) = 0 \tag{4.3}$$

$$\alpha_i = -\frac{\boldsymbol{d}_i^T \boldsymbol{e}_i}{\boldsymbol{d}_i^T \boldsymbol{d}_i} \tag{4.4}$$

Naneštěstí délka směru $\alpha$ závisí na neznámé chybě $\boldsymbol{e}_i$, proto nemůžeme za těchto podmínek délku kroku spočítat.

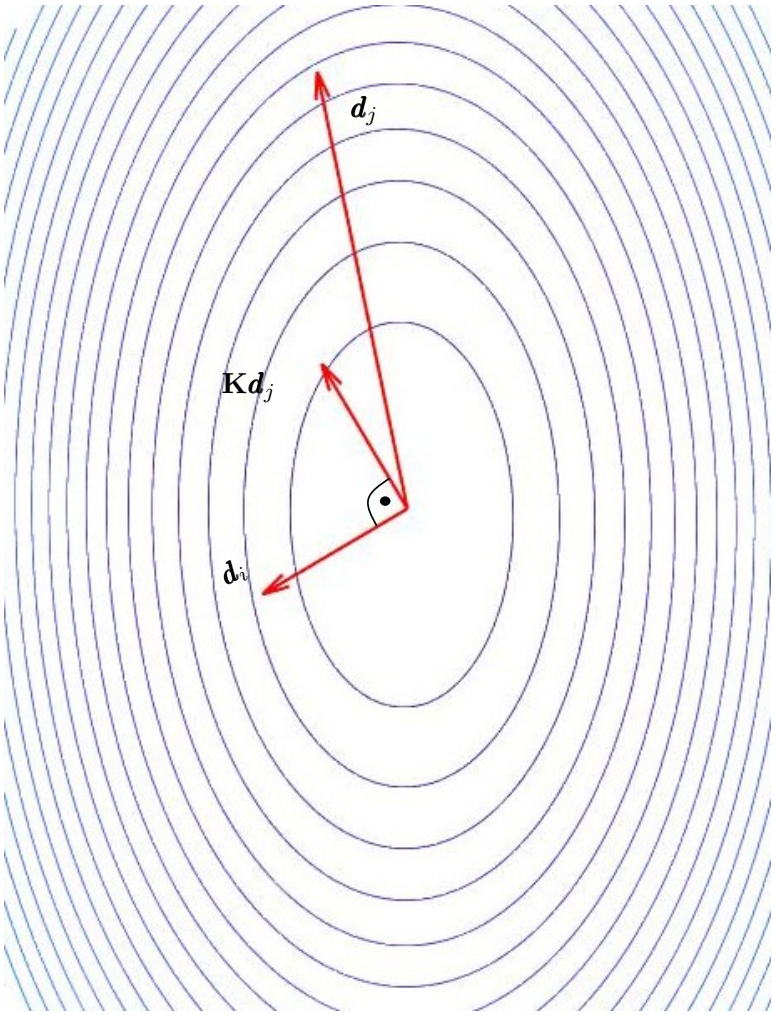
Mohlo by se zdát, že jsme s rovnicí (4.4) uvízli na mrtvém bodě, ale zadefinujme pojem konjugovaných (sdružených) vektorů.

**Definice 4.1.** ***Konjugované vektory*** [*She94*]
*Platí-li*

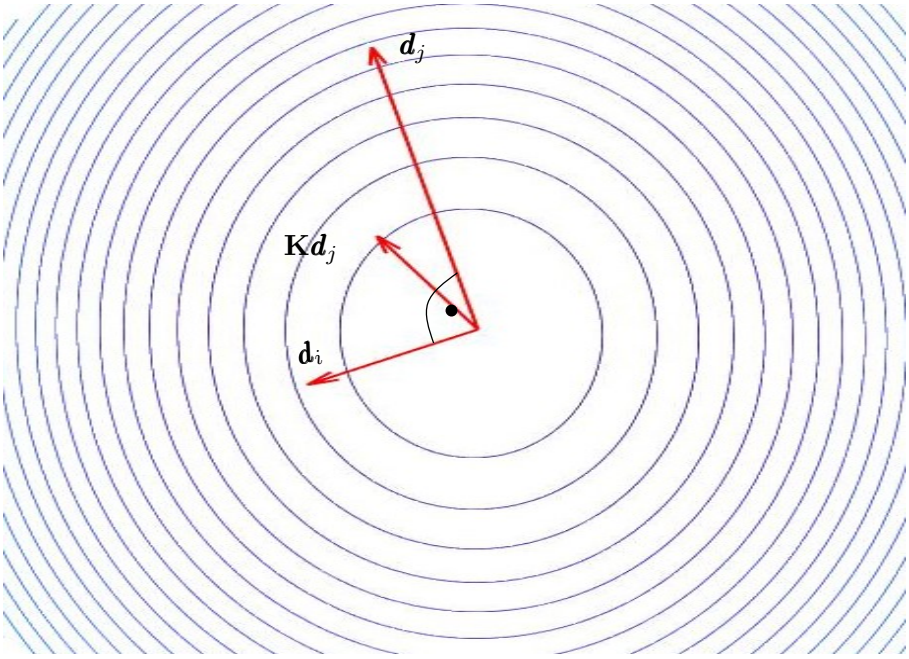
$$\boldsymbol{d}_i^T \mathbf{K} \boldsymbol{d}_j = 0, \tag{4.5}$$

*pro libovolné* $\boldsymbol{d}_i, \boldsymbol{d}_j \in \mathbb{R}^n$ *a* $\mathbf{K} \in \mathbb{R}^{n \times n}$, *kde* $n \in \mathbb{N}$, *pak vektory* $\boldsymbol{d}_i$ *a* $\boldsymbol{d}_j$ *nazýváme konjugované (sdružené) přes matici* $\mathbf{K}$, *nebo-li K-orthogonální.*

Geometrický význam této vlastnosti je velmi zajímavý. Ve své podstatě nám tato definice říká, že pokud naše úloha $\mathbf{K}\boldsymbol{u} = \boldsymbol{f}$ má kvadratickou formu, jejíž tvar je mnohorozměrný, eliptický či rotační paraboloid, a ve stejném prostoru se nacházejí dva vektory $\boldsymbol{d}_i$ a $\boldsymbol{d}_j$, pak pokud tento prostor zdeformujeme tak, aby se z této kvadratické formy stala rotační plocha, pak tyto vektory budou na sebe kolmé.




**Obrázek 4.1:** Konjugované vektory, znázornění kolmosti $\boldsymbol{d}_i^T$ na $\mathbf{K}\boldsymbol{d}_j$




**Obrázek 4.2:** Konjugované vektory, kolmost $\boldsymbol{d}_i$ a $\boldsymbol{d}_j$ na konjugovaném prostoru

Představme si tedy, že chyba je úsečka určená vektorem $\boldsymbol{e}$ a počátkem ve skutečném řešení úlohy, tedy v místě vrcholu kvadratické plochy. Zavedeme-li podmínku, že $\boldsymbol{d}_i$ je K-ortogonální na $\boldsymbol{e}_{i+1}$ zjistíme, že $\boldsymbol{d}_i$ je vektor tečný ke křivkám, které představují ekvidistanty.

Ideální délka kroku ve směru vektoru $\boldsymbol{d}_i$ je tedy taková, která minimalizuje funkční hodnotu $f(\boldsymbol{x}_{i+1}(\alpha_i))$, a to je taková, při které se dostaneme přímo do bodu, ve kterém je v definičním oboru vektor $\boldsymbol{d}_i$ tečný k ekvidistantě.

Z toho vyplývá, že předpoklad K-ortogonality chyby a vyhledávacího směru můžeme ověřit obdobným postupem jako v rovnici (3.8) u metody největšího klesání. [She94]

$$\frac{d}{d\alpha_i} f(\boldsymbol{x}_{i+1}(\alpha_i)) = 0 \tag{4.6}$$

$$\nabla f(\boldsymbol{x}_{i+1}(\alpha_i))^T \frac{d}{d\alpha} \boldsymbol{x}_{i+1}(\alpha_i) = 0 \tag{4.7}$$

Z rovnice (3.6) víme, že $\boldsymbol{r}_{i+1} = -\nabla f(\boldsymbol{x}_{i+1})$.

$$-\boldsymbol{r}_{i+1} \frac{d}{d\alpha}(\boldsymbol{x}_i + \alpha_i \boldsymbol{d}_i) = 0 \tag{4.8}$$

$$-\boldsymbol{r}_{i+1}^T \boldsymbol{d}_i = 0 \tag{4.9}$$

A reziduum je pouze projektovanou chybou viz (3.5). Tedy $\boldsymbol{r}_{i+1} = -\mathbf{K}\boldsymbol{e}_{i+1}$.

$$\boldsymbol{e}_{i+1}^T \mathbf{K}^T \boldsymbol{d}_i = 0 \tag{4.10}$$

$$\boldsymbol{d}_i^T \mathbf{K} \boldsymbol{e}_{i+1} = 0 \tag{4.11}$$

Předpoklad K-ortogonality je tedy splněn a my můžeme poopravit rovnici (4.2) a postup, který ji následuje. [She94]

$$\boldsymbol{d}_i^T \mathbf{K}(\boldsymbol{e}_i + \alpha \boldsymbol{d}_i) = 0 \tag{4.12}$$

$$\alpha = -\frac{\boldsymbol{d}_i^T \mathbf{K} \boldsymbol{e}_i}{\boldsymbol{d}_i^T \mathbf{K} \boldsymbol{d}_i} \tag{4.13}$$

$$\alpha = \frac{\boldsymbol{d}_i^T \boldsymbol{r}_i}{\boldsymbol{d}_i^T \mathbf{K} \boldsymbol{d}_i} \tag{4.14}$$

Tento výraz už dokážeme bez potíží spočítat, navíc pokud se $\boldsymbol{d}$ bude rovnat reziduu, dostaneme opět výrazy pro metodu největšího spádu. Tento vzorec je tedy zobecněním výše odvozených vztahů.

Problémem nadále zůstává, co je a jak se vypočítá vektor $\boldsymbol{d}$.

Vektor $\boldsymbol{d}$ odvodíme pomocí Gram-Schmidtovy ortogonalizace.

**Definice 4.2.** ***Vektor d z Gram-Schmidtovy ortogonalizace*** [*She94*]
*Nechť existuje množina $m$ lineárně nezávislých vektorů $\boldsymbol{l}_i, i \in \{0, 1, \dots m-1\}$, $m \in \mathbb{N}$, potom pro každé $i$ vznikne vektor $\boldsymbol{d}_i$ z vektoru $\boldsymbol{l}_i$ odstraněním všech složek, které nejsou K-ortogonální na předcházející vektory $\boldsymbol{d}_0, \boldsymbol{d}_i \dots \boldsymbol{d}_{i-1}$.*

$$\boldsymbol{d}_0 = \boldsymbol{l}_0 \qquad i = 0 \tag{4.15}$$

$$\boldsymbol{d}_i = \boldsymbol{l}_i + \sum_{k=1}^{i-1} \beta_{ik} \boldsymbol{d}_k \qquad i \neq 0 \tag{4.16}$$

Gramm-Schmidtova ortogonalizace nám tedy generuje sadu vektorů $\boldsymbol{d}$, z nichž každý je K-ortogonální na všechny ostatní.

Pokud předpokládáme, že $i > j$, lze z K-ortogonality vektorů $\boldsymbol{d}$ odvodit $\beta_{ik}$.

$$0 = \boldsymbol{d}_i^T \mathbf{K} \boldsymbol{d}_j = \boldsymbol{l}_i^T \mathbf{K} \boldsymbol{d}_j + \sum_{k=1}^{i-1} \beta_{ik} \boldsymbol{d}_k^T \mathbf{K} \boldsymbol{d}_j \tag{4.17}$$

Všimněme si, že pokud $k \neq j$, pak se $\boldsymbol{d}_k^T \mathbf{K} \boldsymbol{d}_j = 0$.

$$-\boldsymbol{l}_i^T \mathbf{K} \boldsymbol{d}_j = \beta_{ij} \boldsymbol{d}_j^T \mathbf{K} \boldsymbol{d}_j \tag{4.18}$$

$$\beta_{ij} = -\frac{\boldsymbol{l}_i^T \mathbf{K} \boldsymbol{d}_j}{\boldsymbol{d}_j^T \mathbf{K} \boldsymbol{d}_j} \tag{4.19}$$

Problém Gram-Schmidtovy ortogonalizace je, že musíme v paměti držet všechny předchozí vektory pro výpočet dalších, takže náročnost úlohy postupně stoupá.

Vraťme se teď k metodě sdružených gradientů. Metoda sdružených gradientů dosazuje za vektory $\boldsymbol{l}$ rezidua. Z rovnice (4.11) víme, že $\boldsymbol{d}_i$ je K-ortogonální na $\boldsymbol{e}_{i+1}$. Pro každou následující chybu v kroku $i+p$, $i+p < m-1$ musí platit

$$\boldsymbol{e}_{i+p} = \boldsymbol{e}_{i+1} + \sum_{p=1}^{m-i-1} (\alpha_{i+p}\boldsymbol{d}_{i+p}) \tag{4.20}$$

$$-\boldsymbol{d}_i^T\mathbf{K}\boldsymbol{e}_{i+p} = \underbrace{-\boldsymbol{d}_i^T\mathbf{K}\boldsymbol{e}_{i+1}}_{0} \underbrace{-\boldsymbol{d}_i^T\mathbf{K}\sum_{p=1}^{m-i-1}(\alpha_{i+p}\boldsymbol{d}_{i+p})}_{0}. \tag{4.21}$$

Nulovost druhé části výrazu vyplývá z předpokladu, že vektor $\boldsymbol{d}_i$ je K-ortogonání na všechny následující vyhledávací směry $\boldsymbol{d}_{i+p}$. Pokud tedy přepíšeme předchozí rovnici tak, že $j = i+p$, pak

$$-\boldsymbol{d}_i^T\mathbf{K}\boldsymbol{e}_j = 0 \qquad i < j \tag{4.22}$$

$$\boldsymbol{d}_i^T\boldsymbol{r}_j = 0 \qquad i < j. \tag{4.23}$$

Navážu-li na rovnice (4.16) a (4.23), mohu odvodit vlastnost, která bude užita až později.

$$\boldsymbol{d}_i^T\boldsymbol{r}_j = \boldsymbol{r}_i^T\boldsymbol{r}_j + \sum_{k=0}^{i-1}\beta_{ik}\boldsymbol{d}_k^T\boldsymbol{r}_j \tag{4.24}$$

$$\text{pro} \quad i < j \qquad \boldsymbol{r}_i^T\boldsymbol{r}_j = 0 \tag{4.25}$$

$$\text{pro} \quad i = j \qquad \boldsymbol{r}_i^T\boldsymbol{r}_i = \boldsymbol{d}_i^T\boldsymbol{r}_i \tag{4.26}$$

Dle rovnice (3.5) lze navíc odvodit, že vektor $\boldsymbol{r}_{i+1}$ je lineární kombinací $\mathbf{K}\boldsymbol{e}_i$ a $\alpha\mathbf{K}\boldsymbol{d}_i$

$$\boldsymbol{r}_{i+1} = -\mathbf{K}\boldsymbol{e}_{i+1} \tag{4.27}$$

$$\boldsymbol{r}_{i+1} = -\mathbf{K}(\boldsymbol{e}_i + \alpha_i\boldsymbol{d}_i) \tag{4.28}$$

$$\boldsymbol{r}_{i+1} = -\mathbf{K}\boldsymbol{e}_i - \alpha_i\mathbf{K}\boldsymbol{d}_i \tag{4.29}$$

$$\boldsymbol{r}_{i+1} = \boldsymbol{r}_i - \alpha_i\mathbf{K}\boldsymbol{d}_i \tag{4.30}$$

Pokud do rovnice (4.15) z Gramm-Schmidtovi ortogonalizace dosadíme dosadíme $\boldsymbol{l}_i = \boldsymbol{r}_i$, dostaneme vztah

$$\boldsymbol{d}_0 = \boldsymbol{r}_0 \tag{4.31}$$

Následně můžeme zobecnit rovnici (4.30) a z ní zjistíme že libovolné $\boldsymbol{r}_j$ sestává z vektorů $\boldsymbol{d}$.

$$\boldsymbol{r}_j = \boldsymbol{d}_0 - \sum_{k=1}^{j-1}(\alpha_k\mathbf{K}\boldsymbol{d}_k) \tag{4.32}$$

Z (4.23) a předchozí rovnice vyplývá, že každé reziduum musí být kolmé na všechny ostatní vektory $\boldsymbol{r}$.

$$\boldsymbol{r}_i^T \boldsymbol{r}_j = \underbrace{\boldsymbol{r}_i^T \boldsymbol{d}_0}_{0} - \underbrace{\boldsymbol{r}_i^T \sum_{k=1}^{j-1} (\alpha_k \mathbf{K} \boldsymbol{d}_k)}_{0} \qquad i \neq 0 \tag{4.33}$$

$$\boldsymbol{r}_i^T \boldsymbol{r}_j = 0 \qquad i \neq 0, i \neq j \tag{4.34}$$

Pokusme se tedy pro případ, kdy volíme reziduum jako naše lineárně nezávislé vektory, zjednodušit Gram-Schmidtovu konstantu $\beta$ pro $i > 0$. [She94]

$$\boldsymbol{r}_i^T \boldsymbol{r}_{j+1} = \boldsymbol{r}_i^T \boldsymbol{r}_j - \alpha_i \boldsymbol{r}_i^T \boldsymbol{K} \boldsymbol{d}_j \tag{4.35}$$

$$\alpha_i \boldsymbol{r}_i^T \boldsymbol{K} \boldsymbol{d}_j = \boldsymbol{r}_i^T \boldsymbol{r}_j - \boldsymbol{r}_i^T \boldsymbol{r}_{j+1} \tag{4.36}$$

Z toho podle rovnice (4.34) lze odvodit, že

$$\boldsymbol{r}_i^T \boldsymbol{K} \boldsymbol{d}_j = \begin{cases} \frac{1}{\alpha_i} \boldsymbol{r}_i^T \boldsymbol{r}_i & i = j \\ -\frac{1}{\alpha_{i-1}} \boldsymbol{r}_i^T \boldsymbol{r}_i & i = j+1 \\ 0 & i \neq j+1, i \neq j. \end{cases} \tag{4.37}$$

A potom můžeme odvodit ze vztahu (4.18) pro který platí předpoklad $i > j$, následující tvrzení.

$$-\beta_{ij} \boldsymbol{d}_{i-1}^T \mathbf{K} \boldsymbol{d}_{i-1} = \begin{cases} -\frac{1}{\alpha_{i-1}} \boldsymbol{r}_i^T \boldsymbol{r}_i & i = j+1 \\ 0 & i > j+1 \end{cases} \tag{4.38}$$

A z toho vyplývá $\beta_{i,j}$.

$$\beta_{i,i-1} = \begin{cases} \frac{1}{\alpha_{i-1}} \frac{\boldsymbol{r}_i^T \boldsymbol{r}_i}{\boldsymbol{d}_{i-1}^T \mathbf{K} \boldsymbol{d}_{i-1}} & i = j+1 \\ 0 & i > j+1 \end{cases} \tag{4.39}$$

Všimněme si, že už nadále nemusíme zapisovat všechny předchozí vyhledávací vektory. Zjednodušíme zápis $\beta_{i,i-1}$ na $\beta_i$ a použijeme rovnici (4.39), protože v našem případě nás zajímá pouze případ kdy, i a j jsou dva za sebou následující kroky. Dosadíme za $\alpha_{i-1}$ dle rovnice (4.14).

$$\beta_i = \frac{\boldsymbol{d}_{i-1}^T \mathbf{K} \boldsymbol{d}_{i-1}}{\boldsymbol{d}_{i-1}^T \boldsymbol{r}_{i-1}} \frac{\boldsymbol{r}_i^T \boldsymbol{r}_i}{\boldsymbol{d}_{i-1}^T \mathbf{K} \boldsymbol{d}_{i-1}} = \frac{\boldsymbol{r}_i^T \boldsymbol{r}_i}{\boldsymbol{d}_{i-1}^T \boldsymbol{r}_{i-1}} \tag{4.40}$$

Výsledný tvar skaláru $\beta_i$ a $\alpha_i$ dostaneme z výše odvozeného vztahu (4.26):

$$\beta_i = \frac{\boldsymbol{r}_i^T \boldsymbol{r}_i}{\boldsymbol{r}_{i-1}^T \boldsymbol{r}_{i-1}} \qquad \alpha_i = \frac{\boldsymbol{r}_i^T \boldsymbol{r}_i}{\boldsymbol{d}_i^T \mathbf{K} \boldsymbol{d}_i} \tag{4.41}$$

Tímto posledním krokem jsme odvodili celou metodu sdružených gradientů, a tedy ji shrneme.

**Definice 4.3.** ***Metoda CG (Conjugate Gradient)*** [*She94*]
*Nechť* $\mathbf{K} \in \mathbb{R}^{n \times n}$ *je symetrická a pozitivně definitní a vektor* $\boldsymbol{f} \in \mathbb{R}^n$. *Pak problém zapsaný rovnicí* $\mathbf{K}\boldsymbol{u} = \boldsymbol{f}$ *lze pro vhodně zvolený počáteční bod* $\boldsymbol{x}_0$ *řešit jako*

$$\boldsymbol{x}_0 \in \mathbb{R}^n \tag{4.42}$$

$$\boldsymbol{d}_0 = \boldsymbol{r}_0 = \boldsymbol{f} - \mathbf{K}\boldsymbol{x}_0 \tag{4.43}$$

***while*** $\|\boldsymbol{d}_{i+1}\| \leq \epsilon$

$$\alpha_i = \frac{\boldsymbol{r}_i^T \boldsymbol{r}_i}{\boldsymbol{d}_i^T \mathbf{K} \boldsymbol{d}_i} \tag{4.44}$$

$$\boldsymbol{x}_{i+1} = \boldsymbol{x}_i + \alpha_i \boldsymbol{d}_i \tag{4.45}$$

$$\boldsymbol{r}_{i+1} = \boldsymbol{r}_i - \alpha_i \mathbf{K} \boldsymbol{d}_i \tag{4.46}$$

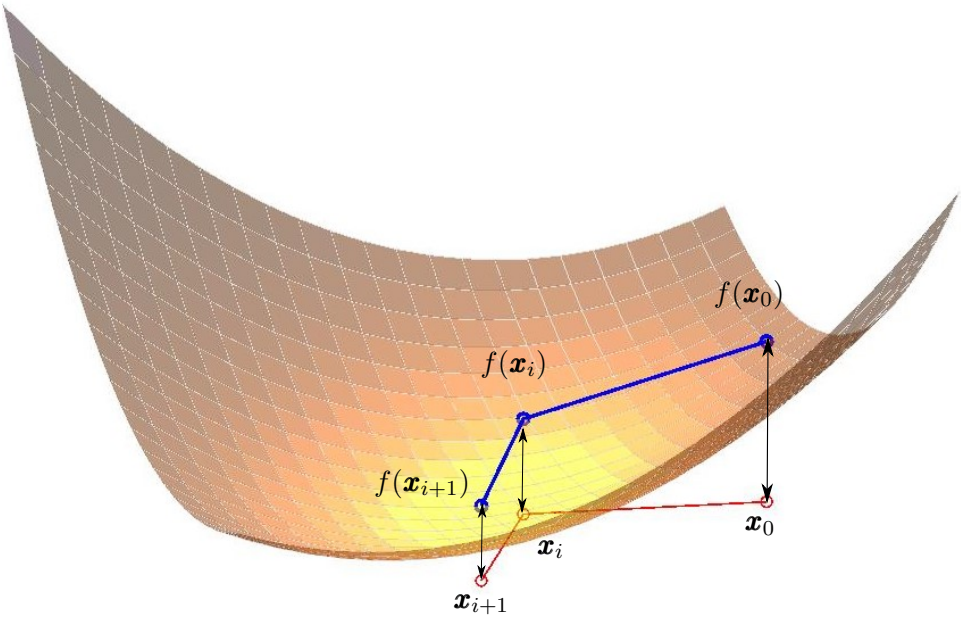
$$\beta_{i+1} = \frac{\boldsymbol{r}_{i+1}^T \boldsymbol{r}_{i+1}}{\boldsymbol{r}_i^T \boldsymbol{r}_i} \tag{4.47}$$

$$\boldsymbol{d}_{i+1} = \boldsymbol{r}_{i+1} + \beta_{i+1} \boldsymbol{d}_i \tag{4.48}$$

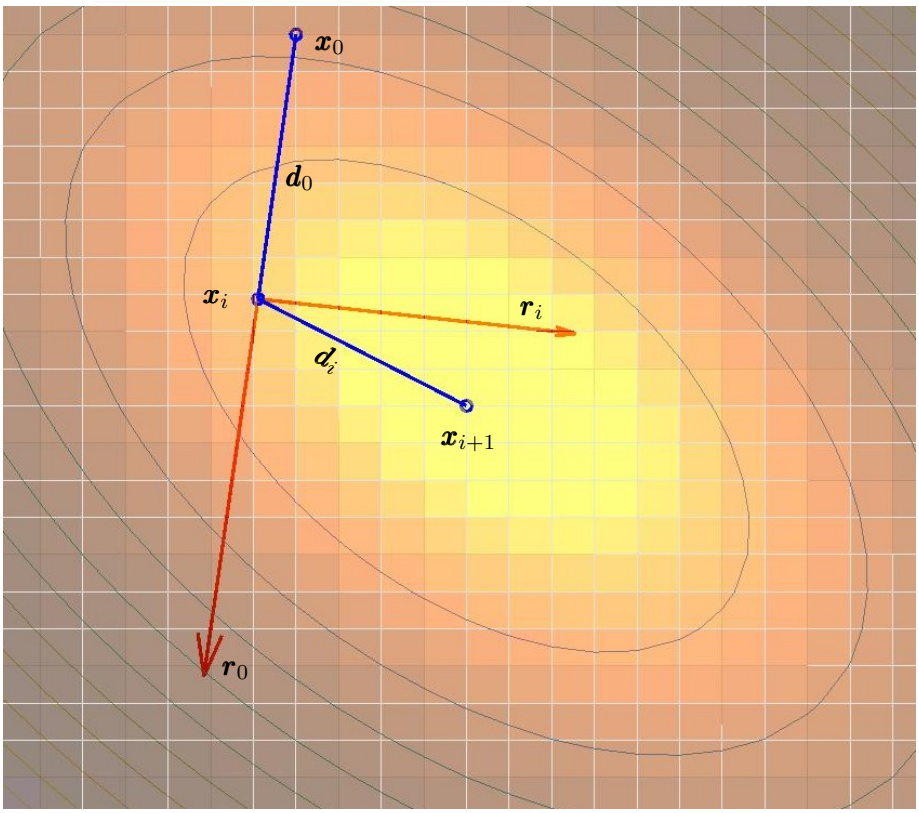
***end***

*pro libovolné* $n \in \mathbb{R}$ *s odchylkou od skutečného řešení rovnou* $\boldsymbol{d}_{i+1}$. *Tolerance* $\epsilon$ *je námi volená hodnota, vybíraná s ohledem na typ úlohy a požadovanou přesnost řešení.*

Z obrázků 4.3 a 4.4 je vidět, jaký významný vliv má konjugace na průběh iterační metody. V dvourozměrném prostoru je metoda již po dvou iteracích, takřka u konce, protože vyhledávací směr zamíří přímo ve směru řešení a nikoliv slepě kolmo na předchozí vyhledávací vektor. To odpovídá tvrzení, že metoda CG v přesné aritmetice vždy nachází řešení maximálně po $n$ krocích. Bohužel obvykle vstupuje do výpočtu výpočtu i numerická chyba.




**Obrázek 4.3:** Příklad průběhu metody sdružených gradientů, $\mathbf{K} \in \mathbb{R}^{2\times 2}$




**Obrázek 4.4:** Pohled na průběh směrových vektorů při metodě CG

# Kapitola 5

# Metoda Lagrangeových multiplikátorů

Metoda Lagrangeových multiplikátorů je matematický postup, který umožňuje najít extrémy funkce vázané podmínkami v podobě jiných funkcí.

V reálných problémech mechaniky lze zavést podmínky podepření některých stupňů volnosti jednoduše tím, že sestavíme matici tuhosti na podproblému, který neobsahuje deformace, které jsou rovné nule. Jednoduše vynecháme příslušné sloupce a řádky z matice $\mathbf{K}$. Proto jsme se v dosavadním výkladu nezabývaly tím, jak podmínky zavést.

Metody, které budeme odvozovat v následujících kapitolách, bohužel obsahují i jiné typy podmínek (například rovnost jednotlivých deformací), a proto zavedeme Lagrangeovy multiplikátory a Langrangeův funkcionál.

Pro větší přehlednost vysvětlíme princip Lagrangeových multiplikátorů na omezeném prostoru i když následující tvrzení lze zobecnit do $\mathbb{R}^n$.

Hledáme extrém vektorové funkce $f(\boldsymbol{x})$, která je vázaná $m$ vazbami $\boldsymbol{g}(\boldsymbol{x}) = \mathbf{0}$,na množině $\mathcal{M}$. V našem případě $m = 2$ a $n = 5$ .

$$\mathcal{M} = \{\boldsymbol{x} \in \mathbb{R}^5, \boldsymbol{g}(\boldsymbol{x}) = \mathbf{0}\}, \tag{5.1}$$

kde $\boldsymbol{g}(\boldsymbol{x}) : \mathbb{R}^5 \to \mathbb{R}^2, \boldsymbol{g}(\boldsymbol{x}) = [g_1(\boldsymbol{x}); g_2(\boldsymbol{x})]$.

Funkce $\boldsymbol{g}(\boldsymbol{x})$ převádí definiční obor z $\mathbb{R}^5$ do oboru hodnot v $\mathbb{R}^2$, proto si pro další postup zvolíme libovolné dvě souřadnice, $x_4$ a $x_5$. Potom pokud v bodě extrému $\boldsymbol{u} \in \mathcal{M}$ je matice parciálních derivací podle $x_4$ a $x_5$

$$\mathbf{G} = \begin{bmatrix} \frac{\partial g_1(\boldsymbol{u})}{\partial x_4} & \frac{\partial g_1(\boldsymbol{u})}{\partial x_5} \\ \frac{\partial g_2(\boldsymbol{u})}{\partial x_4} & \frac{\partial g_2(\boldsymbol{u})}{\partial x_5} \end{bmatrix} \tag{5.2}$$

regulární, existují, podle věty o derivaci implicitní funkce [dO14], takové funkce $\overline{g_1}$ a $\overline{g_2}$, že na okolí bodu $\overline{\boldsymbol{u}} = [u_1; u_2; u_3]$ pro body $\boldsymbol{x} \in \mathcal{M}$ platí

$$x_4 = \overline{g_1}(x_1, x_2, x_3) \tag{5.3}$$

$$x_5 = \overline{g_2}(x_1, x_2, x_3). \tag{5.4}$$

Původní úlohu tedy můžeme přepsat jako

$$f(\boldsymbol{x}) = f\left(x_1, x_2, x_3, \overline{g_1}(x_1, x_2, x_3), \overline{g_2}(x_1, x_2, x_3)\right) \tag{5.5}$$

$$g_1(\boldsymbol{x}) = g_1(x_1, x_2, x_3, \overline{g_1}(x_1, x_2, x_3), \overline{g_2}(x_1, x_2, x_3)) = 0 \tag{5.6}$$

$$g_2(\boldsymbol{x}) = g_2(x_1, x_2, x_3, \overline{g_1}(x_1, x_2, x_3), \overline{g_2}(x_1, x_2, x_3)) = 0. \tag{5.7}$$

Ze všech zúčastněných funkcí se tak stávají složené funkce o třech proměnných. Rozepíšeme-li pak parciální derivace $g_1$ a $g_2$ podle $x_j$ v bodě $\boldsymbol{u}$ , kde $j = 1, 2, 3$ , dostáváme soustavu rovnic

$$\begin{bmatrix} \frac{\partial g_1(\boldsymbol{u})}{\partial x_j} \\ \frac{\partial g_2(\boldsymbol{u})}{\partial x_j} \end{bmatrix} + \begin{bmatrix} \frac{\partial g_1(\boldsymbol{u})}{\partial x_4} & \frac{\partial g_1(\boldsymbol{u})}{\partial x_5} \\ \frac{\partial g_2(\boldsymbol{u})}{\partial x_4} & \frac{\partial g_2(\boldsymbol{u})}{\partial x_5} \end{bmatrix} \begin{bmatrix} \frac{\partial \overline{g_1}(\overline{\boldsymbol{u}})}{\partial x_j} \\ \frac{\partial \overline{g_2}(\overline{\boldsymbol{u}})}{\partial x_j} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \tag{5.8}$$

Pak zjednodušíme zápis pomocí (5.2) a převedeme na pravou stranu.

$$\begin{bmatrix} \frac{\partial g_1(\boldsymbol{u})}{\partial x_j} \\ \frac{\partial g_2(\boldsymbol{u})}{\partial x_j} \end{bmatrix} = -\mathbf{G} \begin{bmatrix} \frac{\partial \overline{g_1}(\overline{\boldsymbol{u}})}{\partial x_j} \\ \frac{\partial \overline{g_2}(\overline{\boldsymbol{u}})}{\partial x_j} \end{bmatrix} \tag{5.9}$$

Díky tomu, že matice $\mathbf{G}$ je regulární lze zní vytvořit inverzní matici.

$$\begin{bmatrix} \frac{\partial \overline{g_1}(\overline{\boldsymbol{u}})}{\partial x_j} \\ \frac{\partial \overline{g_2}(\overline{\boldsymbol{u}})}{\partial x_j} \end{bmatrix} = -\mathbf{G}^{-1} \begin{bmatrix} \frac{\partial g_1(\boldsymbol{u})}{\partial x_j} \\ \frac{\partial g_2(\boldsymbol{u})}{\partial x_j} \end{bmatrix} \tag{5.10}$$

Pokud vrátíme zpět k hledání extrému funkce $f$ uvědomíme si, že místě extrému musí být její parciální derivace podle $\boldsymbol{x}_j$ rovná nule.

$$\frac{\partial f(\boldsymbol{u})}{\partial x_j} + \frac{\partial f(\boldsymbol{u})}{\partial x_4} \frac{\partial \overline{g_1}(\overline{\boldsymbol{u}})}{\partial x_j} + \frac{\partial f(\boldsymbol{u})}{\partial x_5} \frac{\partial \overline{g_2}(\overline{\boldsymbol{u}})}{\partial x_j} = 0 \tag{5.11}$$

Pokud tedy dosadíme z předchozí rovnice získáváme vztah

$$\frac{\partial f(\boldsymbol{u})}{\partial x_j} - \begin{bmatrix} \frac{\partial g_1(\boldsymbol{u})}{\partial x_j} \\ \frac{\partial g_2(\boldsymbol{u})}{\partial x_j} \end{bmatrix}^T \mathbf{G}^{-T} \begin{bmatrix} \frac{\partial f(\boldsymbol{u})}{\partial x_4} \\ \frac{\partial f(\boldsymbol{u})}{\partial x_5} \end{bmatrix} = 0. \tag{5.12}$$

Označíme

$$\boldsymbol{\lambda} = \mathbf{G}^{-T} \begin{bmatrix} \frac{\partial f(\boldsymbol{u})}{\partial x_4} \\ \frac{\partial f(\boldsymbol{u})}{\partial x_5} \end{bmatrix} \tag{5.13}$$

$$\frac{\partial f(\boldsymbol{u})}{\partial x_j} = \begin{bmatrix} \frac{\partial g_1(\boldsymbol{u})}{\partial x_j} \\ \frac{\partial g_2(\boldsymbol{u})}{\partial x_j} \end{bmatrix}^T \begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix}. \tag{5.14}$$

Zároveň obrátíme-li rovnici (5.13) dostáváme

$$\begin{bmatrix} \frac{\partial f(\boldsymbol{u})}{\partial x_4} \\ \frac{\partial f(\boldsymbol{u})}{\partial x_5} \end{bmatrix} = \mathbf{G}^T \boldsymbol{\lambda} \tag{5.15}$$

$$\begin{bmatrix} \frac{\partial f(\boldsymbol{u})}{\partial x_4} \\ \frac{\partial f(\boldsymbol{u})}{\partial x_5} \end{bmatrix} = \begin{bmatrix} \frac{\partial g_1(\boldsymbol{u})}{\partial x_4} & \frac{\partial g_2(\boldsymbol{u})}{\partial x_4} \\ \frac{\partial g_1(\boldsymbol{u})}{\partial x_5} & \frac{\partial g_2(\boldsymbol{u})}{\partial x_5} \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix}. \tag{5.16}$$

Z rovnic (5.14) a (5.16) pak dostáváme výsledné tvrzení o Lagrangeových multiplikátorech.

**Věta 5.1.** ***Tvrzení o Lagrangeových multiplikátorech*** [*Bub06*]

*Nechť $f(\boldsymbol{x})$ má vázaný extrém na množině $\mathcal{M}$ (viz (5.1) v bodě $\boldsymbol{u} \in \mathcal{M}$ a nechť matice parciálních derivací $\mathbf{G}$ je regulární. Pak existuje vektor $\boldsymbol{\lambda} = [\lambda_1, \lambda_2]^T \in \mathbb{R}^2$ takový, že platí*

$$\nabla f(\boldsymbol{u}) = \nabla \boldsymbol{g}(\boldsymbol{u})\boldsymbol{\lambda} = \nabla g_1(\boldsymbol{u})\lambda_1 + g_2(\boldsymbol{u})\lambda_2 \tag{5.17}$$

Toto tvrzení nám v podstatě říká, že v místě extrému musí být tečna k funkci okrajových podmínek totožná s tečnou k vrstevnici omezované funkce, viz obrázek 5.1.

Pro naše další výpočty si vhodně zvolíme funkcionál, který splňuje podmínky (5.17). Opět budeme postupovat v obecném tvaru.

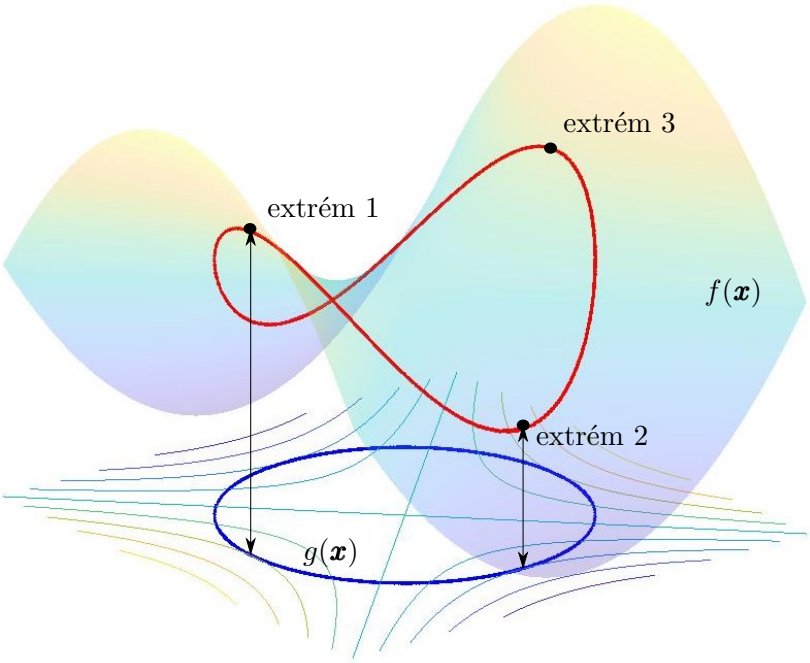
**Věta 5.2.** ***Lagrangeův funkcionál***
*Buď dána funkce $f(\boldsymbol{x}) \in \mathbb{R}^n, n \in \mathbb{N}$ a nechť je dáno $m < n$ podmínek ve tvaru*

$$\boldsymbol{g}(\boldsymbol{x}) = \mathbf{0} \tag{5.18}$$

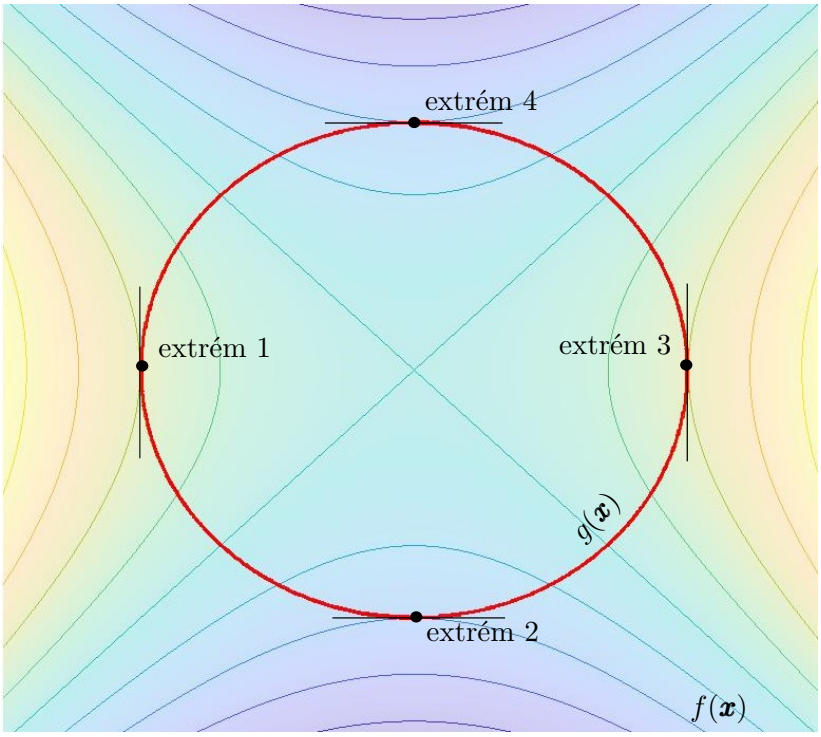
*Pak, má-li funkce*

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}) = f(\boldsymbol{x}) + \boldsymbol{\lambda}^T \boldsymbol{g}(\boldsymbol{x})^T \tag{5.19}$$

*ve svém stacionárním bodě extrém, pak i funkce $f(\boldsymbol{x})$ má v tomto bodě extrém vázaný podmínkami $\boldsymbol{g}(\boldsymbol{x}) = \mathbf{0}$.*




a)




b)

**Obrázek 5.1:** Znázornění extrémů vázaných podmínkami

a) Pohled na průběh funkčních hodnot odpovídajících vymezující funkci
b) Půdorysný pohled znázorňující totožnost tečen v extrémech

# Kapitola 6

# Duální metody výpočtu

V předchozích kapitolách jsme vysvětlili základní iterační metody. Tyto postupy se používají v případech, kdy je matice $\mathbf{K}$ příliš rozměrná a řešení pomocí Gaussovy eliminace by bylo příliš zdlouhavé, nebo přímo neproveditelné, kvůli nárokům na paměť počítače. Za cenu drobného snížení přesnosti výpočtu tak pro rozměrnější úlohy zvyšujeme rychlost a efektivitu výpočtu. Takovým úlohám odpovídají soustavy rovnic vytvořené pomocí metody konečných prvků. Tyto úlohy se běžně pohybují v $\mathbb{R}^{10^6}$ a větších.

Počítače běžně používají takzvaný „řídký“ formát matic. Tento formát využívá toho, že v běžných statických problémech jsou nenulové prvky v matici soustředěny kolem diagonály a ostatní prvky jsou nulové. Proto počítač zapisuje pouze nenulové prvky a jejich umístění uvnitř matice.

Toto opatření ale samo o sobě nestačí. Proto zavádíme duální metody doménové dekompozice.

## 6.1 Duální metoda dělení oblasti

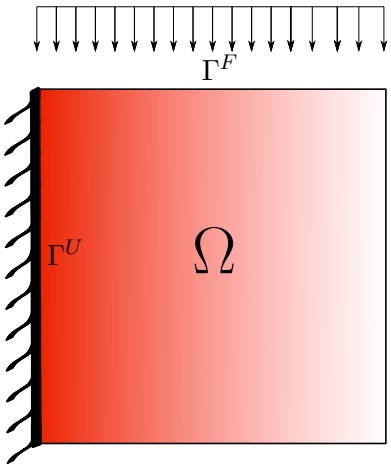
Duální metody využívají takzvané duality optimalizačních úloh. Ta nám říká, že každé optimalizační úloze existuje duální úloha, která má optimum ve stejném místě jako původní úloha. Duální úlohou zde myslíme, že pokud původní funkce byla minimalizací s omezujícími podmínkami ve tvaru nerovnic, tak úloha k ní duálně sdružená bude maximalizací s podmínkami ve tvaru opačně orientovaných nerovnic. Můžeme-li v takovém případě nazvat primární úlohu spodním odhadem řešení úlohy, je úloha k ní duálně sdružená horním odhadem řešení úlohy.

V našem případě budeme těžit z toho, že duální formulace povede na menší soustavu rovnic a tedy se zmenší i výpočetní náročnost úlohy. Navíc využijeme možnosti rozdělit původní problém na menší podproblémy, které je do jisté míry možné počítat odděleně, nebo dokonce paralelně.

**Definice 6.1.** ***Původní úloha***
*Nechť $\Omega$, označuje oblast jejíž vlastnosti jsou popsány maticí tuhosti $\mathbf{K}_g$ a která je v určitých místech zatížena vektorem zatížení $\boldsymbol{f}_g$. Pro tuto úlohu platí*

$$\mathbf{K}_g \boldsymbol{u}_g = \boldsymbol{f}_g \qquad \mathbf{B}_g \boldsymbol{u}_g = \boldsymbol{c}_g \tag{6.1}$$

*kde $\boldsymbol{u}_g$ značí vektor skutečných posunů úlohy a matice $\mathbf{B}_g$ je tak zvaná Booleova matice. Vektor $\boldsymbol{c}_g$ představuje vektor předepsaných posunů.*




**Obrázek 6.1:** Znázornění původní úlohy

Booleova matice obsahuje informaci o podepření jednotlivých stupňů volnosti. Je složena z řádkových vektorů, který každý přísluší jedné okrajové podmínce a obsahuje pouze nuly a jedničky. Má tedy přesně tolik řádků kolik je podepřených stupňů volnosti, v každém řádku je pouze jeden nenulový prvek, který přísluší podepřenému stupni.

Pokud tedy $\boldsymbol{u} \in \mathbb{R}^5$ a z toho $u_1 = 0$ a $u_4 = 0$ pak Booleova matice $\mathbf{B}$ bude mít tvar:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

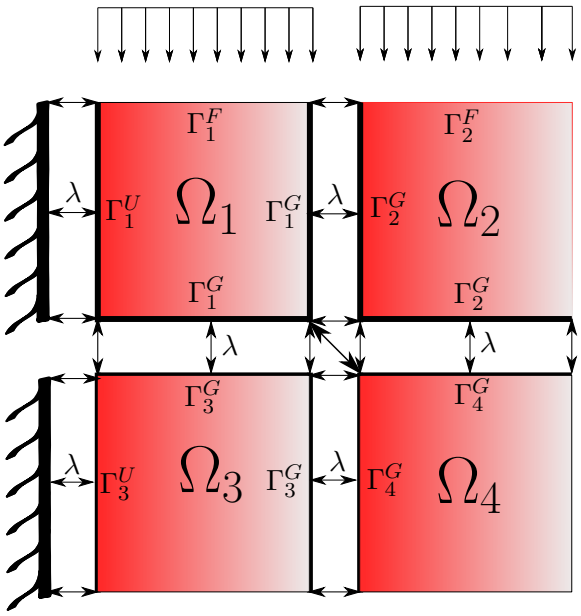
Původní úlohu můžeme vhodným rozdělením na podoblasti a zadefinováním mezidoménových sil $\boldsymbol{\lambda}$ převést na úlohu řešitelnou pomocí duální úlohy. Rozdělením na podoblasti zároveň dosáhneme snížení výpočetní náročnosti, protože každou část můžeme do jisté míry řešit jako samostatnou úlohu.

**Definice 6.2.** ***Původní úloha pro duální řešení***
*Nechť $\Omega$ je oblast úlohy, pak $\Omega_1, \Omega_2 \ldots \Omega_i$ jsou nepřekrývající se podoblasti původní úlohy.*

$$\Omega_1, \Omega_2, \ldots \Omega_i \subseteq \Omega \qquad \wedge \qquad \Omega_k \cap \Omega_l = \varnothing, \quad k \neq l, \tag{6.2}$$

*kde $k, l \leq i$*

*O podoblastech budeme hovořit jako o doménách.*




**Obrázek 6.2:** Úloha rozdělená na domény a připravená pro duální řešení

*Každá doména má svou matici tuhosti* $\mathbf{K}_1$,$\mathbf{K}_2$,$\ldots\mathbf{K}_i$, *která je čtvercová a pozitivně definitní,* $\mathbf{K}_j \in \mathbb{R}^{n\times n}$, *kde* $j \in \{1,2,\ldots i\}$, $n,i \in \mathbb{N}$.

*Každá* $\Omega_j$ *je podepřená pevnými podporami, nebo ostatními doménami tak, že se úloha nemůže chovat jako mechanismus. Zároveň každá doména podléhá předepsanému zatížení. Množinu okrajových podmínek, která z těchto skutečností vyplývá budeme značit* $\Gamma_j$. *Tato množina se sestává ze tří disjunktních částí* $\Gamma_j^F$,$\Gamma_j^U$,$\Gamma_j^G$

$$\Gamma_j = \Gamma_j^F \cup \Gamma_j^U \cup \Gamma_j^G \tag{6.3}$$

*kde* $\Gamma_j^F$ *je množina podmínek odpovídajících předepsaným zatížením na podoblasti* $\Omega_j$. *Hodnoty předepsaného zatížení jsou zastoupeny vektorem* $\boldsymbol{f}_j \in \mathbb{R}^n$. $\Gamma_j^U$ *je množina podmínek odpovídajících předepsaných přemístění. Obě tyto množiny vycházejí z mechanických vlastností původní úlohy.*

$\Gamma_j^G$ *je množina všech mezidoménových podmínek na styku sousedních domén.*

*Pro takto zadefinovanou úlohu pak platí :*

$$\mathbf{K}_j\boldsymbol{u}_j = \boldsymbol{f}_j \qquad \sum_{j=1}^{i} \mathbf{B}_j\boldsymbol{u}_j = \boldsymbol{c}_g \tag{6.4}$$

*kde* $\mathbf{B}_j$ *je Booleova matice obsahující informaci o* $\Gamma_j^U$ *a* $\Gamma_j^G$ *a* $\boldsymbol{c}_g$ *je vektor předepsaných posunů.*

Duální řešení úlohy se od standardního liší především tím, že nejprve hledá Lagrangeovy multiplikátory $\boldsymbol{\lambda}$ a z nich až následně dopočítává vektor posunů $\boldsymbol{u}$. Lagrangeovy multiplikátory fyzikálně představují síly na rozhraní

jednotlivých domén a v místech předepsaných posunů. Více o $\boldsymbol{\lambda}$ bude řečeno v následující kapitole.

Tvar matice $\mathbf{B}$ pro podmínky na $\Gamma^U$ jsme již popsali výše. Podmínky na hranicích domén $\Gamma^G$ vysvětlíme opět na konkrétním případě. Pokud naše úloha sestává ze dvou domén $\Omega_1$ a $\Omega_2$ a $\boldsymbol{u}_1$, $\boldsymbol{u}_2 \in \mathbb{R}^{3\times 1}$ a tedy $\boldsymbol{u}_g \in \mathbb{R}^{6\times 1}$ a současně se domény dotýkají ve 3. stupni volnosti na 1. doméně a 1. stupni volnosti na 2. doméně ( $u_{1,3}$=$u_{2,1}$). Pak příslušný vektor $\mathbf{B}$ sestává z 0,-1 a 1 a má tvar:

$$\begin{bmatrix} 0 & 0 & 1 & -1 & 0 & 0 \end{bmatrix}$$

Díky tomu, že jednotlivé domény nemají žádné společné body, jsou prvky matice tuhosti soustředěny kolem diagonály a tvoří takzvanou diagonálně blokovou strukturu. Proto lze vztah mezi doménovými maticemi a primární maticí zapsat příhodně jako

$$\mathbf{K}_g = \begin{bmatrix} \mathbf{K}_1 & . & . & . \\ . & \mathbf{K}_2 & & \\ . & & \dots & \\ . & & & \mathbf{K}_i \end{bmatrix} \tag{6.5}$$

Obdobně lze zapsat vztahy i pro matici $\mathbf{B}$ a vektory $\boldsymbol{f}$ a $\boldsymbol{u}$.

$$\mathbf{B}_g = \begin{bmatrix} \mathbf{B}_1 & \mathbf{B}_2 & \dots & \mathbf{B}_i \end{bmatrix}$$
$$\boldsymbol{f}_g = \begin{bmatrix} \boldsymbol{f}_1 & \boldsymbol{f}_2 & \dots & \boldsymbol{f}_i \end{bmatrix}^T \qquad \boldsymbol{u}_g = \begin{bmatrix} \boldsymbol{u}_1 & \boldsymbol{u}_2 & \dots & \boldsymbol{u}_i \end{bmatrix}^T \tag{6.6}$$

## 6.2 FETI metoda

Finite Element Tearing and Interconnecting Method (FETI) je duální metoda doménové dekompozice, kterou v našem případě budeme kombinovat s předpodmíněnou metodou sdružených gradientů. Je vhodná pro paralelní výpočty a je efektivní a praktická pro řešení rozměrných úloh.

### 6.2.1 Základní metoda FETI

Oproti definici 6.2 je podobnější klasickému ručnímu výpočtu, který nám je znám z učebních předmětů oboru mechaniky. Matice jsou sestavovány na podproblému, který vynechává stupně volnosti, které jsou podepřené vnějšími podporami. Z matice tuhosti jsou tedy vynechány řádky a sloupce příslušící $\Gamma^U$, které předepisují $u$=0. Odpovídajícím způsobem je upravena i matice $\mathbf{B}$ a $\mathbf{B}_j$.

Mějme tedy zadanou úlohu podle (6.4)

$$\mathbf{K}_j \boldsymbol{u}_j = \boldsymbol{f}_j \qquad \sum_{j=1}^{i} \mathbf{B}_j \boldsymbol{u}_j = \boldsymbol{c}_g, \tag{6.7}$$

pro kterou platí, že množina okrajových podmínek domény $\Omega_j$

$$\Gamma_j = \Gamma_j^F \cup \Gamma_j^G \tag{6.8}$$

kde $j \in \{1, 2 \ldots, i\}$, $n, i \in \mathbb{N}$.

Podle postupu odvozeného v kapitole 2.2, převedeme úlohu na kvadratickou formu vázanou soustavou rovnic $\boldsymbol{g}(\boldsymbol{x}_j)^T = \mathbf{0}$.

$$f(\boldsymbol{x}) = \frac{1}{2}\sum_{j=1}^{i} \boldsymbol{x}_j^T \mathbf{K}_j \boldsymbol{x}_j - \sum_{j=1}^{i} \boldsymbol{x}_j^T \boldsymbol{f}_j \qquad \boldsymbol{g}(\boldsymbol{x}_j)^T = \sum_{j=1}^{i} \mathbf{B}_j \boldsymbol{x}_j - \boldsymbol{c}_g = \mathbf{0} \tag{6.9}$$

Sumační tvar, který je jediným rozdílem oproti předchozím odvozením, je zde pouze důsledkem rozdělení úlohy na domény. Pokud se pozorněji zadíváme na výše uvedené matice, je zřejmé že funkce $f(\boldsymbol{x})$ ze vztahu (6.9) je totožná s funkcí v (2.5) pro původní úlohu. Zároveň pokud víme, že námi hledané řešení je minimem funkce $f(\boldsymbol{x})$, pak se jedná o primární formulaci optimalizační úlohy.

Jelikož hledáme extrém vázané funkce, můžeme, podle kapitoly 5, převést rovnice 6.9 na Lagrangeův funkcionál [DF01]

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}) = \frac{1}{2}\sum_{j=1}^{i} \boldsymbol{x}_j^T \mathbf{K}_j \boldsymbol{x}_j - \sum_{j=1}^{i} \boldsymbol{x}_j^T \boldsymbol{f}_j + \boldsymbol{\lambda}^T \sum_{j=1}^{i} \mathbf{B}_j \boldsymbol{x}_j - \boldsymbol{c}_g. \tag{6.10}$$

Jak už jsme uvedli výše, extrém Lagrangeova funkcionálu $\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda})$ odpovídá extrému funkce vázané v (6.9) a extrém funkce vázané je řešením úlohy (6.7). Vektor $\boldsymbol{u}$ tedy nalezneme jako extrém funkcionálu $\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda})$.

Bohužel tím, že jsme úlohu převedli na Lagrangeův funkcionál, jsme ztratili příhodný tvar kvadratické formy. Kdybychom vykreslili průběh funkcionálu 6.10, zjistili bychom, že má typický tvar sedlobodové úlohy.

V oblasti řešení sedlobodových úloh je dobře známo, že řešení úlohy nalezneme minimalizací podle $\boldsymbol{x}$ a maximalizací podle $\boldsymbol{\lambda}$ [BF91].

$$\mathcal{L}(\boldsymbol{u}, \overline{\boldsymbol{\lambda}}) = \min_{\boldsymbol{x}} \max_{\boldsymbol{\lambda}} \mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}) \tag{6.11}$$

V našem textu se spokojíme pouze s grafickým zobrazením jednoduché úlohy 6.3.

Je-li $\mathbf{K}_g$ symetrická a pozitivně definitní, potom $f(\boldsymbol{x})$ je konvexní funkcí a řešení úlohy je možné najít minimalizací podle $\boldsymbol{x}$.

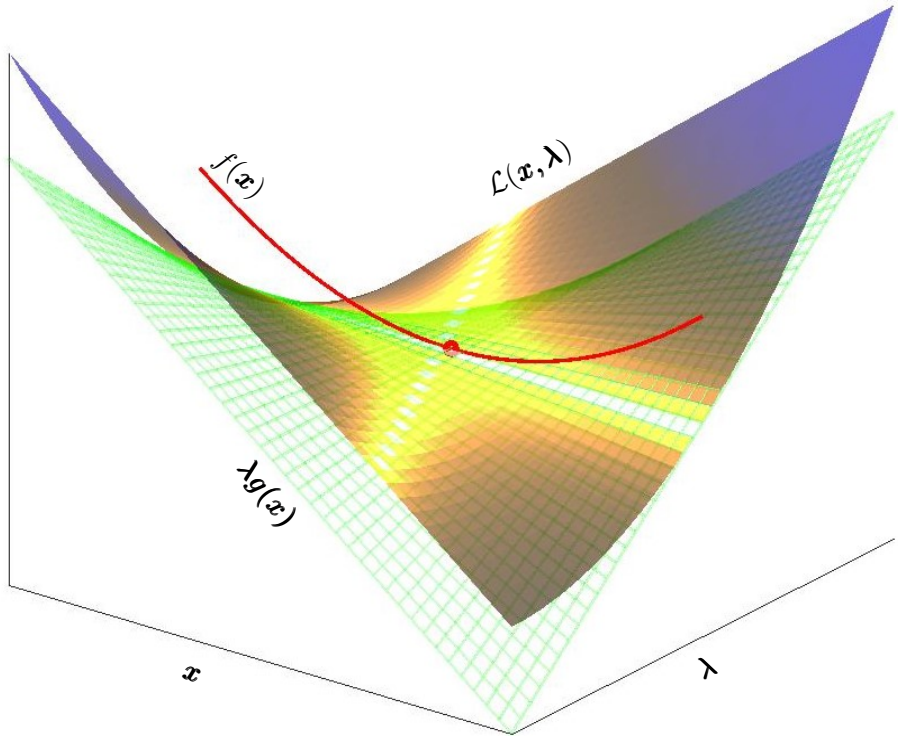
Změní-li se tvar této funkce přičtením $\boldsymbol{\lambda}^T \boldsymbol{g}(\boldsymbol{x}_j)^T$ na sedlobodovou úlohu, můžeme z tvaru úlohy vyvozovat, že řešení dosáhneme minimalizací podle $\boldsymbol{x}$ a maximalizací podle $\boldsymbol{\lambda}$.

Zavedeme si dva pomocné funkcionály

$$\mathcal{D}(\boldsymbol{\lambda}) = \min_{\boldsymbol{x}} \mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}) \tag{6.12}$$

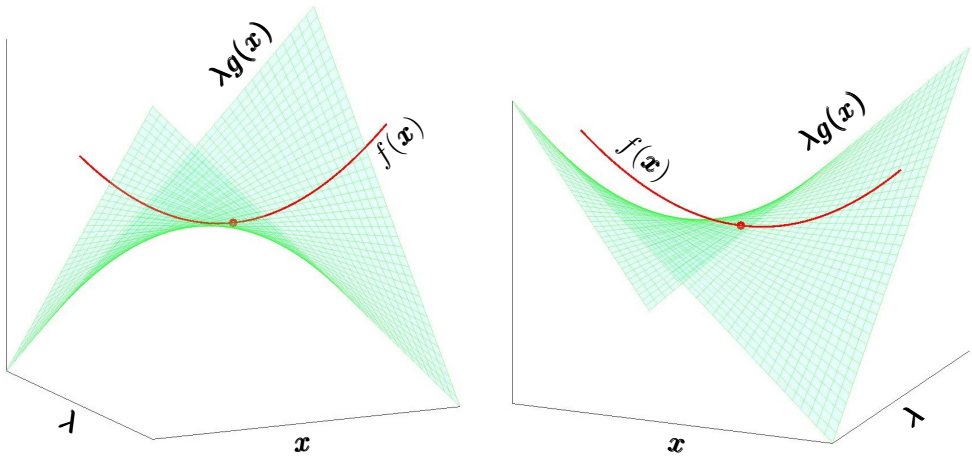
$$\mathcal{E}(\boldsymbol{x}) = \max_{\boldsymbol{\lambda}} \mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}), \tag{6.13}$$

pro které platí, že vektor $\boldsymbol{x}$ je přípustný právě tehdy, pokud $\mathcal{E}(\boldsymbol{x})$ je konečný a vektor $\boldsymbol{\lambda}$ je přípustný právě tehdy platí-li že $\mathcal{D}(\boldsymbol{\lambda})$ je konečný.




$\boldsymbol{x}, \boldsymbol{\lambda} \in \mathbb{R}^1$

**Obrázek 6.3:** Znázornění $\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda})$ a jeho rozklad na $f(\boldsymbol{x})$ a $\boldsymbol{\lambda}^T \boldsymbol{g}(\boldsymbol{x}^T)$




a) b)

**Obrázek 6.4:** Rozklad na $f(\boldsymbol{x})$ a $\boldsymbol{\lambda}^T \boldsymbol{g}(\boldsymbol{x})^T$

a) Pohled zdůrazňující konkávnost $\boldsymbol{\lambda}^T \boldsymbol{g}(\boldsymbol{x})^T$ podle $\boldsymbol{\lambda}$
b) Pohled znázorňující konvexnost $f(\boldsymbol{x})$ a $\boldsymbol{\lambda}^T \boldsymbol{g}(\boldsymbol{x})^T$ podle $\boldsymbol{x}$

$$\mathcal{L}(\boldsymbol{u}, \overline{\boldsymbol{\lambda}}) = \max_{\boldsymbol{\lambda}} \mathcal{D}(\boldsymbol{\lambda}) = \min_{\boldsymbol{x}} \mathcal{E}(\boldsymbol{x}) \tag{6.14}$$

Je tedy zřejmé, že pro řešení úlohy platí

$$\mathcal{D}(\overline{\boldsymbol{\lambda}}) = \mathcal{E}(\boldsymbol{u}). \tag{6.15}$$

V tomto bodě se musíme vrátit zpět k rovnici (6.10). Vytvoříme si parciální derivace $\mathcal{L}$ podle $\boldsymbol{x}$ a $\boldsymbol{\lambda}$.

$$\frac{\partial \mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda})}{\partial \boldsymbol{x}_j} = \sum_{j=1}^{i} \left( \frac{1}{2} (\boldsymbol{x}_j^T \mathbf{K}_j + \mathbf{K}_j \boldsymbol{x}_j) - \boldsymbol{f}_j + \boldsymbol{\lambda}^T \mathbf{B}_j \right) = \tag{6.16}$$

$$= \sum_{j=1}^{i} \left( \mathbf{K}_j \boldsymbol{x}_j - \boldsymbol{f}_j + \mathbf{B}_j^T \boldsymbol{\lambda} \right)$$

$$\frac{\partial \mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda})}{\partial \boldsymbol{\lambda}} = \sum_{j=1}^{i} (\mathbf{B}_j \boldsymbol{x}_j) - \boldsymbol{c}_g \tag{6.17}$$

Z toho je možné ukázat, že pokud [FMR94]

$$\mathcal{D}(\boldsymbol{\lambda}) > -\infty \tag{6.18}$$

$$\mathcal{E}(\boldsymbol{x}) < +\infty, \tag{6.19}$$

pak musí platit

$$\boldsymbol{f}_j - \mathbf{B}_j^T \boldsymbol{\lambda} \perp \ker \mathbf{K} \tag{6.20}$$

$$\sum_{j=1}^{i} (\mathbf{B}_j \boldsymbol{x}_j) - \boldsymbol{c}_g = \mathbf{0}. \tag{6.21}$$

Podmínka (6.20) je nezbytná pro existenci řešení na plovoucí doméně, tedy na doméně, která je držena pouze ostatními doménami a nikoliv vnějšími podepřeními. Rovnice 6.21 reprezentuje podmínky kontinuity deformací mezi stupni volnosti na hranici domén a mezi doménami a předepsanými posuny.

Značka „ker“ zde představuje tak zvané jádro matice $\mathbf{K}$. Jádro matice je vektorovým prostorem, jehož nenulové prvky splňují rovnici $\mathbf{K}\boldsymbol{u} = \mathbf{0}$.

Řešením $\boldsymbol{u}, \overline{\boldsymbol{\lambda}}$ našeho problému je inflexní bod na sedlové ploše, pro který musí platit

$$\left[ \frac{\partial \mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda})}{\partial \boldsymbol{x}_j} \right]_{\boldsymbol{u}, \overline{\boldsymbol{\lambda}}} = \mathbf{0} \tag{6.22}$$

$$\left[ \frac{\partial \mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda})}{\partial \boldsymbol{\lambda}} \right]_{\boldsymbol{u}, \overline{\boldsymbol{\lambda}}} = \mathbf{0} \tag{6.23}$$

Z toho vyplývá

$$\mathbf{K}_j \boldsymbol{u}_j + \mathbf{B}_j^T \overline{\boldsymbol{\lambda}} = \boldsymbol{f}_j \tag{6.24}$$

$$\sum_{j=1}^{i} (\mathbf{B}_j \boldsymbol{u}_j) = \boldsymbol{c}_g \tag{6.25}$$

Použijeme-li zápis z (6.5) a (6.6) lze tuto soustavu rovnic zapsat v maticovém formátu [BF91].

$$\begin{bmatrix} \mathbf{K}_g & \mathbf{B}_g^T \\ \mathbf{B}_g & \mathbf{O} \end{bmatrix} \begin{bmatrix} \boldsymbol{u}_g \\ \overline{\boldsymbol{\lambda}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{f}_g \\ \boldsymbol{c}_g \end{bmatrix} \tag{6.26}$$

kde $\mathbf{O}$ je nulová matice.

V této chvíli máme již vše, abychom mohli odvodit metodu FETI jako metodu sdružených gradientů pro maximalizaci funkcionálu $\mathcal{D}(\boldsymbol{\lambda})$. Fyzikálně tento postup představuje maximalizaci energetického funkcionálu.

Podmínku (6.20) lze ekvivalentně zapsat jako [KVM+13]

$$\mathbf{R}^T (\boldsymbol{f}_j - \mathbf{B}_j^T \boldsymbol{\lambda}) = \mathbf{0} \tag{6.27}$$

kde $\mathbf{R}$ je báze jádra matice $\mathbf{K}$.

Pro doménu, která nemá zabráněné žádné stupně volnosti, lze $\mathbf{R}_j$ odvodit explicitně z geometrie jednotlivých domén. V $\mathbb{R}^2$ je matice $\mathbf{R}_j$ každé domény tvořena třemi sloupcovými vektory skládanými z bloků, které jsou určeny pro každý bod $\boldsymbol{a}_{i,j} = [x_a, y_a]$. [KVM+13]

$$\mathbf{R}_{i,j} = \begin{bmatrix} -y_a & 1 & 0 \\ x_a & 0 & 1 \end{bmatrix} \qquad \mathbf{R}_j = \begin{bmatrix} \mathbf{R}_{1,j} \\ \mathbf{R}_{2,j} \\ \dots \\ \mathbf{R}_{m.j} \end{bmatrix} \tag{6.28}$$

$\mathbf{R}_{i,j} \in \mathbb{R}^3$ se skládá z šesti sloupcových vektorů pro každý bod $\boldsymbol{a}_{i,j} = [x_a, y_a, z_a]$.

$$\mathbf{R}_{i,j} = \begin{bmatrix} 0 & -z_a & y_a & 1 & 0 & 0 \\ z_a & 0 & -x_a & 0 & 1 & 0 \\ -y_a & x_a & 0 & 0 & 0 & 1 \end{bmatrix} \tag{6.29}$$

A vztah mezi maticemi jednotlivých domén je následující

$$\mathbf{R}_g = \begin{bmatrix} \mathbf{R}_1 & & & \\ & \mathbf{R}_2 & & \\ & & \dots & \\ & & & \mathbf{R}_n \end{bmatrix} \tag{6.30}$$

Z (6.24) můžeme vyjádřit vektor $\boldsymbol{u}_j$. Musíme si však uvědomit, že $\boldsymbol{u}_j$ se skládá ze dvou složek.

$$\boldsymbol{u}_j = \boldsymbol{u}_j^{\perp} + \mathbf{R}_j \alpha \tag{6.31}$$

kde pro druhou složku platí, že

$$\mathbf{K}_j \mathbf{R}_j \boldsymbol{\alpha}_j = \mathbf{0} \tag{6.32}$$

$\mathbf{R}_j \boldsymbol{\alpha}_j$ zde představuje posunutí celých domén, které nevyvodí silovou odezvu a tedy je nezávislé na $\boldsymbol{\lambda}$.

Protože $\mathbf{R}_j \boldsymbol{\alpha}_j$ po vynásobení maticí $\mathbf{K}_j$ zmizí, dostaneme po vyjádření z rovnice (6.24) pouze $\boldsymbol{u}_j^{\perp}$.

$$\boldsymbol{u}_j^{\perp} = \mathbf{K}_j^{\dagger} \left( \boldsymbol{f}_j - \mathbf{B}_j^T \overline{\boldsymbol{\lambda}} \right) \tag{6.33}$$

$$\boldsymbol{u}_j - \mathbf{R}_j \boldsymbol{\alpha}_j = \mathbf{K}_j^{\dagger} \left( \boldsymbol{f}_j - \mathbf{B}_j^T \overline{\boldsymbol{\lambda}} \right) \tag{6.34}$$

Z toho lze vyjádřit vektor $\boldsymbol{u}_j$ jako [KVM+13]

$$\boldsymbol{u}_j = \mathbf{K}_j^{\dagger} \left( \boldsymbol{f}_j - \mathbf{B}_j^T \overline{\boldsymbol{\lambda}} \right) + \mathbf{R}_j \overline{\boldsymbol{\alpha}_j} \tag{6.35}$$

pro konkrétní vektor $\overline{\boldsymbol{\alpha}_j} \in \mathbb{R}^l, l \in \mathbb{N}$, který přísluší řešení problému $\boldsymbol{u}_j$.

Matice $\mathbf{K}_j^{\dagger}$ je generalizovaná inverze matice $\mathbf{K}$, neboli takzvaná pseudoinveze. Pro domény, které nejsou plovoucí, platí $\mathbf{K}_j^{\dagger} = \mathbf{K}_j^{-1}$. V opačné případě musíme využít například Moore-Penroseovu pseudoinversi, protože matice $\mathbf{K}_j$ může být singulární.

Pokud takto zjištěné $\boldsymbol{u}_j$ dosadíme do rovnice(6.25) získáváme

$$-\sum_{j=1}^{i} \mathbf{B}_j \mathbf{K}_j^{\dagger} \mathbf{B}_j^T \overline{\boldsymbol{\lambda}} + \sum_{j=1}^{i} \mathbf{B}_j \mathbf{R}_g \overline{\boldsymbol{\alpha}}_g = \boldsymbol{c}_g - \sum_{j=1}^{i} \mathbf{B}_j \mathbf{K}_j^{\dagger} \boldsymbol{f}_j \tag{6.36}$$

Spojíme-li tuto rovnici s (6.27) dostáváme soustavu rovnic

$$\begin{bmatrix} -\sum_{j=1}^{i} \mathbf{B}_j \mathbf{K}_j^{\dagger} \mathbf{B}_j^T & \sum_{j=1}^{i} \mathbf{B}_j \mathbf{R}_g \\ -\mathbf{R}_j^T \mathbf{B}_j^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} \overline{\boldsymbol{\lambda}} \\ \overline{\boldsymbol{\alpha}}_g \end{bmatrix} = \begin{bmatrix} \boldsymbol{c}_g - \sum_{j=1}^{i} \mathbf{B}_j \mathbf{K}_j^{\dagger} \boldsymbol{f}_j \\ -\mathbf{R}_j^T \boldsymbol{f}_j \end{bmatrix} \tag{6.37}$$

V tuto chvíli je výhodné předefinovat problém do nového značení

**Definice 6.3.** ***Problém na rozhraních***
*Nechť existují matice* $\mathbf{F}, \mathbf{G}$ *a vektory* $\boldsymbol{h}, \boldsymbol{e}$ *tak, že platí*

$$\mathbf{F}_j = \mathbf{B}_j \mathbf{K}_j^{\dagger} \mathbf{B}_j^T \qquad \mathbf{F}_g = \sum_{j=1}^{i} \mathbf{B}_j \mathbf{K}_j^{\dagger} \mathbf{B}_j^T \tag{6.38}$$

$$\mathbf{G}_j = -\mathbf{R}_j^T \mathbf{B}_j^T \qquad \mathbf{G}_g = [\mathbf{G}_1, \mathbf{G}_2, \dots \mathbf{G}_i] \tag{6.39}$$

$$\boldsymbol{h}_g = \sum_{j=1}^{i} \left( \mathbf{B}_j \mathbf{K}_j^{\dagger} \boldsymbol{f}_j \right) - \boldsymbol{c}_g \tag{6.40}$$

$$\boldsymbol{e}_j = -\mathbf{R}_j^T \boldsymbol{f}_j \qquad \boldsymbol{e}_g = [\boldsymbol{e}_1^T, \boldsymbol{e}_2^T, \dots \boldsymbol{e}_i^T]^T \tag{6.41}$$

$$\begin{bmatrix} \mathbf{F}_g & \mathbf{G}_g^T \\ \mathbf{G}_g & \mathbf{0} \end{bmatrix} \begin{bmatrix} \overline{\boldsymbol{\lambda}} \\ \overline{\boldsymbol{\alpha}}_g \end{bmatrix} = \begin{bmatrix} \boldsymbol{h}_g \\ \boldsymbol{e}_g \end{bmatrix} \tag{6.42}$$

*kde* $\mathbf{K}_j \in \mathbb{R}^{n\times n}$ *je symetrická, kde* $j \in \langle 1; i\rangle$, $n, i \in \mathbb{N}$. $\mathbf{B}_j \in \mathbb{R}^{m\times n}$ *je Booleovská matice příslušící doméně* $j$, *kde* $m \in \mathbb{N}$ *je počet podmínek na množině* $\Gamma$. *To znamená, že matice* $\mathbf{B}_j$ *má stejný rozměr jako* $\mathbf{B}_g$, *ale jsou v ní zapsány pouze stupně volnosti příslušící* $\Omega_j$, *pro ostatní stupně volnosti je matice* $\mathbf{B}_j$ *nulová. Matice* $\mathbf{R}_j$ *představuje bázi jádra matice* $\mathbf{K}_j$. *Vektor* $\boldsymbol{c}_j \in \mathbb{R}^{m\times 1}$ *představuje vektor předepsaných posunů na doméně* $j$.

Tento tvar úlohy je duální formulací úlohy sestavenou na rozhraních jednotlivých domén a je s výhodami řešen iterativními postupy.

### 6.2.2 Řešení metody FETI pomocí metody PCPG

Metoda FETI je obvykle řešena metodou předpodmíněných, sdružených projektovaných gradientů.

Indefinitní problém 6.3 převedeme na semidefinitní rozdělením soustavy rovnic (6.42) a odstraněním podmínky rovnováhy $\mathbf{G}_g\boldsymbol{\lambda} = \boldsymbol{e}_g$

**Definice 6.4.** ***Rozdělení*** $\boldsymbol{\lambda}$*[DF01][KVM+13]*
*Nechť existuje námi hledaný vektor* $\boldsymbol{\lambda} \in \mathbb{R}^m, m \in \mathbb{N}$, *pro který platí*

$$\boldsymbol{\lambda} = \boldsymbol{\lambda}_0 + \boldsymbol{\lambda}_{ker} \tag{6.43}$$

*kde* $\boldsymbol{\lambda}_0 \in Im\mathbf{G}_g$ *je takzvané partikulární řešení* $\boldsymbol{\lambda}$ *vzniklé z rovnice* $\mathbf{G}_g\boldsymbol{\lambda} = \boldsymbol{e}_g$ *a* $\boldsymbol{\lambda}_{ker} \in \ker \mathbf{G}_g$.

Označení $\mathrm{Im}\mathbf{G}_g$ obraz matice $\mathbf{G}_g$. To představuje prostor vzniklý vynásobením $\mathbf{G}_g$ vektorem proměnných, tedy prostor všech možných výsledků součinu $\mathbf{G}_g\boldsymbol{x}$.

Vytvoříme si matici $\mathbf{P}_G$; projekční matici sestavenou z projektoru ortogonálního na prostor $\ker \mathbf{G}_g$.

Protože matice $\mathbf{B}_g$ má plnou řádkovou hodnost, tak matice $\mathbf{G}_g$ musí mít plnou řádkovou hodnost. Potom můžeme projekční matici $\mathbf{P}_G$ sestavit jako následující vztah. [KVM+13]

$$\mathbf{P}_G = I - \mathbf{G}_g^T(\mathbf{G}_g\mathbf{G}_g^T)^{-1}\mathbf{G}_g \tag{6.44}$$

kde $\mathbf{G}_g^T(\mathbf{G}_g\mathbf{G}_g^T)^{-1}\mathbf{G}_g$ je již zmíněný ortogonální projektor a pro matici $\mathbf{P}_G$ platí, že $\mathbf{P}_G\mathbf{G}_g^T = \mathbf{0}$.

Partikulární řešení $\boldsymbol{\lambda}_0$ dostaneme z druhé rovnice (6.42), po projekci do $\mathrm{Im}\mathbf{G}_g$, jako [KVM+13]

$$\boldsymbol{\lambda}_0 = \mathbf{G}_g^T(\mathbf{G}_g\mathbf{G}_g^T)^{-1}\boldsymbol{e}_g \tag{6.45}$$

Dosadíme-li novou rovnici (6.43) do první rovnice z výrazu (6.42) dostaneme rovnici

$$\mathbf{F}_g\boldsymbol{\lambda}_0 + \mathbf{F}_g\boldsymbol{\lambda}_{ker} + \mathbf{G}_g^T\boldsymbol{\alpha}_g = \boldsymbol{h}_g \tag{6.46}$$

$$\mathbf{F}_g\boldsymbol{\lambda}_{ker} = \boldsymbol{h}_g - \mathbf{F}_g\boldsymbol{\lambda}_0 - \mathbf{G}_g^T\boldsymbol{\alpha}_g \tag{6.47}$$

Potom obě strany rovnice vynásobíme projekční maticí $\mathbf{P}_G$ [HKKP07].

$$\mathbf{P}_G\mathbf{F}_g\boldsymbol{\lambda}_{ker} = \mathbf{P}_G(\boldsymbol{h}_g - \mathbf{F}_g\boldsymbol{\lambda}_0) \tag{6.48}$$

Tato rovnice je jednoznačně řešitelná.

Známe-li $\boldsymbol{\lambda}$, můžeme dopočítat $\boldsymbol{\alpha}_g$ tak, že první rovnici z (6.42) vynásobíme $\mathbf{G}_g$ zleva a vyřešíme pro $\boldsymbol{\alpha}_g$.

$$\mathbf{G}_g\mathbf{F}_g\boldsymbol{\lambda} + \mathbf{G}_g\mathbf{G}_g^T\boldsymbol{\alpha}_g = \mathbf{G}_g\boldsymbol{h}_g \tag{6.49}$$

$$\boldsymbol{\alpha}_g = (\mathbf{G}_g\mathbf{G}_g^T)^{-1}\mathbf{G}_g(\boldsymbol{h}_g - \mathbf{F}_g\boldsymbol{\lambda}) \tag{6.50}$$

Nakonec určíme jednoduchý symetrický předpodmiňovač $\overline{\mathbf{F}}^{-1}$. Předpodmiňovač modifikuje původní úlohu a tím urychluje nalezení řešení. Tato matice se v různých publikacích liší. V této části textu bude užit předpomiňovač podle [FMR94] a [KVM+13]

$$\overline{\mathbf{F}}^{-1} = \mathbf{B}_g\mathbf{K}_g\mathbf{B}_g^T \tag{6.51}$$

Nyní máme vše, pro iterační řešení metody FETI.

**Definice 6.5.** ***PCPG algoritmus pro řešení metody FETI*** *[KVM+13]*

*Nechť existuje úloha ve tvaru (6.7). Potom po převedení na problém 6.42 lze nalézt řešení iterativním procesem :*

$$\overline{\boldsymbol{\lambda}}_0 = \mathbf{G}_g^T(\mathbf{G}_g\mathbf{G}_g^T)^{-1}\boldsymbol{e}_g \tag{6.52}$$

$$\boldsymbol{\lambda}_{ker}^0 = \mathbf{G}_g^T(\mathbf{G}_g\mathbf{G}_g^T)^{-1}\boldsymbol{e}_g \tag{6.53}$$

$$\boldsymbol{d}^0 = \boldsymbol{r}^0 = \boldsymbol{h}_g - \mathbf{F}_g\boldsymbol{\lambda}_{ker}^0 \tag{6.54}$$

$$x_0 \tag{6.55}$$

***while*** $\|\boldsymbol{w}_i\| \leq \epsilon$

*Projekce* $$\boldsymbol{w}^i = \mathbf{P}_G\boldsymbol{r}^i \tag{6.56}$$

*Předpodmínění* $$\boldsymbol{z}^i = \overline{\mathbf{F}}^{-1}\boldsymbol{w}^i \tag{6.57}$$

*Projekce* $$\boldsymbol{y}^i = \mathbf{P}_G\boldsymbol{z}^i \tag{6.58}$$

$$\beta^i = \begin{cases} \frac{(\boldsymbol{y}^i)^T\boldsymbol{w}^i}{(\boldsymbol{y}^{i-1})^T\boldsymbol{w}^{i-1}} & i > 0 \\ 0 & i = 0 \end{cases} \tag{6.59}$$

$$\boldsymbol{p}^i = \begin{cases} \boldsymbol{y}^i + \beta^i\boldsymbol{p}^{i-1} & i > 0 \\ \boldsymbol{y}^0 & i = 0 \end{cases} \tag{6.60}$$

$$\alpha_i = \frac{(\boldsymbol{y}^i)^T\boldsymbol{w}^i}{(\boldsymbol{p}^i)^T\mathbf{F}_g\boldsymbol{p}^i} \tag{6.61}$$

$$\boldsymbol{\lambda}_{ker}^{i+1} = \boldsymbol{\lambda}_{ker}^i + \alpha^i\boldsymbol{p}^i \tag{6.62}$$

$$\boldsymbol{r}^{i+1} = \boldsymbol{r}^i - \alpha^i\mathbf{F}_g\boldsymbol{p}^i \tag{6.63}$$

***end***

$$\overline{\boldsymbol{\lambda}}_{ker} = \boldsymbol{\lambda}_{ker}^{m+1} \tag{6.64}$$

$$\overline{\boldsymbol{\lambda}}_g = \overline{\boldsymbol{\lambda}}_0 + \overline{\boldsymbol{\lambda}}_{ker} \tag{6.65}$$

$$\overline{\boldsymbol{\alpha}}_g = (\mathbf{G}_g\mathbf{G}_g^T)^{-1}\mathbf{G}_g(\boldsymbol{h}_g - \mathbf{F}_g\overline{\boldsymbol{\lambda}}_g) \tag{6.66}$$

$$\boldsymbol{u}_g = \mathbf{K}_g^\dagger\left(\boldsymbol{f}_g - \mathbf{B}_g^T\overline{\boldsymbol{\lambda}}_g + \mathbf{R}_g\overline{\boldsymbol{\alpha}}_g\right) \tag{6.67}$$

*Tolerance $\epsilon$ je námi volená hodnota, vybíraná s ohledem na typ úlohy a požadovanou přesnost řešení.*

# Kapitola 7

# Kontaktní úlohy

Za kontaktní úlohy se označují soustavy rovnic omezené podmínkami ve formě rovnic i nerovnic. Fyzikálně si můžeme takovou úlohu představovat jako soustavu domén, které na sebe v určitých místech přímo doléhají, tak jak to známe z metody FETI, a v určitých místech jsou mezi nimi mezery. Pak platí, že ve chvíli, kdy deformace domén přesáhnou velikost mezery, dojde mezi doménami ke kontaktu a v daném místě se začne přenášet síla.

Taková úloha může mít i opačný tvar. Tedy že vlivem velkých vnitřních sil se konstrukce začne rozpojovat a tedy v některých místech se síla přestane přenášet.

**Definice 7.1.** ***Kontaktní úloha***
*Nechť existuje symetrická matice $\mathbf{K}_g \in \mathbb{R}^{n\times n}$, která je pozitivně definitní na prostoru omezeném podmínkami. Dále existuje vektor deformací $\boldsymbol{u}_g$ a vektor zatížení $\boldsymbol{f}_g \in \mathbb{R}^n$. Dohromady tvoří soustavu rovnic*

$$\mathbf{K}_g \boldsymbol{u}_g = \boldsymbol{f}_g, \tag{7.1}$$

*která je omezená soustavou rovnic*

$$\mathbf{B}_g \boldsymbol{u}_g = \boldsymbol{c}_g^U \;\Leftrightarrow\; \boldsymbol{b}_k^T \boldsymbol{u}_g = c_k \quad k \in \Gamma^U \cup \Gamma^G \tag{7.2}$$

*a nerovnic*

$$\mathbf{C}_g \boldsymbol{u}_g \geq \boldsymbol{c}_g^C \;\Leftrightarrow\; \boldsymbol{b}_k^T \boldsymbol{u}_g \geq c_k \quad k \in \Gamma^C, \tag{7.3}$$

*kde $\mathbf{B}_g \in \{-1,0,1\}^{m\times n}$ a $\mathbf{C}_g \in \{-1,0,1\}^{l\times n}$ jsou Booleovské matice představují soustavy rovnic předepsaných deformací úlohy, $\boldsymbol{c}_g^U$ je vektor předepsaných posunů v podporách a na hranicích domén a $\boldsymbol{c}_g^C$ je vektor předepsaných přetvoření potřebných pro přenášení síly v místě kontaktu. $m,n,l \in \mathbb{N}$*

*$\boldsymbol{b}_k$ je vektor skládající se pouze z 0,1,-1, který představuje konkrétní podmínku označenou $k$. $k \in \Gamma \subset \mathbb{N}$.*

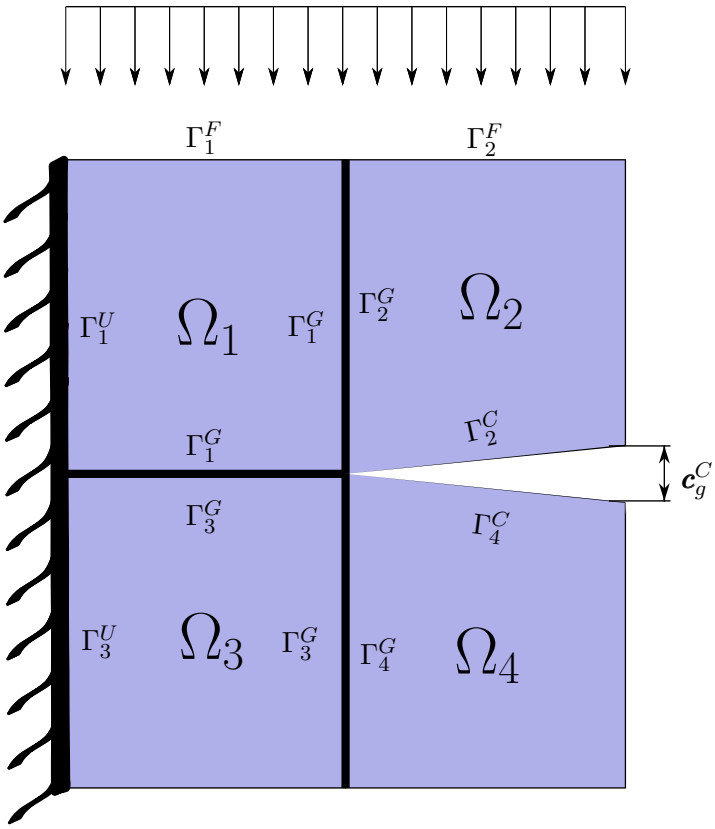
*Po rozdělení úlohy $\Omega$ na domény $\Omega_1, \Omega_2, \ldots \Omega_i$, kde $i \in \mathbb{N}$, lze množinu vázacích podmínek jedné domény $\Gamma_j$ rozdělit na čtyři podmnožiny.*

$$\Gamma_j = \Gamma_j^F \cup \Gamma_j^U \cup \Gamma_j^G \cup \Gamma_j^C \tag{7.4}$$

*kde $\Gamma_j^F$ je množina předepsaných zatížení a je zastoupena vektorem $\boldsymbol{f}_j \in \mathbb{R}^{n_j}$. $\Gamma_j^U$ je množina předepsaných přemístění. $\Gamma_j^G$ je množina všech mezidoménových podmínek na styku sousedních domén a $\Gamma_j^C$ je množina všech kontaktních podmínek. $n_j \in \mathbb{N}, j \in \{1,2,\ldots i\}$.*

*Pro množinu podmínek celé úlohy platí*

$$\Gamma = \bigcup_{j=1}^{i} \Gamma_j \quad \Gamma^F = \bigcup_{j=1}^{i} \Gamma_j^F \quad \Gamma^U = \bigcup_{j=1}^{i} \Gamma_j^U$$
$$\Gamma^G = \bigcup_{j=1}^{i} \Gamma_j^G \quad \Gamma^C = \bigcup_{j=1}^{i} \Gamma_j^C \tag{7.5}$$




**Obrázek 7.1:** Vyobrazení kontaktní úlohy

Kontaktními podmínkami se zde rozumí podmínky zadané v podobě nerovnic.

Hlavním úskalím kontaktních úloh je skutečnost, že nevíme, které kontaktní podmínky budou při daném zatížení platné, tedy aktivní. Navíc vnitřní síly a deformace se budou měnit v závislosti na tom, kolik podmínek bude aktivních a kolik ne. Taktéž množství aktivních podmínek záleží na tom, jak se konstrukce zdeformuje a na kterých místech bude přenášena síla.

Je tedy zřejmé, že kontaktní úlohy vedou vždy na iterativní proces.

## 7.1 Metoda aktivní množiny

Metoda aktivní množiny může být primární, duální, nebo primárně-duální. Zabývá se řešením konvexních nebo konkávních kvadratických problémů vázaných podmínkami ve tvaru nerovnic.

V tomto textu se zaměříme pouze na primární metodu a stejně jako v předchozích částech se budeme zabývat pouze konvexními úlohami. Z toho vyplývá, že $\mathbf{K}_g$ musí být pozitivně definitní na oblasti omezené podmínkami.

V této kapitole se budeme věnovat metodě aktivní množiny bez využití dělení na domén, proto musíme množinu vázacích podmínek upravit následujícím způsobem.

$$\Gamma = \Gamma^F \cup \Gamma^U \cup \Gamma^C \tag{7.6}$$

Nejprve úlohu převedeme na kvadratický problém postupem vysvětleným v kapitole 2.2 a převedením nerovnic na rovnice.

**Definice 7.2.** ***kvadratická kontaktní úloha****[NW06]*

$$\boldsymbol{u}_g = \arg\min_{x_g} f(\boldsymbol{x}_g) = \arg\min_{x_g} \left( \frac{1}{2} \boldsymbol{x}_g^T \mathbf{K}_g \boldsymbol{x}_g - \boldsymbol{x}_g^T \boldsymbol{f}_g \right) \tag{7.7}$$

$$\widehat{\mathbf{B}}_g \boldsymbol{u}_g = \widehat{\boldsymbol{c}}_g \;\Leftrightarrow\; \boldsymbol{b}_k^T \boldsymbol{u}_g = c_k, \quad k \in \mathcal{A}(\boldsymbol{u}_g) \tag{7.8}$$

$$\boldsymbol{b}_k^T \boldsymbol{u}_g \geq c_k, \quad k \in \Gamma^C \setminus \mathcal{A}(\boldsymbol{u}_g) \tag{7.9}$$

*kde rovnice (7.8) představuje soustavu všech aktivních omezujících podmínek v místě optima $\boldsymbol{u}_g$.*

*Množina $\mathcal{A}$ je nazývána aktivní množinou a představuje všechny kontaktní a stálé podmínky, které jsou při daném zatížení a skutečných deformacích aktivní.*

$$\mathcal{A}(\boldsymbol{u}_g) \subset \Gamma \tag{7.10}$$

Zásadním problémem je, že neznáme správnou aktivní množinu $\mathcal{A}$. Nalezení správných aktivních podmínek je tedy podstatou metody aktivní množiny.

To, jestli jsou podmínky aktivní či nikoliv, je závislé na tom, zda se v místě podmínky přenáší síla. V kapitole 6.2 jsme se zmínili, že Lagrangeovy multiplikátory představují sílu přenášenou v místě podmínky. Proto je pro nás v tuto chvíli výhodné zadefinovat i Lagrangeův funkcionál.

**Definice 7.3.** ***Lagrangeův funkcionál pro kontaktní úlohu****[NW06]*

$$\mathcal{L}(\boldsymbol{x}_g, \boldsymbol{\lambda}) = \frac{1}{2} \boldsymbol{x}_g^T \mathbf{K}_g \boldsymbol{x}_g - \boldsymbol{x}_g^T \boldsymbol{f}_g - \sum_{k \in \Gamma^U \cup \Gamma^C} \lambda_k (\boldsymbol{b}_k^T \boldsymbol{x}_g - c_k) \tag{7.11}$$

Jak už jsme odvodili v kapitole 6.2 pro takto zadefinovaný Lagrangeův funkcionál musí v místě optima $\boldsymbol{u}_g$ platit následující podmínky.[NW06]

$$\boldsymbol{b}_k^T \boldsymbol{u}_g = c_k, \quad k \in \mathcal{A}(\boldsymbol{u}_g) \tag{7.12}$$

$$\boldsymbol{b}_k^T \boldsymbol{u}_g \geq c_k, \quad k \in \Gamma^C \setminus \mathcal{A}(\boldsymbol{u}_g) \tag{7.13}$$

$$\lambda_k \geq 0, \quad k \in \Gamma^C \cap \mathcal{A}(\boldsymbol{u}_g) \tag{7.14}$$

$$\mathbf{K}_g \boldsymbol{x}_g - \boldsymbol{f}_g - \sum_{k \in \mathcal{A}(\boldsymbol{u}_g)} (\lambda_k \boldsymbol{b}_k) = 0 \tag{7.15}$$

Rovnice (7.15) je pouze přepisem (6.24) a rovnice (7.14) vyplývá z faktu, že v místech kontaktu musí být tlaková síla, jinak podmínka nemůže být aktivní. Z mínusu v (7.11) je zřejmé, že zde máme tlakovou sílu zadefinovánu kladně. Toto je drobná změna oproti předchozím definicím.

Metoda aktivní množiny je iterativní proces, který hledá krok, z jedné iterace do druhé, vyřešením kvadratického podproblému, který je omezen všemi podmínkami zadanými ve tvaru rovnic a některými podmínkami zadanými ve tvaru nerovnic převedenými na rovnice.

Podprostor obsahující podmínky ve tvaru nerovnic převedené do tvaru rovnic v iteraci $i$ budeme nazývat pracovní množinou $\mathcal{W}^i$. Důležitou podmínkou je, že pro všechna $k \in \mathcal{W}^i$ musí být všechny řádky $\widehat{\mathbf{B}}_g$ lineárně nezávislé a to i v případě, že by tomu tak nebylo ve stavu $\mathcal{W}^i = \Gamma^C$.[DF01]

Prvním krokem metody aktivní množiny je výběr přípustného počátečního bodu $\boldsymbol{x}_g^0$ a počáteční pracovní množiny $\mathcal{W}^0$. Je více způsobů jak nalézt přípustný počáteční bod, například přístup „Phase I“, nebo „Big M“ [NW06].

V tomto textu se těmito přístupy nebudeme zabývat. Tyto metody budou s výhodami užívány ve chvíli, kdy $\Gamma^C$ není složeno pouze z lineárně nezávislých podmínek a tedy je obtížné určit počáteční bod, který by odpovídal všem podmínkám.

V našem případě tedy počáteční bod $\boldsymbol{x}_g^0$ vhodně zvolíme tak, aby neporušoval žádnou z podmínek a následně pro něj odvodíme počáteční pracovní set $\mathcal{W}^0$.

Pro každou iteraci nejdříve ověříme, zda bod $\boldsymbol{x}_g^i$ minimalizuje funkci $f(\boldsymbol{x}_g)$ na podprostoru $\mathcal{W}^i$. Pokud tomu tak není, vypočteme $\boldsymbol{p}$, neboli následující směr výpočetního kroku, vyřešením kvadratického podproblému, ve kterém jsou všechny podmínky příslušící $\mathcal{W}^i$ brány jako rovnice a ostatní podmínky z $\Gamma^C$ jsou zanedbány.

$$\boldsymbol{p} = \boldsymbol{x}_g - \boldsymbol{x}_g^i \tag{7.16}$$

Pokud takto zadefinovaný krok dosadíme do rovnic (7.7) a (7.8) dostaneme kvadratický vztah

$$\begin{aligned} f(\boldsymbol{x}_g) = f(\boldsymbol{x}_g^i + \boldsymbol{p}) = \frac{1}{2}(\boldsymbol{x}_g^i + \boldsymbol{p})^T \mathbf{K}_g (\boldsymbol{x}_g^i + \boldsymbol{p}) - (\boldsymbol{x}_g^i + \boldsymbol{p})^T \boldsymbol{f}_g = \\ = \frac{1}{2}\left((\boldsymbol{x}_g^i)^T \mathbf{K}_g \boldsymbol{x}_g^i + (\boldsymbol{x}_g^i)^T \mathbf{K}_g \boldsymbol{p} + \boldsymbol{p}^T \mathbf{K}_g \boldsymbol{x}_g^i + \boldsymbol{p}^T \mathbf{K}_g \boldsymbol{p}\right) - (\boldsymbol{x}_g^i)^T \boldsymbol{f}_g - \boldsymbol{p}^T \boldsymbol{f}_g \end{aligned} \tag{7.17}$$

Protože matice $\mathbf{K}_g$ je symetrická a vektor $\boldsymbol{x}_g^i$ je předem daná konstanta, lze rovnici přepsat jako

$$\begin{aligned} f(\boldsymbol{x}_g^i + \boldsymbol{p}) &= \frac{1}{2}\boldsymbol{p}^T\mathbf{K}_g\boldsymbol{p} + (\boldsymbol{x}_g^i)^T\mathbf{K}_g^T\boldsymbol{p} - \boldsymbol{f}_g^T\boldsymbol{p} + \frac{1}{2}(\boldsymbol{x}_g^i)^T\mathbf{K}_g\boldsymbol{x}_g^i - \boldsymbol{f}_g^T\boldsymbol{x}_g^i = \\ &= \frac{1}{2}\boldsymbol{p}^T\mathbf{K}_g\boldsymbol{p} + (\mathbf{K}_g\boldsymbol{x}_g^i - \boldsymbol{f}_g)^T\boldsymbol{p} + f(\boldsymbol{x}_g^i) \end{aligned} \tag{7.18}$$

Protože $\boldsymbol{x}_g^i$ je konstantní musí i $f(\boldsymbol{x}_g^i)$ být konstantní. Proto můžeme nalézt směr výpočetního kroku v iteraci $i$ jako řešení kvadratického podproblému [NW06]

$$\boldsymbol{p}^i = \arg\min_{\boldsymbol{p}\in\mathcal{P}} \left(\frac{1}{2}\boldsymbol{p}^T\mathbf{K}_g\boldsymbol{p} + (\mathbf{K}_g\boldsymbol{x}_g^i - \boldsymbol{f}_g)^T\boldsymbol{p}\right) \tag{7.19}$$

$$\mathcal{P} = \{\boldsymbol{p} \in \mathbb{R}^n, \widehat{\mathbf{B}}_g\boldsymbol{p} = \mathbf{0}, \forall\, \boldsymbol{b}_k^C, c_k^C;\ k \in \mathcal{W}^i\} \tag{7.20}$$

Omezující podmínka (7.20) vyplývá z rovnice (7.16) a z předpokladu, že pokud platí rovnice (7.8), musí platit i tvrzení $\widehat{\mathbf{B}}_g\boldsymbol{x}_g^i = \widehat{\boldsymbol{c}}_g$.

Z této skutečnosti vyplývá i tvrzení, že pokud je $\boldsymbol{x}_g^i$ přípustný pro pracovní set $\mathcal{W}^i$, musí být pro $\mathcal{W}^i$ přípustný i následující bod iterace

$$\boldsymbol{x}_g^{i+1} = \boldsymbol{x}_g^i + \alpha^i\boldsymbol{p}^i \tag{7.21}$$

kde $\alpha^i$ je stejně jako v metodě sdružených gradientů délka kroku $\boldsymbol{p}^i$.

Koeficient $\alpha^i$ odpovídá největší možné délce kroku na intervalu $\langle 0;1\rangle$, pro kterou jsou splněny všechny podmínky z $\mathcal{W}^i$.

Pro případ, kdy nejsou všechny podmínky splněny, bez ohledu na délku $\alpha^i$ (tedy $k \notin \mathcal{W}^i$), a v případě že $\alpha^i \geq 0$ a $\widehat{\mathbf{B}}_g\boldsymbol{p}^i \leq \mathbf{0}$ musí pro bod $\boldsymbol{x}_g^{i+1}$ platit

$$\widehat{\mathbf{B}}_g\boldsymbol{x}_g^{i+1} \geq \widehat{\boldsymbol{c}}_g \tag{7.22}$$

$$\widehat{\mathbf{B}}_g(\boldsymbol{x}_g^i + \alpha^i\boldsymbol{p}^i) \geq \widehat{\mathbf{B}}_g\boldsymbol{x}_g^i \geq \widehat{\boldsymbol{c}}_g \tag{7.23}$$

Z toho lze vyjádřit $\alpha^i$.[NW06]

$$\alpha^i\widehat{\mathbf{B}}_g\boldsymbol{p}^i \geq \widehat{\boldsymbol{c}}_g - \widehat{\mathbf{B}}_g\boldsymbol{x}_g^i \tag{7.24}$$

$$\alpha^i \leq \frac{\widehat{\boldsymbol{c}}_g - \widehat{\mathbf{B}}_g\boldsymbol{x}_g^i}{\widehat{\mathbf{B}}_g\boldsymbol{p}^i} \tag{7.25}$$

Pro skalár $\alpha^i$ jsme tedy nalezly sadu podmínek, z nichž rozhodne ta nejnižší.

Z výše uvedených vztahů tedy můžeme délku kroku zadefinovat jako

**Definice 7.4.** ***Délka kroku*** $\alpha^i$

$$\overline{\alpha}^i = \min\left[1, \min_{k\notin\mathcal{W}^i, \boldsymbol{b}_k^T\boldsymbol{p}^i<0} \frac{\boldsymbol{c}_k - \boldsymbol{b}_k^T\boldsymbol{x}_g^i}{\boldsymbol{b}_k^T\boldsymbol{p}^i}\right] \qquad [NW06] \tag{7.26}$$

$$\alpha^i = \max[0, \overline{\alpha}^i] \tag{7.27}$$

V případě, že je $\alpha^i = 1$, jsou všechny podmínky splněny a aktuální pracovní množinu $\mathcal{W}^i$ není potřeba měnit. V opačném případě je $\alpha^i$ pozměněno takzvanou blokovací podmínkou. To znamená, že existuje podmínka, která není součástí $\mathcal{W}^k$ i přes to, že je aktivní.

Na základě tohoto zjištění je upravena pracovní množina zahrnutím jedné blokovací podmínky do $\mathcal{W}^i$

V tomto duchu je iterováno dokud nenalezneme bod $\widehat{\boldsymbol{x}}_g$, který je minimem funkce $f(\boldsymbol{x}_g)$ na aktuální pracovní množině $\widehat{\mathcal{W}}$. V takovém bodě musí být $\boldsymbol{p} = \mathbf{0}$.

Máme-li předpokládané řešení $\widehat{\boldsymbol{x}}_g$, musíme ověřit zda jsou splněny podmínky kompatibility. Toho dosáhneme tak, že rovnici (7.15) budeme brát jako splněnou a pro ni ověříme předpoklad (7.14).

Z rovnice (7.15) vyjádříme $\widehat{\boldsymbol{\lambda}}_g$ vyřešením soustavy rovnic

$$\widehat{\mathbf{B}_g}\widehat{\boldsymbol{\lambda}}_g = \mathbf{K}_g\widehat{\boldsymbol{x}}_g - \boldsymbol{f}_g \tag{7.28}$$

kde $\widehat{\boldsymbol{\lambda}}_g$ je vektor složený ze všech $\widehat{\lambda}_k, k \in \widehat{\mathcal{W}}$.

Poté ověříme, že všechna $\widehat{\lambda}_k, k \in \Gamma^C \cap \widehat{\mathcal{W}}$ jsou kladná. Pokud tomu tak je, podmínky kompatibility jsou splněny a vektor $\widehat{\boldsymbol{x}}_g$ je řešením $\boldsymbol{u}_g$ původního problému 7.2.

Pokud je však některé z $\widehat{\lambda}_k$ záporné, pak některá z podmínek není splněna. Musíme tedy příslušné indexy $k$, pro které je $\widehat{\lambda}_k$ na kontaktních podmínkách náležících pracovní množině záporné, odstranit z $\widehat{\mathcal{W}}$ a takto získaný bod a pracovní množinu použít jako zadání nové úlohy.

Pro takto vzniklou úlohu opět provedeme iterativní proces, který byl popsán výše a to opakujeme dokud nenalezneme stav, kdy budou všechna $\widehat{\lambda}_k$ kladná.

**Definice 7.5.** ***algoritmus metody aktivního množiny****[NW06]*

*Nalezení přípustného* $\boldsymbol{x}_g^0$ *a následné odvození* $\mathcal{W}^0$

***for*** $i = 0, 1, 2 \ldots$

$$\boldsymbol{p}^i = \arg\min_{p \in \mathcal{P}} \left( \frac{1}{2} \boldsymbol{p}^T \mathbf{K}_g \boldsymbol{p} + (\mathbf{K}_g \boldsymbol{x}_g^i - \boldsymbol{f}_g)^T \boldsymbol{p} \right) \tag{7.29}$$

***if*** $\boldsymbol{p}^i = 0$

$$\widehat{\mathcal{W}}^i = \mathcal{W}^i \tag{7.30}$$

$$\widehat{\mathbf{B}}_g \widehat{\boldsymbol{\lambda}}_g^i = \mathbf{K}_g \widehat{\boldsymbol{x}}_g^i - \boldsymbol{f}_g \rightarrow \widehat{\boldsymbol{\lambda}}_g^i \tag{7.31}$$

***if*** $\widehat{\lambda}_k^i \geq 0, k \in \Gamma^C \cap \widehat{\mathcal{W}}^i$

$$\boldsymbol{u}_g = \boldsymbol{x}_g^i$$

***stop***

***else***

*Nalezni* $k, k \in \Gamma^C \cap \widehat{\mathcal{W}}^i$,

*kterému odpovídá nejnižší* $\lambda_k^i$ *z* $\boldsymbol{\lambda}_g^i$

$$\boldsymbol{x}_g^{i+1} = \boldsymbol{x}_g^i \tag{7.32}$$

$$\mathcal{W}^{i+1} = \mathcal{W}^i \setminus \{k\} \tag{7.33}$$

***else*** $\boldsymbol{p}^i \neq 0$

$$\overline{\alpha}^i = \min \left[ 1, \min_{k \notin \mathcal{W}^i, \boldsymbol{b}_k^T \boldsymbol{p}^i < 0} \frac{\boldsymbol{c}_k - \boldsymbol{b}_k^T \boldsymbol{x}_g^i}{\boldsymbol{b}_k^T \boldsymbol{p}^i} \right] \tag{7.34}$$

$$\alpha^i = \max[0, \overline{\alpha}^i] \tag{7.35}$$

$$\boldsymbol{x}_g^{i+1} = \boldsymbol{x}_g^i + \alpha^i \boldsymbol{p}^i \tag{7.36}$$

***if*** $\alpha^i = 1$

$$\mathcal{W}^{i+1} = \mathcal{W}^i \tag{7.37}$$

***else***

*Nalezni* $k$, *pro které*

$$\alpha^i \geq \frac{\boldsymbol{c}_k - \boldsymbol{b}_k^T \boldsymbol{x}_g^i}{\boldsymbol{b}_k^T \boldsymbol{p}^i} \tag{7.38}$$

$$\mathcal{W}^{i+1} = \mathcal{W}^i \cup \{k\} \tag{7.39}$$

***end***

# Kapitola 8

# Metoda FETI-C

Tato kapitola se bude věnovat využití metody FETI pro řešení kontaktních úloh.

Metoda FETI-C je ve své podstatě propojením metody FETI, představené v kapitole 6.2, a metody aktivní množiny, popsané v kapitole 7.1. Toto spojení je výhodné pro řešení rozměrných úloh, protože využívá výhody FETI, tedy dělení na domény a zmenšení výpočetní náročnosti, a možnosti řešení kontaktních úloh. Navíc v FETI-C se snažíme ověřit platnost aktivních podmínek již v průběhu řešení kvadratického podproblému, nikoliv až po jeho vyřešení, jak to dělá metoda aktivní množiny. Toto je umožněno díky iterativnímu způsobu hledání Lagrangeových multiplikátoru.

V předchozích kapitolách jsme již v definici 7.1 zadefinovali úlohu, kterou se budeme s malou úpravou zabývat. Tou úpravou je rozdělení úlohy na domény podle metody FETI.

**Definice 8.1.** ***úloha FETI-C***

$$\mathbf{K}_j \boldsymbol{u}_j = \boldsymbol{f}_j \tag{8.1}$$

$$\sum_{j=1}^{i} (\mathbf{B}_j \boldsymbol{u}_j) = \sum_{j=1}^{i} \boldsymbol{c}_j^U \;\Leftrightarrow\; \sum_{j=1}^{i} \boldsymbol{b}_{k,j}^T \boldsymbol{u}_j = c_k \quad k \in \Gamma_j^U \cup \Gamma_j^G \tag{8.2}$$

$$\sum_{j=1}^{i} (\mathbf{C}_j \boldsymbol{u}_j) \geq \sum_{j=1}^{i} \boldsymbol{c}_j^C \;\Leftrightarrow\; \sum_{j=1}^{i} \boldsymbol{b}_{k,j}^T \boldsymbol{u}_j \geq c_k \quad k \in \Gamma_j^C \tag{8.3}$$

*kde* $j \in \{1, 2 \ldots, i\}$, $i \in \mathbb{N}$

Význam $\Gamma_j$ byl vysvětlen v definici 7.1. Vztahy pro převod zápisu z celkové úlohy na rozdělení na domény jsou uvedeny v (6.5) a (6.6).

Po vzoru (7.2) převedeme úlohu na kvadratický podproblém a ten následně převedeme na Lagrangeův funkcionál, stejně jako v rovnici (7.11).

I když zde uvedeme zápis v sumačním tvaru pro jednotlivé domény, je zřejmé, že tyto vztahy se dají ekvivalentně zapsat pomocí vektorů a matic celého problému.

**Definice 8.2.** ***Lagrangeuv funkcionál pro FETI-C***

$$\mathcal{L}(\boldsymbol{x}_j, \boldsymbol{\lambda}_k) = \sum_{j=1}^{i} \left( \frac{1}{2} \boldsymbol{x}_j^T \mathbf{K}_j \boldsymbol{x}_j - \boldsymbol{x}_j^T \boldsymbol{f}_j \right) - \sum_{k \in \Gamma^U \cup \Gamma^C} \lambda_k \left( \sum_{j=1}^{i} (\boldsymbol{b}_{k,j}^T \boldsymbol{x}_j) - c_k \right) \tag{8.4}$$

*dále platí*

$$\sum_{j=1}^{i} \boldsymbol{b}_{k,j}^T \boldsymbol{u}_j = c_k, \quad k \in \mathcal{A}(\boldsymbol{u}_g) \tag{8.5}$$

$$\sum_{j=1}^{i} \boldsymbol{b}_{k,j}^T \boldsymbol{u}_j \geq c_k, \quad k \in \Gamma^C \setminus \mathcal{A}(\boldsymbol{u}_g) \tag{8.6}$$

$$\lambda_k \geq 0, \quad k \in \Gamma^C \cap \mathcal{A}(\boldsymbol{u}_g) \tag{8.7}$$

$$(\mathbf{K}_j \boldsymbol{x}_j - \boldsymbol{f}_j) - \sum_{k \in \mathcal{A}(\boldsymbol{u}_g)} (\lambda_k \boldsymbol{b}_{k,j}) = 0 \tag{8.8}$$

Vektor $\boldsymbol{b}_{k,j}$ zde představuje část vektoru $\boldsymbol{b}_k$, která přísluší všem stupňům volnosti náležejícím doméně $\Omega_j$.

Stejně jako v metodě FETI lze i zde definici 8.2 převést na úlohu na rozhraních

**Definice 8.3.** ***Problém na rozhraních pro FETI-C***

$$\widehat{\mathbf{F}}_j = \mathbf{B}_j \mathbf{K}_j^\dagger \widehat{\mathbf{B}}_j^T \qquad \widehat{\mathbf{F}}_g = \sum_{j=1}^{i} \mathbf{B}_j \mathbf{K}_j^\dagger \widehat{\mathbf{B}}_j^T \tag{8.9}$$

$$\widehat{\mathbf{G}}_j = -\mathbf{R}_j^T \widehat{\mathbf{B}}_j^T \qquad \widehat{\mathbf{G}}_g = [\widehat{\mathbf{G}}_1, \widehat{\mathbf{G}}_2, \dots \widehat{\mathbf{G}}_i] \tag{8.10}$$

$$\boldsymbol{h}_j = \mathbf{B}_j \mathbf{K}_j^\dagger \boldsymbol{f}_j - \boldsymbol{c}_j \qquad \boldsymbol{h}_g = \sum_{j=1}^{i} \left( \mathbf{B}_j \mathbf{K}_j^\dagger \boldsymbol{f}_j - \boldsymbol{c}_j \right) \tag{8.11}$$

$$\boldsymbol{e}_j = -\mathbf{R}_j^T \boldsymbol{f}_j \qquad \boldsymbol{e}_g = [\boldsymbol{e}_1^T, \boldsymbol{e}_2^T, \dots \boldsymbol{e}_i^T]^T \tag{8.12}$$

$$\begin{bmatrix} \widehat{\mathbf{F}}_g & \widehat{\mathbf{G}}_g^T \\ \widehat{\mathbf{G}}_g & \mathbf{0} \end{bmatrix} \begin{bmatrix} \overline{\boldsymbol{\lambda}}_g \\ \overline{\boldsymbol{\alpha}}_g \end{bmatrix} = \begin{bmatrix} \boldsymbol{h}_g \\ \boldsymbol{e}_g \end{bmatrix} \tag{8.13}$$

$\widehat{\mathbf{B}}_j \in \mathbb{R}^{\widehat{m} \times n}$ *je Booleovská matice příslušící doméně* $j$*, kde* $\widehat{m} \in \mathbb{N}$ *je součet všech stálých a aktivních kontaktních podmínek na množině* $\Gamma_j$*. Matice* $\widehat{\mathbf{B}}_j$ *je tedy složena ze všech vektorů* $\boldsymbol{b}_{k,j}$ *příslušících aktivnímu setu* ($k \in \mathcal{A}(\boldsymbol{u}_g)$) *omezených na stupně volnosti příslušící doméně* $\Omega_j$*.*

Je zřejmé že výraz (8.13) je řešením funkcionálu ve tvaru [DF01]

$$\mathcal{S}(\overline{\boldsymbol{\lambda}}_g, \overline{\boldsymbol{\alpha}}_g) = \max \left( \frac{1}{2} \boldsymbol{\lambda}_g^T \widehat{\mathbf{F}}_g \boldsymbol{\lambda}_g - \boldsymbol{\lambda}_g^T \boldsymbol{h}_g - \boldsymbol{\alpha}_g^T (\widehat{\mathbf{G}}_g^T \boldsymbol{\lambda}_g - \boldsymbol{e}_g) \right), \tag{8.14}$$

který představuje duální formulaci k maximalizaci rovnice( 8.2).

Podívejme se v tuto chvíli, jak je v metodě FETI zadefinován krok z jedné iterace do druhé (6.54),(6.56) a rozepišme jej pomocí definice (8.3) a rovnic (6.45) a (6.46).

$$\begin{aligned} \boldsymbol{w}^i = \mathbf{P}_G \boldsymbol{r}^i = \mathbf{P}_G(\boldsymbol{h}_g - \widehat{\mathbf{F}}_g \boldsymbol{\lambda}_g^i + \widehat{\mathbf{G}}_g^T \boldsymbol{\alpha}_g^i) = \\ = \mathbf{P}_G \left( \sum_{j=1}^{i} \left( \mathbf{B}_j \mathbf{K}_j^\dagger \boldsymbol{f}_j \right) - \boldsymbol{c}_g - \sum_{j=1}^{i} \mathbf{B}_j \mathbf{K}_j^\dagger \widehat{\mathbf{B}}_j^T \boldsymbol{\lambda}_g^i + \widehat{\mathbf{G}}_g^T \boldsymbol{\alpha}_g^i \right) = \\ = \mathbf{P}_G \left( \sum_{j=1}^{i} \left( \mathbf{B}_j \mathbf{K}_j^\dagger \boldsymbol{f}_j - \mathbf{B}_j \mathbf{K}_j^\dagger \widehat{\mathbf{B}}_j^T \boldsymbol{\lambda}_g^i \right) + \widehat{\mathbf{G}}_g^T \boldsymbol{\alpha}_g^i - \boldsymbol{c}_g \right) \end{aligned} \tag{8.15}$$

Z tohoto výrazu lze odvodit, že na správné aktivní množině $\mathcal{A}$ bude pro každé kladné $\lambda_k^i > 0$ vyhledávací směr $w_k^i$ rovný nule a pro každé $\lambda_k^i = 0$ musí $w_k$ být záporné. Pokud bude $\lambda_k^i = 0$, pak to znamená, že nedochází ke kontaktu a protažení prutů $\sum_{j=1}^{i} \left( \mathbf{B}_j \mathbf{K}_j^\dagger \boldsymbol{f}_j \right)$ musí být menší než protažení předepsané pro vznik kontaktu $\boldsymbol{c}_g$. A zároveň pokud bude $\lambda_k^i > 0$, pak musí protažení prutů $\sum_{j=1}^{i} \left( \mathbf{B}_j \mathbf{K}_j^\dagger \boldsymbol{f}_j \right)$ bez protažení které působící silovou odezvu v místech podmínek $\sum_{j=1}^{i} \left( \mathbf{B}_j \mathbf{K}_j^\dagger \widehat{\mathbf{B}}_j^T \boldsymbol{\lambda}_g^i \right)$ být rovné $\boldsymbol{c}_g$.

Proto lze kontaktní podmínky přepsat jako [DF01]

$$\lambda_k \geq 0 \tag{8.16}$$

$$w_k \leq 0 \tag{8.17}$$

$$\lambda_k w_k = 0 \tag{8.18}$$

pro $k \in \Gamma^C \cap \mathcal{A}$.

Podmínky v tomto tvaru pro nás budou výhodné pro ověření, zda je podmínka součástí aktivní setu již v průběhu iterace.

Když máme takto zadefinovánu řešenou úlohu, můžeme ji řešit pomocí metody FETI-C.

Nejprve určíme počáteční přípustný bod, ze kterého budeme pokračovat metodou sdružených gradientů. Protože ještě nevíme, jak bude vypadat výsledná aktivní množina, určíme $\boldsymbol{\lambda}_0$ třeba pro případ, ve kterém jsou všechny kontaktní podmínky aktivní. V takovém případě $\widehat{\mathbf{G}} = \mathbf{G}$.[DF01]

$$\mathcal{W}^0 = \Gamma^C \cup \Gamma^U \cup \Gamma^G \tag{8.19}$$

$$\boldsymbol{\lambda}^0 = \mathbf{G}_g^T (\mathbf{G}_g \mathbf{G}_g^T)^{-1} \boldsymbol{e}_g \tag{8.20}$$

Následně je třeba ověřit, zda jsou splněny všechny kontaktní podmínky. Zavedeme tedy tak zvanou plánovací proceduru, která je sama o sobě iterativní a kterou budeme značit $\mathcal{P}(\boldsymbol{\lambda})$.

Nejprve ověříme, zda na všech kontaktních podmínkách je $\boldsymbol{\lambda}^l$ kladný, případně upravíme $\boldsymbol{\lambda}^l$ tak, aby splňoval rovnici (8.16).

$$\widetilde{\lambda}_k^l = \begin{cases} \lambda_k^l & k \in \Gamma^U \cup \Gamma^G \\ \lambda_k^l & k \in \Gamma^C \wedge \lambda_k^l > 0 \\ 0 & k \in \Gamma^C \wedge \lambda_k^l \leq 0 \end{cases} \tag{8.21}$$

Na základě těchto úprav zároveň poopravíme i aktuální pracovní set $\widehat{W}^l$.

$$k \in \widehat{\mathcal{W}}^{l+1} \quad \text{pokud } \widetilde{\lambda}_k^l > 0 \tag{8.22}$$

$$k \notin \widehat{\mathcal{W}}^{l+1} \quad \text{pokud } \widetilde{\lambda}_k^l = 0 \tag{8.23}$$

Z toho důvodu musí být opraveno i $\widehat{\mathbf{B}}$ a $\widehat{\mathbf{G}}$.

Protože jsme upravily vektor $\boldsymbol{\lambda}^l$ pouze na základě dodržení kontaktních podmínek, musíme ověřit, zda jsou síly, které tento vektor představuje, v rovnováze.

Ověříme tedy, zda platí druhý řádek z rovnice (8.13). Pokud tato rovnice platí, je plánovací krok u konce. Pokud tomu tak není, musíme opět upravit $\boldsymbol{\lambda}^{l+1}$ tak, aby splňoval podmínky rovnováhy.

Nejprve si vytvoříme ortogonální projektor podle vzoru (6.44)

$$\mathbf{P}_G = I - \widehat{\mathbf{G}}_g^T (\widehat{\mathbf{G}}_g \widehat{\mathbf{G}}_g^T)^{-1} \mathbf{G}_g \tag{8.24}$$

A poté zajistíme vlastní rovnováhu vektoru $\boldsymbol{\lambda}^l$ jako

$$\boldsymbol{\lambda}^{l+1} = \mathbf{P}_G \widetilde{\boldsymbol{\lambda}}^l + \widehat{\mathbf{G}}_g^T (\widehat{\mathbf{G}}_g \widehat{\mathbf{G}}_g^T)^{-1} \boldsymbol{e}_g \tag{8.25}$$

kde $\widehat{\mathbf{G}}_g^T (\widehat{\mathbf{G}}_g \widehat{\mathbf{G}}_g^T)^{-1} \boldsymbol{e}_g$ je ve své podstatě partikulární řešení sestavené na aktuální pracovní množině. Z aktuálního řešení tedy odstraníme všechny složky náležející $\ker \widehat{\mathbf{G}}_g$ a následně je opět přičteme tak, aby splňovaly podmínky rovnováhy. Tento krok je zde nutný, protože nemůžeme vektor $\boldsymbol{\lambda}$ rozdělit na dvě složky, tak jak jsme to udělali v metodě FETI, právě proto, že neznáme skutečnou aktivní množinu.

Poté opět ověříme, zda některé $\lambda_k^{l+1}$ náležící kontaktním podmínkám není záporné. Pokud tomu tak není, ukončíme plánovací proceduru. Pokud je $\lambda_k^{l+1} \in \Gamma^C < 0$, pak se vrátíme opět na začátek plánovací procedury a celý proces zopakujeme.

**Definice 8.4.** ***Plánovací procedura*** [*DF01*]

$$
\begin{aligned}
&\mathcal{P}(\boldsymbol{\lambda}) = \\
&\quad \textbf{\textit{for}}\ l = 1, 2, \ldots \\
&\qquad \widetilde{\lambda}_k^l = \lambda_k^l \quad pro\ \{k \in \Gamma^U \cup \Gamma^G\} \ \vee\ \{k \in \Gamma^C \wedge \lambda_k^l > 0\} \\
&\qquad\qquad \widetilde{\lambda}_k^l = 0 \quad pro\ k \in \Gamma^C \wedge \lambda_k^l \leq 0 \\
&\qquad\qquad k \in \widehat{\mathcal{W}}^{l+1} \quad pokud\ \widetilde{\lambda}_k^l > 0 \\
&\qquad\qquad k \notin \widehat{\mathcal{W}}^{l+1} \quad pokud\ \widetilde{\lambda}_k^l = 0 \\
&\qquad\qquad\qquad \Rightarrow \widehat{\mathbf{B}}_g, \widehat{\mathbf{G}}_g \\
&\qquad \textbf{\textit{if}}\ \widehat{\mathbf{G}}_g \widetilde{\boldsymbol{\lambda}}_k^l = \boldsymbol{e} \\
&\qquad\qquad\qquad\qquad stop \\
&\qquad \textbf{\textit{else}} \\
&\qquad\qquad \mathbf{P}_G = I - \widehat{\mathbf{G}}_g^T (\widehat{\mathbf{G}}_g \widehat{\mathbf{G}}_g^T)^{-1} \mathbf{G}_g \\
&\qquad\qquad \boldsymbol{\lambda}^{l+1} = \mathbf{P}_G \widetilde{\boldsymbol{\lambda}}^l + \widehat{\mathbf{G}}_g^T (\widehat{\mathbf{G}}_g \widehat{\mathbf{G}}_g^T)^{-1} \boldsymbol{e}_g \\
&\qquad \textbf{\textit{end}} \\
&\qquad \textbf{\textit{if}}\ \boldsymbol{\lambda}^{l+1} \geq 0\ \forall k \in \Gamma^C \\
&\qquad\qquad\qquad\qquad stop \\
&\qquad \textbf{\textit{end}} \\
&\quad \textbf{\textit{end}}
\end{aligned}
\tag{8.26}
$$

Máme-li spočteno správné $\boldsymbol{\lambda}^0$ a správnou počáteční pracovní množinu, pak máme vše potřebné pro metodu FETI. Zapíšeme si správný ortogonální projektor a poté vypočteme reziduum.

Následně budeme pokračovat stejně jako v definici 6.5, jen s několika drobnými úpravami.

Především po vypočtení kroku $\boldsymbol{w}^i$ ověříme, zda vektor vyhovuje podmínce (8.17), a pokud tomu tak nebude, vektor $\boldsymbol{w}^i$ upravíme.

Po vypočtení $\boldsymbol{\lambda}^i$ v jedné iteraci podrobíme vektor $\boldsymbol{\lambda}^i$ plánovací proceduře. Pokud se nám v průběhu ověřování kontaktních podmínek pro vektor $\boldsymbol{\lambda}^i$ změní pracovní množina, nemůžeme následně reziduum pro následující iteraci spočíst jako v metodě sdružených gradientů, ale musíme reziduum přepočítat. Důvodem je změna matice $\widehat{\mathbf{F}}_g$

A poslední změnou bude využití efektivnějšího předpodmiňovače [DF01]. Předpodmiňovač je s výhodami sestavován na úrovni domén, zde ho však pro jednoduchost zápisu zadefinujeme v globálním systému.

$$
\widehat{\overline{\mathbf{F}}}^{-1} = \widehat{\widehat{\mathbf{B}}}_g \begin{bmatrix} \mathbf{S}_{bb} & \mathbf{0}_2 \\ \mathbf{0}_2^T & \mathbf{0}_1 \end{bmatrix} \widehat{\widehat{\mathbf{B}}}_g^T, \tag{8.27}
$$

kde $\mathbf{0}_1$ a $\mathbf{0}_2$ jsou nulové bloky doplňující $\mathbf{S}_{bb}$ tak aby celá matice měla stejnou velikost jako matice $\mathbf{K}_g$. $\mathbf{S}_{bb}$ je Schurův doplněk matice $\mathbf{K}_g$ a matice

$\hat{\hat{\mathbf{B}}}_g$ booleovská matice sestavená na všech stálých podmínkách a na všech kontaktních podmínkách, na kterých je $\boldsymbol{w}^i$ nenulové.

Schurův doplněk je kondenzace matice $\mathbf{K}_g$ na vybrané stupně volnosti. V našem případě bude Schůrův doplněk spočítán pouze jednou a to pro případ, kdy jsou všechny podmínky aktivní. Tedy $k \in \Gamma^C \cup \Gamma^G \cup \Gamma^U$. Princip Schurova rozkladu spočívá v tom, že kdyby nám byly známé všechny stupně volnosti, kterým přísluší některé podmínky, tak pokud je úloha dostatečně podmíněná, můžeme z těchto stupňů volnosti vypočítat všechny ostatní. Tedy můžeme tyto stupně volnosti vyjádřit v závislosti na jiných stupních volnosti.

Pokud bychom seřadili matici $\mathbf{K_g}$, tak aby všechny stupně volnosti, kterým přísluší nějaká podmínka, byly napravo, měla by matice tuhosti tvar [Hú14]

$$\mathbf{K}_g = \begin{bmatrix} \mathbf{K}_{aa} & \mathbf{K}_{ab} \\ \mathbf{K}_{ab}^T & \mathbf{K}_{bb} \end{bmatrix}, \tag{8.28}$$

kde $\mathbf{K}_{aa} \in \mathbb{R}^{a \times a}$ je singulární část matice, $\mathbf{K}_{bb} \in \mathbb{R}^{b \times b}$ je část příslušící všem stupňům volnosti, kterým přísluší nějaká podmínka a je tedy regulární a tedy invertovatelná. $\mathbf{K}_{ab} \in \mathbb{R}^{a \times b}$ jsou obdélníkové prvky, jejichž symetrie vyplývá ze symetrie původní matice.

Pokud tedy rozepíšeme nově vzniklou soustavu rovnic $\mathbf{K}_g \boldsymbol{u}_g = \boldsymbol{f}_g$ dostaneme

$$\mathbf{K}_{aa}\boldsymbol{u}_a + \mathbf{K}_{ab}\boldsymbol{u}_b = \boldsymbol{f}_a \tag{8.29}$$

$$\mathbf{K}_{ab}^T\boldsymbol{u}_a + \mathbf{K}_{bb}\boldsymbol{u}_b = \boldsymbol{f}_b \tag{8.30}$$

poté provedeme eliminaci tak, že vyjádříme $\boldsymbol{u}_b$

$$\boldsymbol{u}_b = \mathbf{K}_{bb}^{-1}\boldsymbol{f}_b - \mathbf{K}_{bb}^{-1}\mathbf{K}_{ab}^T\boldsymbol{u}_a \tag{8.31}$$

a dosadíme do první rovnice

$$\mathbf{K}_{aa}\boldsymbol{u}_a + \mathbf{K}_{ab}(\mathbf{K}_{bb}^{-1}\boldsymbol{f}_b - \mathbf{K}_{bb}^{-1}\mathbf{K}_{ab}^T\boldsymbol{u}_a) = \boldsymbol{f}_a \tag{8.32}$$

$$(\mathbf{K}_{aa} - \mathbf{K}_{ab}\mathbf{K}_{bb}^{-1}\mathbf{K}_{ab}^T)\boldsymbol{u}_a = \boldsymbol{f}_a - \mathbf{K}_{ab}\mathbf{K}_{bb}^{-1}\boldsymbol{f}_b \tag{8.33}$$

což lze zapsat jako

$$\mathbf{S}_{bb}\boldsymbol{u}_a = \boldsymbol{f}_{ab} \tag{8.34}$$

kde [Hú14]

$$\mathbf{S}_{bb} = \mathbf{K}_{aa} - \mathbf{K}_{ab}\mathbf{K}_{bb}^{-1}\mathbf{K}_{ab}^T \tag{8.35}$$

Schurův doplněk se využívá jako přímá metoda řešení úlohy. Zde ho sice využíváme jako předpodmiňovač, ale je důležité si uvědomit, že pokud seřadíme prvky matice $\mathbf{K}$ tak, aby vyhovovala formě v jaké jsme odvodili Schurův doplněk, musíme v této formě psát i všechny ostatní matice. Pokud nebudeme tyto matice tak řadit, musíme $\mathbf{S}_{bb}$ rozložit v předpodmiňovači tak, aby příslušné prvky odpovídaly příslušným stupňům volnosti.

Dále je nutné si uvědomit, že pokud takto řadíme sloupce matice $\mathbf{K}$ musíme zároveň přesunout i příslušné řádky, jinak bychom nedosáhli blokové struktury, která je pro Schurův doplněk potřeba.

V praxi se běžně matice nepřeřazuje, ale využívá se lokalizace.

Nyní máme předpodmiňovač, který na rozdíl od předchozího nevnáší do vektoru směru z jedné iterace do druhé složky, které by příslušely neaktivním podmínkám. Správná matice $\overline{\widehat{\mathbf{F}}}^{-1}$ je velmi důležitá, protože má velký vliv na omezení oscilací algoritmu a na směr, ve kterém hledáme další bod iterace. Chyba v předpomiňovači velmi často vede k selhání algoritmu.

V tuto chvíli máme vše potřebné pro řešení metodou FETI-C. V předchozích metodách jsme užívali jako ukončovací kritérium algoritmu velikost směrového vektoru $\boldsymbol{r}, \boldsymbol{d}$, nebo $\boldsymbol{w}^i$. Ani v tomto případě není důvod tuto podmínku změnit.

Algoritmus zastavíme například v případě, kdy velikost směrového vektoru $\|\widehat{\boldsymbol{w}}\|^i$ bude menší než výraz [DF01]

$$\epsilon \left( \left\| \widehat{\mathbf{B}}_g^{i+1} \mathbf{K}_g^{\dagger} \boldsymbol{f}_g \right\| - \|\boldsymbol{c}_j\| \right) \tag{8.36}$$

kde $\epsilon$ je námi zvolená malá hodnota, která reprezentuje požadovanou přesnost výpočtu, a zbytek výrazu zohledňuje velikost a typ úlohy tak, aby byly zohledněny případné numerické chyby.

Je zřejmé, že velikost ukončovacího kritéria ovlivňuje dobu a přesnost výpočtu.

**Definice 8.5.** ***PCPG algoritmus pro řešení metody FETI-C*** *[DF01]*

*Nechť existuje úloha ve tvaru 7.1. Potom po převedení úlohy do tvaru 8.3 lze nalézt řešení iterativním procesem :*

$$\mathcal{W}^0 = \Gamma^C \cup \Gamma^U \cup \Gamma^G \tag{8.37}$$

$$\boldsymbol{\lambda}^0 = \mathbf{G}_g^T(\mathbf{G}_g\mathbf{G}_g^T)^{-1}\boldsymbol{e}_g \tag{8.38}$$

$$\widehat{\boldsymbol{\lambda}}^0 = \mathcal{P}(\overline{\boldsymbol{\lambda}}^0) \rightarrow \widehat{\mathbf{B}}_g^0, \widehat{\mathbf{G}}_g^0, \mathbf{P}_G^0, \widehat{\mathcal{W}}^0 \tag{8.39}$$

$$\boldsymbol{r}^0 = \boldsymbol{h}_g - \widehat{\mathbf{F}}_g^0\widehat{\boldsymbol{\lambda}}^0 \tag{8.40}$$

$$\boldsymbol{while}\left\|\widehat{\boldsymbol{w}}^i\right\| = \epsilon\left(\left\|\widehat{\mathbf{B}}_g^{i+1}\mathbf{K}_g^\dagger\boldsymbol{f}_g\right\| - \|\boldsymbol{c}_j\|\right)$$

$$\textit{Projekce} \qquad \boldsymbol{w}^i = \mathbf{P}_G^i\boldsymbol{r}^i \tag{8.41}$$

$$\widehat{w}_k^i = 0 \qquad k \in \Gamma^C \setminus \widehat{\mathcal{W}}^i \wedge w_i^k \geq 0 \tag{8.42}$$

$$\widehat{w}_k^i = w_i^k \qquad k \in \widehat{\mathcal{W}}^i \tag{8.43}$$

$$\textit{Předpodmínění} \qquad \boldsymbol{z}^i = \overline{\widehat{\mathbf{F}}}^{-1}\widehat{\boldsymbol{w}}^i \tag{8.44}$$

$$\textit{Projekce} \qquad \boldsymbol{y}^i = \mathbf{P}_G^i\boldsymbol{z}^i \tag{8.45}$$

$$\beta^i = \frac{(\boldsymbol{y}^i)^T\widehat{\boldsymbol{w}}^i}{(\boldsymbol{y}^{i-1})^T\widehat{\boldsymbol{w}}^{i-1}} \qquad \textit{pro} \qquad i = 0, \beta^0 = 0 \tag{8.46}$$

$$\boldsymbol{p}^i = \boldsymbol{y}^i + \beta^i\boldsymbol{p}^{i-1} \qquad \textit{pro} \qquad i = 0, \boldsymbol{p}^0 = \boldsymbol{y}^0 \tag{8.47}$$

$$\alpha^i = \frac{(\boldsymbol{y}^i)^T\widehat{\boldsymbol{w}}^i}{(\boldsymbol{p}^i)^T\widehat{\mathbf{F}}_g\boldsymbol{p}^i} \tag{8.48}$$

$$\boldsymbol{\lambda}^{i+1} = \widehat{\boldsymbol{\lambda}}^i + \alpha^i\boldsymbol{p}^i \tag{8.49}$$

$$\widehat{\boldsymbol{\lambda}}^{i+1} = \mathcal{P}(\boldsymbol{\lambda}^{i+1}) \rightarrow \widehat{\mathbf{B}}_g^{i+1}, \widehat{\mathbf{G}}_g^{i+1}, \mathbf{P}_G^{i+1}, \widehat{\mathcal{W}}^{i+1} \tag{8.50}$$

$$\boldsymbol{if} \quad \widehat{\mathcal{W}}^{i+1} = \widehat{\mathcal{W}}^i \tag{8.51}$$

$$\boldsymbol{r}^{i+1} = \boldsymbol{r}^i - \alpha^i\widehat{\mathbf{F}}_g^{i+1}\boldsymbol{p}^i \tag{8.52}$$

$$\boldsymbol{else} \tag{8.53}$$

$$\boldsymbol{r}^{i+1} = \boldsymbol{h}_g - \widehat{\mathbf{F}}_g^{i+1}\boldsymbol{\lambda}^{i+1} \tag{8.54}$$

$$\boldsymbol{end} \tag{8.55}$$

$$\boldsymbol{end}$$

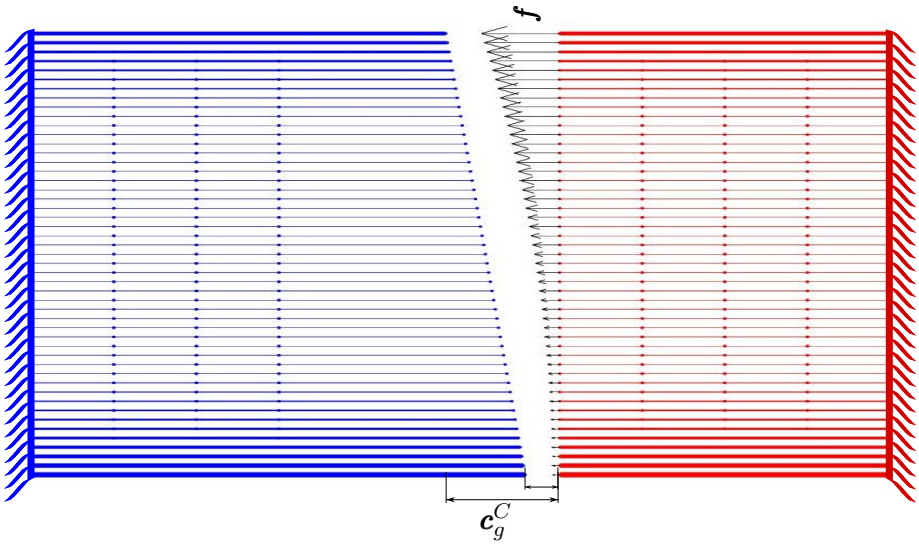
$$\overline{\boldsymbol{\lambda}} = \widehat{\boldsymbol{\lambda}}^i \tag{8.56}$$

$$\overline{\boldsymbol{\alpha}}_g = (\widehat{\mathbf{G}}_g\widehat{\mathbf{G}}_g^T)^{-1}\widehat{\mathbf{G}}_g(\boldsymbol{h}_g - \widehat{\mathbf{F}}_g\overline{\boldsymbol{\lambda}}_g) \tag{8.57}$$

$$\boldsymbol{u}_g = \mathbf{K}_g^\dagger\left(\boldsymbol{f}_g - \widehat{\mathbf{B}}_g^T\overline{\boldsymbol{\lambda}}_g + \mathbf{R}_g\overline{\boldsymbol{\alpha}}_g\right) \tag{8.58}$$

# Kapitola 9

# Implementace metody aktivní množiny a metody FETI-C

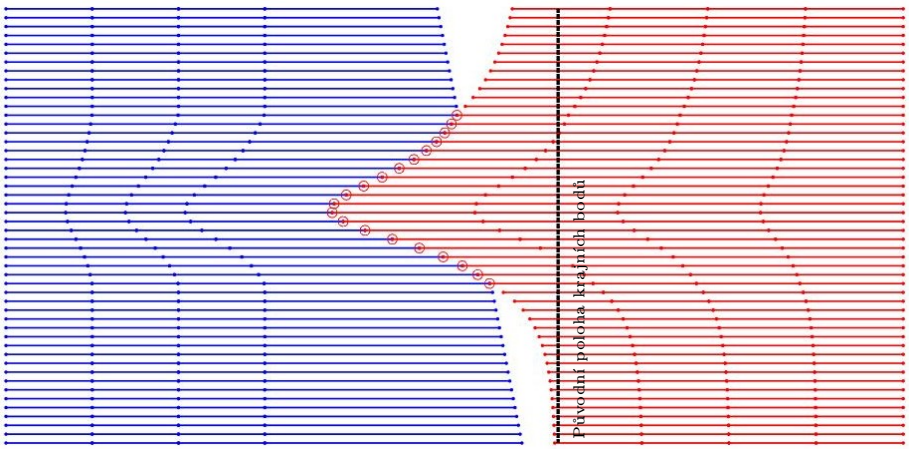



**Obrázek 9.1:** Zadání testovací úlohy

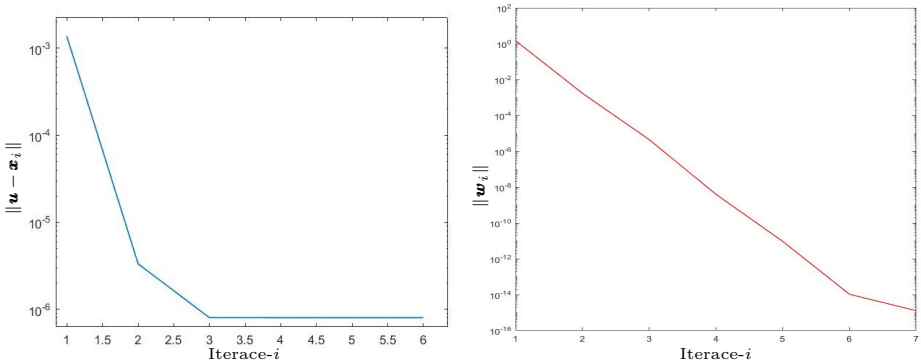
Pro implementaci metod aktivní množiny a metody FETI-C jsme využili prostředí programu MATLAB®. Pro testování jsme si zvolili jednoduchou úlohu a při jejím výpočtu jsme sledovali průběh několika důležitých ukazatelů, jako například průběhu odchylky od skutečného řešení a změnu pracovního setu v průběhu iterací.

Naše zkušební úloha je tvořena dvěma sadami protilehlých prutů, které jsou na protilehlých stranách vetknuty. Pruty na levé straně mají proměnou délku. Pruty na pravé straně jsou všechny stejně dlouhé, ale podléhají vnějšímu zatížení, které způsobuje, že se pruty na pravé straně protahují a mohou se dotknout prutů na levé straně. Různé pruty podléhají různému zatížení a mají různou tuhost. Každý prut je tvořen dvěma doménami a pěti body.

Řešení testovací úlohy můžete vidět na obrázku 9.2. Díky různému rozdělení zatížení a tuhostí materiálu se místa s aktivním působením kontaktních podmínek nacházejí ve středu úlohy, nikoliv v místě s nejmenší vzdáleností mezi levou a pravou skupinou prutů. Místa, kde dochází ke kontaktu, jsou označena červeným kroužkem. V našem zkušebním příkladu byly zadefinovány pouze podmínky bod na bod, takže povolujeme přenášení kontaktních sil jen v místě uzlů.




**Obrázek 9.2:** Řešení testovací úlohy



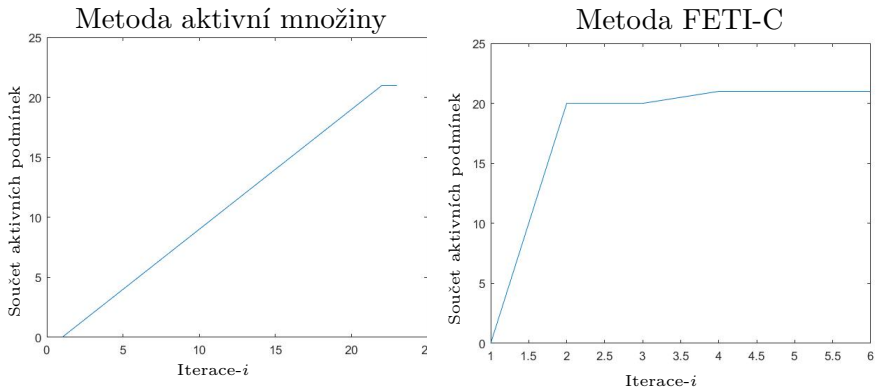

**a)** **b)**

**Obrázek 9.3:** Porovnání vývoje chyby a velikosti vyhledávacího kroku metoda FETI-C

**a)** Velikost chyby v jednotlivých iteracích.
**b)** Velikost vyhledávacího směru.

Na obrázku 9.3 můžeme vidět porovnání průběhu velikosti chyby $\|\boldsymbol{e}\|$ a velikosti vyhledávacího kroku $\|\boldsymbol{w}\|$ v jednotlivých iteracích metody FETI-C. Všimněme si, že v takto jednoduché úloze nachází metoda FETI-C řešení s přesností na šest desetinných míst již po šesti iteracích, a to v případě, kdy vektor $\boldsymbol{u}$ obsahuje pět set stupňů volnosti.

Pro zjištění skutečného řešení byla použita funkce *quadprog*, která řeší úlohu 9.1 přímo a obsahuje metodu aktivní množiny, která byla vysvětlena v kapitole 7.1 .




**a)** **b)**

**Obrázek 9.4:** Porovnání vývoje množství aktivních podmínek při metodě aktivního setu a metodě FETI-C

**a)**Množství všech aktivních podmínek při metodě aktivního setu
**b)**Množství všech aktivních podmínek metoda FETI-C

Všimněme si, že u takto jednoduché úlohy nachází metoda FETI-C řešení velmi rychle, i aktivní množinu nalezne téměř okamžitě. Oproti tomu metoda aktivní množiny vyžaduje podstatně větší množství iterací. Důvodem je to, že metoda aktivní množiny mění pracovní množinu pouze přidáním nebo odebráním jedné podmínky, za to metoda FETI-C zahrnuje všechny podmínky, které jsou v dané iteraci platné.

Metoda aktivní množiny musí postupovat tímto způsobem, protože kdyby přidávala větší množství podmínek, mohlo by docházet k oscilacím a v důsledku toho i ke kolapsu úlohy. Metoda FETI-C sice těmto oscilacím podléhá také, ale je stabilnější a nalézá řešení i v případech, kdy k oscilacím dochází.

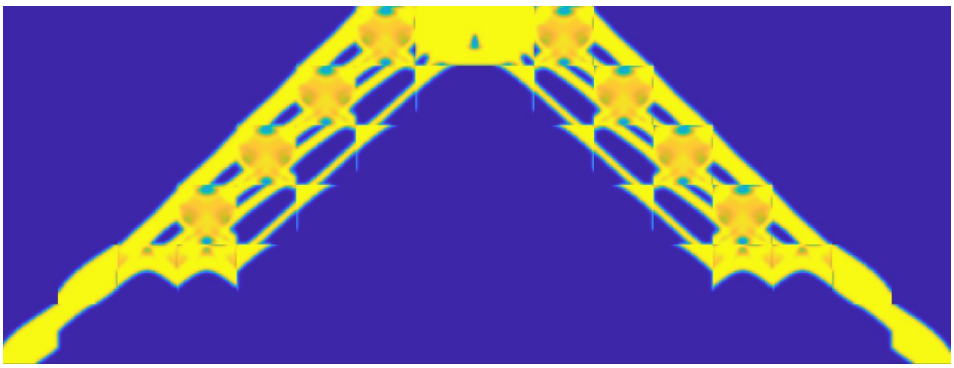
# Kapitola 10

# Závěr

V předchozích kapitolách jsme hovořili o různých numerických metodách.

Postupně jsme se seznámili se způsoby, jak řešit soustavu lineárních rovnic jako úlohu kvadratického programování s využitím gradientních řešičů a metody FETI. Potom jsme se zabývali kontaktními úlohami neboli soustavami lineárních rovnic s omezeními ve tvaru nerovnic a následně jsme je řešili metodou aktivní množiny. Nakonec jsme tyto metody propojili do metody FETI-C.

Tyto postupy nám umožňují řešit velké množství soustav lineárních rovnic o velkém množství neznámých. Velmi často je taková soustava rovnic výstupem z metody konečných prvků, kde jednotlivé prvky reprezentují chování materiálu klidně i na mikroskopické a menší úrovni. Díky tomu můžeme zkoumat chování konstrukce při různých rozloženích materiálů, a tedy i tuhostech v konstrukci.



**Obrázek 10.1:** Příklad úlohy s různým rozložením tuhostí materiálu
Topologie MBB nosníku získaná metodou SIMP -převzato z [MDPZ21]

Představme si, že jsme schopni vypočítat rozložení napětí v železobetonu na úrovni kameniva a výztuže, a tím jsme schopni určit význam každé části směsi pro přenos zatížení. Schopnost vypočítat takové úlohy s takovou přesností nám umožňuje dosáhnout lepší optimalizace konstrukcí a může vést k lepšímu pochopení chování jednotlivých složek konstrukce.

Navíc, díky tomu, že jsme propojili schopnost řešit velké soustavy lineárních rovnic se schopností řešit kontaktní úlohy, jsme umožnili výpočet se zahrnutím vlivu vzniku trhlin nebo delaminace konstrukce. Tento způsob výpočtu nám umožňuje řešit nejen otázky, zda trhlina vznikla, ale i to, jak se budou síly přenášet v případě opětovného svírání trhliny.

Kontaktní problémy se nenacházejí pouze ve stavebních úlohách. Mnohem obvyklejší jsou v oblasti strojního inženýrství, kde je potřeba řešit stav, kdy nejsou materiály pevně spojeny a dochází mezi nimi ke kontaktu mnohem častěji a kde nutnost optimalizovat výrobek je mnohem výraznější než u stavebních konstrukcí, které jsou vždy svým způsobem unikátní.

Součástí práce byla i implementace jednotlivých metod v programu MATLAB®. Z porovnání jednotlivých implementací ke konci práce plynou výhody FETI-C oproti metodě aktivní množiny. Z výsledků je zřejmé, že FETI-C nachází řešení v menším počtu iterací a při tom dosahuje požadované přesnosti.

Avšak v obou metodách je rozdíl v jejich inicializaci. Zatímco metoda aktivní množiny potřebuje předem znát pouze zadání úlohy, metoda FETI-C potřebuje vypočítat množství dalších matic, z nichž například tvorba pseudoinverzní matice $\mathbf{K}_g^\dagger$ nebo hledání jádra $\mathbf{R}_g$ pro podepřené domény, může být výpočetně náročné.

Nicméně pro rozměrné úlohy je obvykle méně náročné spočítat předem několik složitějších matic než opakovaně hledat optimum kvadratického problému.

# Literatura


[BF91] Franco Brezzi and Michel Fortin, *Mixed and hybrid finite element methods*, Springer Series in Comutational Mathematics, vol. 15, Springer Science & Business Media, 1991 (English).

[Bub06] František Bubeník, *Matematika 2*, vol. 172, Česká technika - nakladatelství ČVUT, 2006 (Czech).

[DF01] David Dureisseix and Charbel Farhat, *A numerically scalable domain decomposition method for the solution of frictionless contact problems*, International Journal for Numerical Methods in Engineering **50** (2001), no. 12, 2643–2666 (English).

[dO14] Oswaldo de Oliveira, *The implicit and the inverse function theorems: easy proofs*, Real Analysis Exchange **39** (2014), no. 1, 207–218.

[FMR94] Charbel Farhat, Jan Mandel, and Francois Xavier Roux, *Optimal convergence properties of the feti domain decomposition method*, Computer methods in applied mechanics and engineering **115** (1994), no. 3-4, 365–385 (English).

[HKKP07] Jaroslav Haslinger, Tomas Kozubek, Radek Kučera, and Gunther Peichl, *Projected schur complement method for solving nonsymmetric systems arising from a smooth fictitious domain approach*, Numerical Linear Algebra with Applications **14** (2007), no. 9, 713–739 (English).

[Hú14] Jana Húrková, *Metoda rozkladu oblasti-porovnání přístupů založených na primární a duální formulaci*, 2014.

[KVM+13] Tomás Kozubek, Vít Vondrák, M Menšık, David Horák, Zdenek Dostál, Vaclav Hapla, P Kabelıková, and Martin Čermák, *Total feti domain decomposition method and its massively parallel implementation*, Advances in Engineering Software **60** (2013), 14–22 (English).

[MDPZ21] Tomáš Medřickỳ, Martin Doškář, Ivana Pultarová, and Jan Zeman, *Comparison of feti-based domain decomposition methods for topology optimization problems*, Acta Polytechnica CTU Proceedings (2021), 7 (English).

[NW06] Jorge Nocedal and Stephen J Wright, *Numerical optimization*, Springer Series in Oprations Research, vol. 651, Springer, 2006 (English).

[She94] Jonathan Richard Shewchuk, *An introduction to the conjugate gradient method without the agonizing pain*, 1994.