

UNIVERZA V LJUBLJANI
FAKULTETA ZA MATEMATIKO IN FIZIKO

Matematika – 2. stopnja

Jure Slak

**REŠEVANJE LINEARNIH ELASTOSTATIČNIH
PROBLEMOV Z BREZMREŽNIMI METODAMI**

Magistrsko delo

Mentor: doc. dr. George Mejak

Somentor: dr. Gregor Kosec

Ljubljana, 2017

Univerza v Ljubljani
Fakulteta za matematiko in fiziko

Izjava o avtorstvu, istovetnosti tiskane in elektronske verzije magistrskega dela in
objavi osebnih podatkov študenta

Spodaj podpisani študent Jure Slak avtor magistrskega dela (v nadaljevanju:
pisnega zaključnega dela študija) z naslovom:

Reševanje linearnih elastostatičnih problemov z brezmrežnimi metodami

IZJAVLJAM

1. *Obkrožite eno od variant a) ali b)*

- a) da sem pisno zaključno delo študija izdelal samostojno;
- b) da je pisno zaključno delo študija rezultat lastnega dela več kandidatov in izpolnjuje pogoje, ki jih Statut UL določa za skupna zaključna dela študija ter je v zahtevanem deležu rezultat mojega samostojnega dela;

pod mentorstvom doc. dr. Georgea Mejaka
in somentorstvom dr. Gregorja Kosca.

- 2. da je tiskana oblika pisnega zaključnega dela študija istovetna elektronski obliki pisnega zaključnega dela študija;
- 3. da sem pridobil vsa potrebna dovoljenja za uporabo podatkov in avtorskih del v pisnem zaključnem delu študija in jih v pisnem zaključnem delu študija jasno označil;
- 4. da sem pri pripravi pisnega zaključnega dela študija ravnal v skladu z etičnimi načeli in, kjer je to potrebno, za raziskavo pridobil soglasje etične komisije;
- 5. da soglašam, da se elektronska oblika pisnega zaključnega dela študija uporabi za preverjanje podobnosti vsebine z drugimi deli s programsko opremo za preverjanje podobnosti vsebine, ki je povezana s študijskim informacijskim sistemom fakultete;
- 6. da na UL neodplačno, neizključno, prostorsko in časovno neomejeno prenašam pravico shranitve avtorskega dela v elektronski obliki, pravico reproduciranja ter pravico dajanja pisnega zaključnega dela študija na voljo javnosti na svetovnem spletu preko Repozitorija UL;
- 7. da dovoljujem objavo svojih osebnih podatkov, ki so navedeni v pisnem zaključnem delu študija in tej izjavi, skupaj z objavo pisnega zaključnega dela študija.

Kraj:

Podpis študenta:

Datum:

Kazalo vsebine

1 Uvod	1
1.1 Notacija	1
1.2 Osnovne trditve vektorske analize	2
1.3 Osnovni izreki funkcionalne analize	4
2 Teorija linearne elastičnosti	7
2.1 Osnove gibanja	7
2.1.1 Aksiomi gibanja	10
2.2 Napetostni tenzor	12
2.3 Enačbe gibanja	16
2.4 Konstitutivne enačbe	17
2.4.1 Mera deformacije	17
2.4.2 Hookov zakon	21
2.5 Navierova enačba	23
2.6 Obstoj in enoličnost rešitve	25
2.7 Priprava na numerično reševanje	28
2.7.1 Poenostavitev na dve dimenziji	28
2.7.2 Robni pogoji	29
3 Numerična metoda	31
3.1 Izpeljava	31
3.1.1 Ideja in motivacija	31
3.1.2 Splošna izpeljava	33
3.2 Posebni primeri	37
3.3 Algoritem	40
3.3.1 Diskretizacija	40
3.3.2 Goščenje diskretizacije	44
3.3.3 Iskanje najbližjih sosedov	45
3.3.4 Reševanje razpršenega sistema	46
3.3.5 Časovna zahtevnost	46
3.3.6 Prostorska zahtevnost	47
3.4 Pogoste vrednosti parametrov	47
4 Osnovni numerični zgledi	51
4.1 Enodimensionalni robni problem	51
4.2 Poissonova enačba	53
4.3 Vpet nosilec	57
4.4 Hertzev kontaktni problem	61
5 Zgled iz mehanike utrujenosti materialov	67
6 Implementacija	73
6.1 Vzporedno računanje	74
7 Zaključek	77
Literatura	79

Kazalo slik

1	Primer gibanja enotske kocke.	8
2	Valj, uporabljen v dokazu Cauchyjeve recipročne relacije.	13
3	Tetraeder, uporabljen v dokazu Cauchyjevega izreka o napetosti.	15
4	Razteg tanke palice vzdolž njene osi.	18
5	Domena in diskretizacija notranjosti in roba.	34
6	Primeri bolj zapletenih domen in njihovih diskretizacij.	42
7	Primerjava naključno diskretizirane in izboljšane domene.	43
8	Sprememba kvalitete diskretizacije po izboljšavi.	43
9	Algoritem za goščenje točk v domeni.	45
10	Najpogosteje uporabljene radialne bazne funkcije	49
10a	Multikvadratične	49
10b	Inverzne multikvadratične	49
10c	Gaussove funkcije	49
11	Napaka FDM in MLSM metode.	51
12	Primerjava časa izvajanja MLSM in FDM metod.	52
13	Konvergenca MLSM pri reševanju Poissonove enačbe	54
14	Čas izvajanja MLSM pri reševanju Poissonove enačbe	55
15	Konvergenčne krivulje za Gaussove funkcije.	55
16	Reševanje Poissonove enačbe z Gaussovimi funkcijami in utežjo.	56
16a	Napaka v odvisnosti od σ_b in σ_w	56
16b	Število odrezanih singularnih vrednosti.	56
17	Reševanje $\Delta u = 1$ na zanimivejših domenah.	57
18	Matrika sistema enačb pri reševanju problema vpetega nosilca.	58
19	Numerična rešitev problema vpetega nosilca.	59
20	Konvergenca numerične metode pri reševanju problema vpetega nosilca.	60
21	Numerična rešitev upogiba nosilca z nekaj luknjami.	60
22	Obravnavan Hertzev kontaktni problem.	61
22a	Stik dveh vzporednih valjev.	61
22b	Robni pogoji za numerično rešitev.	61
23	Napetosti pod območjem kontakta med valjem in polravnino.	63
24	Konvergenca metode pri reševanju Hertzevega kontaktnega problema.	63
25	Napaka v odvisnosti od velikosti domene.	64
26	Primer zgoščene domene s katero rešujemo Hertzev kontaktni problem.	65
26a	Sedemnajstkrat zgoščena diskretizacija.	65
26b	Gostota diskretizacije zgoščene domene.	65
27	Napaka metode pri različnih nivojih sekundarnega zgoščevanja.	65
28	Numerična rešitev Hertzevega kontaktnega problema.	66
29	Eksperiment za raziskovanje utrujanja materialov.	67
29a	Shema eksperimenta.	67
29b	Fotografija eksperimenta	67
30	Domena in robni pogoji za numerično reševanje problema (5.5).	69
30a	Primer napetosti $p(x)$ in $q(x)$	69
30b	Domena in robni pogoji.	69
31	Ujemanje MLSM in FEM rešitve pri analizi kontakta.	69
32	Primer rešitve problema (5.5) za $\mu = 0.65$ in $R = 50$ mm.	71
33	Diagram implementacije in izvedbe MLSM	73

34	Čas izvajanja delov programa pri reševanju Poissonove enačbe.	74
35	Faktorji pohitritve pri reševanju Poissonove enačbe.	75
36	Čas izvajanja delov programa pri uporabi 12 niti.	75

Program dela

V delu predstavite teorijo linearne elastičnosti s poudarkom na natančni matematični formulaciji. Izpeljite pripadajoče enačbe ravnovesja in nato z orodji funkcionalne analize obravnavajte enoličnost obstoja rešitve.

Iz metode končnih diferenc izpeljite bolj splošno numerično metodo, ki bo omogočala obravnavo neregularnih oblik, zgoščevanje porazdelitve računskih točk in spreminjanje reda natančnosti aproksimacij odvodov. Analizirajte delovanje metode na preprostih primerih in postopoma nadgrajujte problem do Hertzevega problema. Kot zadnji primer predstavite rešitev Hertzevega kontakta z upoštevanjem trenja in lepenja na stični površini. Kjer je mogoče rešitve podprite s primerjavami z analitičnimi rešitvami in konvergenčnimi analizami.

Osnovna literatura: [LC09], [Sla12], [Her82] [Gur82] (prvi del), [LL02], [ZT00], [LG01], [CLE06], [TK15] (drugi del).

Podpis mentorja:

Podpis somentorja:

Reševanje linearnih elastostatičnih problemov z brezmrežnimi metodami

POVZETEK

V delu je predstavljena teorija linearne elastičnosti s poudarkom na natančni matematični formulaciji. Izpeljana je klasična Cauchyjeva momentna enačba, ki opisuje gibanje kontinuma pod vplivom zunanjih sil. Pojasnjena je konstitutivna relacija med deformacijo in napetostjo, s pomočjo katere je izpeljana Navierova enačba. Pod primernimi pogoji sta dokazana obstoj in enoličnost njenih rešitev v šibkem smislu. Za pripravo na numerično reševanje sta obravnavani poenostaviti enačbe na dve dimenziji. Iz metode končnih differenc je izpeljana novejša, brezmrežna MLSM (Meshless Local Strong Form Method) metoda. Predstavljena je skupaj s postopki za generiranje, goščenje in razporejanje diskretizacijskih točk. Na klasičnem primeru Poissonove enačbe je narejena analiza obnašanja metode glede na različne izbire parametrov. Pokazano je, da je tudi s to metodo možno rešiti klasične probleme iz elastostatike, in sicer je rešen problem vpetega nosilca in Hertzov kontaktni problem, na katerem je prikazano tudi delovanje algoritma za goščenje diskretizacije do vsaj 17 nivojev globoko. To služi kot predpripjava za reševanje elastostatičnega problema, ki izvira iz novejših raziskav na področju mehanike utrujenosti materialov in pokazano je ujemanje rešitve, dobljene z MLSM, z objavljenim referenčno rešitvijo. Za konec se posvetimo še implementaciji metode, času izvajanja in možnosti paralelizacije.

Solving linear elastostatic problems using meshless methods

ABSTRACT

A theory of linear elasticity with focus on exact mathematical formulation is presented in the work. The classical Cauchy momentum equation describing the motion of a continuum under external forces is described. Constitutive relation between stress and strain is explained, leading to the formulation of the Navier equation. Under suitable assumptions, the existence and uniqueness of its weak solutions is proven. Two simplifications that reduce the problem to two dimensions are presented. A recently developed MLSM (Meshless Local Strong Form Method) method is derived from the Finite Difference Method. It is presented along with algorithms for domain discretization, relaxation and refinement. Analysis of the method's behaviour is made using the Poisson equation as a basic example. It is demonstrated that MLSM can be used to successfully solve elastostatic problems, namely the cantilever beam problem and Hertzian contact problem. The latter is also used to demonstrate extensive use of domain refinement to the depth of 17 layers. This serves as a preparation for a more complex elastostatic problem which originates from recent research on fretting fatigue. It is shown that the solution acquired with MLSM matches a published reference solution. Finally, implementation details, execution time and parallelisation possibilities are discussed.

Math. Subj. Class. (2010): 74B05, 65N99, 65Y20

Ključne besede: linearna elastostatika, brezmrežne metode, problem vpetega nosilca, Hertzov kontaktni problem, vzporedno izvajanje

Keywords: linear elastostatics, meshless methods, cantilever beam problem, Hertzian contact problem, parallel execution

1 Uvod

Teorija statične linearne elastičnosti opisuje trdnine pod obremenitvijo v mirovanju. Enačbe, ki opisujejo deformacije, so zelo redko rešljive v zaprti obliki, zato se njihovega reševanja lotimo z uporabo numeričnih metod. Najpogosteje se probleme linearne elastičnosti rešuje v šibki obliki z metodo končnih elementov [ZT00]. Že v preteklosti pa so te probleme reševali tudi v krepki obliki z metodo končnih diferenc [HH95], kasneje pa so se za reševanje elastostatičnih problemov začele uporabljati tudi novejše, brezmrežne metode v krepki in šibki obliki [CLE06, Mv15]. Prednost brezmrežnih metod je, da, za razliko od metode končnih elementov, ne potrebujejo polne diskretizacije domene z večotniki, temveč le nekaj diskretizacijskih točk v domeni in poznavanje njihove lokalne medsebojne lege. To brezmrežnim metodam omogoča večjo splošnost, lažjo posplošitev na višje dimenzije [DAM14, LCC13] in elegantnejšo implementacijo [KKS16]. Poleg tega imajo brezmrežne metode prednost pri reševanju nekaterih problemov, pri katerih je potrebno premikati rob domene ali točke v notranjosti, kot npr. simulacija širjenja razpok [RR00]. Eno izmed brezmrežnih metod [Kos16], ki je opisana tudi v tem delu, razvijamo s sodelavci v Laboratoriju za vzporedno in porazdeljeno računanje na Institutu "Jožef Stefan". S to brezmrežno metodo želimo razviti tudi originalno numerično rešitev problema širjenja razpok [PBT⁺16] v sklopu mednarodnega projekta "Multi-analysis of fretting fatigue using physical and virtual experiments" skupaj z Univerzo v Gentu in Univerzo v Luksemburgu.

V preostanku uvoda bomo ponovili matematične osnove, potrebne za razumevanje dela. V razdelku 2 si bomo ogledali matematično teorijo linearne elastodinamike, ki bo izpeljana iz splošnih aksiomov gibanja. Jasno bo označeno, na katerih mestih so narejene poenostavitve in pod kakšnimi predpostavkami. S tem bo nakazano, v katero smer se lahko razvijajo posplošitve teorije. Poseben poudarek bo na Navierovi enačbi ter enoličnosti in obstoju njenih rešitev, opisanih v razdelku 2.6, in poenostavtvah, ki jo naredijo primernejšo za numerično reševanje. Zgoraj omenjeno brezmrežno metodo bomo v razdelku 3 izpeljali iz znane metode končnih diferenc. Metoda bo izpeljana v sorazmerno visoki splošnosti in v posebnih primerih bomo prepoznali druge znane metode. Metodo bomo v razdelku 4 testirali na klasičnih zgledih in analizirali njeno obnašanje glede na različne izbire parametrov, ogledali pa si bomo tudi njeno obnašanje na problemu vpetega nosilca in Hertzevem kontaktnem problemu. Slednji bo služil tudi kot priprava na reševanje elastostatičnega kontaktnega problema, motiviranega s strani mehanike utrujenosti materialov. Rezultati, ki se ujemajo z nedavno objavljenimi [PBT⁺16], so opisani v razdelku 5. V razdelku 6 si ogledamo še implementacijske podrobnosti metode, komentiramo čas izvajanja in se na kratko posvetimo paralelizaciji.

1.1 Notacija

Skalarje bomo označevali z malimi latinskimi ali grškimi črkami, npr. $a, \alpha \in \mathbb{R}$. Vektorje iz \mathbb{R}^3 bomo označevali s puščico nad črko, npr. $\vec{v} \in \mathbb{R}^3$. Tenzorje (drugega reda) bomo označevali z malimi latinskimi ali grškimi črkami, npr. $t, \sigma \in \mathbb{R}^3 \otimes \mathbb{R}^3$. Identični tenzor, definiran z $I\vec{v} = \vec{v}$ bomo označevali z I . Komponente tenzorjev ali vektorjev bomo označevali z indeksom spodaj, npr. v_i ali t_{ij} . V razdelkih 1.2 in 2 bomo uporabljali tudi sumacijsko konvencijo na ponovljenem indeksu. Tako na

primer enačba

$$t_{ij}v_j = 0$$

vsebuje implicitno vsoto po j , razume pa se tudi, da velja za vsak i . Z $\vec{a} \cdot \vec{b}$ bomo označevali skalarni produkt (enojno kontrakcijo) vektorjev. Za skalarni produkt tenzorjev (dvojno kontrakcijo) bomo uporabljali oznako $s : t$, skalarni produkt v funkcionalno analitičnem smislu na nekem prostoru X pa bomo označevali z $\langle u, v \rangle_X$. S t^\top bomo označevali transpozicijo tenzorja ali matrike, definirano z $t^\top \vec{a} \cdot \vec{b} = \vec{a} \cdot t\vec{b}$. Z oznako $\langle \vec{a}, \vec{b}, \vec{c} \rangle$ bomo označevali mešani produkt vektorjev \vec{a} , \vec{b} in \vec{c} , definiran kot $\langle \vec{a}, \vec{b}, \vec{c} \rangle = (\vec{a} \times \vec{b}) \cdot \vec{c}$. Linearno ogrinjačo množice vektorjev bomo označevali z $\mathcal{L}in$. Enotske vektorje v tridimenzionalnem kartezičnem koordinatnem sistemu bomo označevali z \vec{i} , \vec{j} in \vec{k} .

Divergenco, gradient in Laplaceov operator bomo označevali z div , grad , Δ ali pa z $\nabla \cdot$, ∇ in ∇^2 . Odvode po koordinatah bomo označevali tudi z indeksom za vejico, npr. $\frac{\partial v_i}{\partial x_j} = v_{i,j}$. Prostor k -krat zvezno odvedljivih funkcij označujemo s C^k , prostor neskončnokrat zvezno odvedljivih funkcij s C^∞ , prostor neskončnokrat zvezno odvedljivih funkcij s kompaktnim nosilcem pa s C_c^∞ .

V razdelkih 3 in 4 bomo imeli opravka tudi z vektorji in matrikami splošnih dimenzijs, ki ne predstavljajo mehaničnih količin, ampak samo končno zaporedje skalarjev. Take vektorje bomo označevali odebeleno, npr. $\mathbf{u}, \boldsymbol{\varphi} \in \mathbb{R}^n$, matrike pa z velikimi tiskanimi črkami $A, B \in \mathbb{R}^{m \times n}$. Komponente bomo označevali z indeksom spodaj, npr. u_i ali A_{ij} .

1.2 Osnovne trditve vektorske analize

Spomnimo se nekaj osnovnih trditev vektorske analize, ki jih bomo potrebovali v kasnejših izpeljavah. Če ni drugače navedeno, bomo predpostavili obstoj in gladkost toliko odvodov, kot jih potrebujemo, ponavadi do drugega reda.

Prostor tenzorjev drugega reda opremimo s skalarnim produkтом

$$s : t = s_{ij}t_{ij} = \text{tr}(s^\top t). \quad (1.1)$$

Trditev 1.1. V zgornjem skalarnem produktu razpade prostor tenzorjev na direktno ortogonalno vsoto podprostorov simetričnih in antisimetričnih tenzorjev.

$$V \otimes V = \text{Sym}(V) \overset{\perp}{\oplus} \text{Asym}(V) \quad (1.2)$$

Dokaz. Vsak tenzor s lahko zapišemo kot $s = \frac{1}{2}(s + s^\top) + \frac{1}{2}(s - s^\top)$. Če je $s \in \text{Sym}(V) \cap \text{Asym}(V)$ je $s^\top = -s$ od koder sledi $s = 0$. Vsota je res direktna. Ker velja za simetričen s in antisimetričen t tudi

$$s : t = \text{tr}(s^\top t) = -\text{tr}(st^\top) = -\text{tr}(s^\top t) = -s : t,$$

od koder sledi $s : t = 0$, je vsota ortogonalna. \square

Definicija 1.2. Divergencia tenzorja t drugega reda je vektorsko polje, za katerega za vsak konstanten vektor \vec{a} velja

$$\text{div}(t) \cdot \vec{a} = \text{div}(t^\top \vec{a}). \quad (1.3)$$

V koordinatah je $(\text{div } t)_i = t_{ij,j}$.

Zakaj je ravno to prava definicija, pove naslednji izrek.

Izrek 1.3 (Gauss). *Naj bo Ω odprta povezana omejena množica z odsekoma gladkim robom, \vec{n} zunanja enotska normala na $\partial\Omega$ in t tenzorsko polje na Ω . Potem velja*

$$\int_{\partial\Omega} t\vec{n}dS = \int_{\Omega} \operatorname{div} t dV. \quad (1.4)$$

Dokaz. Naj bo \vec{a} poljuben konstanten vektor. Izračunajmo

$$\begin{aligned} \vec{a} \cdot \left(\int_{\partial\Omega} t\vec{n}dS \right) &= \int_{\partial\Omega} \vec{a} \cdot t\vec{n}dS = \int_{\partial\Omega} t^T \vec{a} \cdot \vec{n}dS = \int_{\Omega} \operatorname{div}(t^T \vec{a}) dV = \\ &= \int_{\Omega} \operatorname{div}(t) \cdot \vec{a} dV = \left(\int_{\Omega} \operatorname{div} t dV \right) \cdot \vec{a}. \end{aligned}$$

Pri računu smo uporabili definicijo t^T , definicijo divergence in Gaussov izrek za vektorska polja. Ker enakost velja za vsak vektor \vec{a} , velja tudi enakost vektorskih polj v izreku. \square

Definicija 1.4. Gradient vektorskega polja \vec{v} je tenzor drugega reda, definiran kot

$$(\operatorname{grad} \vec{v})^T \vec{a} = \operatorname{grad}(\vec{v} \cdot \vec{a}). \quad (1.5)$$

V koordinatah je $(\operatorname{grad} \vec{v})_{ij} = v_{i,j}$.

Opomba 1.5. Gradient vektorskega polja je diferencial (oz. v neki bazi Jacobijeva matrika) preslikave $\vec{x} \mapsto \vec{v}(\vec{x})$ in ga označujemo tudi $\frac{\partial \vec{v}}{\partial \vec{x}}$.

Definicija 1.6. Laplaceov operator je na vektorskem polju \vec{v} definiran kot

$$\Delta \vec{v} = \operatorname{div} \operatorname{grad} \vec{v}. \quad (1.6)$$

Pokažimo še nekaj osnovnih trditev, ki jih bomo potrebovali pri kasnejših izpeljavah.

Trditev 1.7. Za poljubno vektorsko polje \vec{v} velja

$$\operatorname{tr} \operatorname{grad} \vec{v} = \operatorname{div} \vec{v}. \quad (1.7)$$

Dokaz.

$$\operatorname{tr} \operatorname{grad} \vec{v} = (\operatorname{grad} \vec{v})_{ii} = v_{i,i} = \operatorname{div} \vec{v}. \quad \square$$

Trditev 1.8. Za poljubno vektorsko polje \vec{v} velja

$$\operatorname{div}(\operatorname{grad} \vec{v}^T) = \operatorname{grad} \operatorname{div} \vec{v}. \quad (1.8)$$

Dokaz.

$$\begin{aligned} \operatorname{div}(\operatorname{grad} \vec{v}^T)_i &= (\operatorname{grad} \vec{v}^T)_{ij,j} = (\operatorname{grad} \vec{v})_{ji,j} = v_{j,ij} = \\ &= v_{j,ji} = (\operatorname{grad}(v_{j,j}))_i = (\operatorname{grad} \operatorname{div} \vec{v})_i. \end{aligned}$$

Pri tem smo uporabili C^2 gladkost polja pri menjavi vrstnega reda parcialnih odvodov. \square

Trditev 1.9. Za poljubno skalarno polje φ velja

$$\operatorname{div}(\varphi I) = \operatorname{grad} \varphi. \quad (1.9)$$

Dokaz.

$$(\operatorname{div}(\varphi I))_i = (\varphi I)_{ij,j} = (\varphi \delta_{ij})_{,j} = \varphi_{,i} = (\operatorname{grad} \varphi)_i$$

□

Trditev 1.10. Za poljubno vektorsko polje \vec{v} in tenzor t velja

$$\operatorname{div}(t^\top \vec{v}) = \vec{v} \cdot \operatorname{div} t + t : \operatorname{grad} \vec{v}. \quad (1.10)$$

Dokaz.

$$\begin{aligned} (\operatorname{div}(t^\top \vec{v}))_i &= (t^\top \vec{v})_{i,i} = (t_{ij}^\top v_j)_{,i} = t_{ij,i}^\top v_j + t_{ij}^\top v_{j,i} = t_{ji,i} v_j + t_{ji} v_{j,i} = \\ &= (\operatorname{div} t)_j v_j + t : \operatorname{grad} \vec{v} = \operatorname{div} t \cdot \vec{v} + t : \operatorname{grad} \vec{v} \end{aligned}$$

□

1.3 Osnovni izreki funkcionalne analize

Ponovimo še nekaj osnovnih definicij in izrekov funkcionalne analize, ki jih bomo potrebovali pri obravnavi enoličnosti in obstoja rešitve linearnih parcialnih diferencialnih enačb. Enačbe bodo veljale na neki množici Ω , za katero bomo predpostavili, da je odprta in povezana z odsekoma gladkim robom. Taki množici bomo rekli *domena*. Gaussov izrek 1.3 tako velja za omejene domene. Za matematično natančno obravnavo parcialnih diferencialnih enačb in njihovih odvodov bomo potrebovali ustrezne funkcijalne prostore, t.i. prostore Soboljeva. Na temo prostorov Soboljeva je na voljo ogromno literature [AF03, DNPV12, Maz85]. Izreki in definicije v tem razdelku so bili povzeti po [AF03].

Definicija 1.11. Prostor Soboljeva $W_{k,p}(\Omega)$ za $k \in \mathbb{N}$ in $p \in [1, \infty)$ nad domeno Ω je definiran kot

$$W^{k,p}(\Omega) = \{u: \mathbb{R}^d \rightarrow \mathbb{R}; D^\alpha u \in L^p \text{ za vsak } |\alpha| \leq k\}. \quad (1.11)$$

Pri tem je $\alpha = (\alpha_1, \dots, \alpha_d)$ multiindeks, $|\alpha| = \sum_i \alpha_i$ in $D^\alpha f = \frac{\partial^{|\alpha|}}{\partial x_1^{\alpha_1} \dots x_d^{\alpha_d}}$ šibki odvod funkcije f .

Opomba 1.12. Pravimo, da je v šibki odvod funkcije u , če je

$$\int_\Omega u D^\alpha \varphi = (-1)^{|\alpha|} \int_\Omega v \varphi, \quad (1.12)$$

za vsako testno funkcijo $\varphi \in C_c^\infty(\Omega)$.

Prostori Soboljeva opremimo z normo

$$\|u\|_{W^{k,p}(\Omega)} = \left(\sum_{|\alpha| \leq k} \|D^\alpha u\|_{L^p(\Omega)}^p \right)^{\frac{1}{p}}. \quad (1.13)$$

Obstajajo tudi druge definicije norme na prostorih Soboljeva, vendar vse pogosto uporabljene definicije definirajo ekvivalentne norme. V zgoraj definirani normi so prostori $W^{k,p}(\Omega)$ Banachovi. Za poseben primer $p = 2$ označimo $H^k(\Omega) = W^{k,2}(\Omega)$, ta prostor je Hilbertov. Vse zgornje definicije so obravnavale samo skalarne funkcije. Vektorske funkcije bomo obravnavali po komponentah in pisali $\vec{v} \in [H^1(\Omega)]^3$, kar pomeni, da je $v_i \in H^1(\Omega)$ za vsak i .

Trditev 1.13 (Kornova neenakost). Za vsako funkcijo $\vec{u} \in [H^1(\Omega)]^3$ in pozitivno definiten tenzor četrtega reda C velja

$$\int_{\Omega} (|\vec{u}|^2 + \|\operatorname{grad} \vec{u}\|^2) dV \leq c_1 \int_{\Omega} \frac{1}{4} (\operatorname{grad} \vec{u} + \operatorname{grad} \vec{u}^T) : C : (\operatorname{grad} \vec{u} + \operatorname{grad} \vec{u}^T) dV, \quad (1.14)$$

za neko konstanto c_1 , neodvisno od \vec{u} .

Kornova neenakost je močan rezultat, ki povezuje normo prostorov Soboljeva z energijskimi normami, kar bomo videli v razdelku 2.6. Dokaz neenakosti je enostaven, če je \vec{u} gladka funkcija in enaka 0 na $\partial\Omega$. Velja pa neenakost tudi pri šibkejših predpostavkah na Ω in za \vec{u} z neničelnimi vrednostmi na robu. Dokaz neenakosti v posebnem primeru najdemo v [LC09, str. 229] splošnejši dokaz pa v [Cia10].

Vložitve prostorov Soboljeva v Lebesgueove L^p prostore je raziskoval že Soboljev sam [Sob38], kasneje pa še mnogi drugi [BW80, ER00, EH87]. V [AF03] je problemu vložitve posvečeno celo četrtto poglavje, kjer je tudi dokaz spodnjega izreka [AF03, str. 85, izrek 4.12].

Izrek 1.14 (Rellich–Kondrachov). Naj bo Ω domena v \mathbb{R}^d in $1 \leq kp < d$. Naj bo $p^* = \frac{dp}{d-kp}$. Potem lahko prostor Soboljeva $W^{k,p}(\Omega)$ zvezno vložimo v prostor $L^{p^*}(\Omega)$ in kompaktno vložimo v $L^q(\Omega)$ za $1 \leq q < p^*$.

Opomba 1.15. Za poseben primer $d = 3$, $k = 1$ in $p = 2$ dobimo $p^* = 6$ in s tem zvezno vložitev

$$H^1(\Omega) \hookrightarrow L^6(\Omega). \quad (1.15)$$

Poleg zgornjega izreka, ki bo prek zveznosti in posledično omejenosti vložitve poskrbel za oceno norme, potrebujemo tudi izrek, ki bo enako naredil za rob domene. O tem govori izrek o sledi [AF03, str. 164, izrek 5.36].

Izrek 1.16 (Izrek o sledi). Naj bo Ω domena v \mathbb{R}^d in $1 \leq kp < d$. Naj bo $p^* = \frac{dp}{d-kp}$. Potem je operator zožitve, ki slika $W^{m,p}(\Omega) \rightarrow L^q(\partial\Omega)$, zvezen.

Opomba 1.17. Pogosto se tudi tej preslikavi reče vložitev, saj zožitve funkcij na rob domene vlagamo v druge prostore. Za primer $d = 3$, $k = 1$ in $p = 2$ dobimo $p^* = 4$ in s tem zvezno vložitev

$$H^1(\Omega) \hookrightarrow L^4(\partial\Omega). \quad (1.16)$$

Pri dokazu obstaja in enoličnosti rešitev linearnih parcialnih diferencialnih enačb si bomo pomagali tudi s klasičnim izrekom funkcionalne analize, dokazanim v vsakem učbeniku, ki obravnava to tematiko, npr. [Kre89, str. 188, izrek 3.8-1].

Izrek 1.18 (Rieszov reprezentacijski izrek). Naj bo H realen Hilbertov prostor in H^* njegov dualen prostor, torej prostor vseh zveznih linearnih funkcionalov na H . Tedaj obstaja izometrični izomorfizem $\Phi: H \rightarrow H^*$.

Pogosteje najdemo izrek zapisan v nekoliko drugačni obliki.

Posledica 1.19. Naj bo H Hilbertov prostor in $f: H \rightarrow \mathbb{R}$ zvezen linearen funkcional na H . Tedaj obstaja natanko en element $x_f \in H$, da je za vsak $y \in H$

$$f(y) = \langle y, x_f \rangle. \quad (1.17)$$

Poleg tega velja še $\|x_f\| = \|f\|$.

Opomba 1.20. Preslikava Φ iz izreka 1.18 priredi vsakemu $x_f \in H$ njegov $f \in H^*$ iz posledice 1.19.

2 Teorija linearne elastičnosti

Teorija linearne elastičnosti govori o deformaciji in napetostih v trdninah kot posledici delovanja zunanjih obremenitev. Trdnine modeliramo kot zvezno maso, ki zavzema celoten prostor, ki ga zaseda, in s tem zaobidemo težave, ki bi nastale z obravnavo njihove diskretne atomske strukture. To ne pomeni, da atomsko strukturo ignoriramo, temveč se zavedamo, da moramo vzeti v obravnavo dovolj velik volumen, ki ga imenujemo *reprezentativni volumski element* (REV), da odraža vse lastnosti materiala. Za volumne, manjše od RVE, ne moremo pričakovati, da bi jih teorija mehanike kontinuma dobro opisala. Dejstvo, da trdnino modeliramo kot kontinuum, nam omogoča, da jo delimo na poljubno majhne dele, in s tem dobimo osnovo za uporabo teorije infinitezimalnega računa. To modelsko predpostavko imenujemo *hipoteza kontinuuma*. Poleg tega bomo predpostavili tudi gladkost drugih obravnavanih količin, kot npr. gostota ali hitrost, s pomočjo katerih bomo na podlagi privzetih fizikalnih zakonov izpeljali diferencialne enačbe, ki opisujejo deformacije in napetosti v materialu. Izkaže se, da ta teorija zelo dobro opisuje materiale, ki se uporabljajo v gradbeništvu in strojništву, kot so kovine, beton in steklo, in je zaradi tega zelo pogosto uporabljen. Paketi za numerično analizo strukture zgradb so zato doživeli velik razvoj in komercialni uspeh [Sol, HKS, Aut].

Začnimo s tem, da natančno definiramo pojme kot so telo in gibanje, s pomočjo katerih bomo izpeljali enačbe, ki opisujejo pojave, ki nas zanimajo. Osnovna predstavitev teorije bo povzeta po [Sla12].

2.1 Osnove gibanja

Definicija 2.1. Materialno telo \mathcal{B} je odprta povezana podmnoga v \mathbb{R}^3 z odsekoma gladkim robom skupaj z družino bijekcij

$$\boldsymbol{\chi} = \{\chi: \mathcal{B} \rightarrow \chi(\mathcal{B}) \subseteq \mathbb{R}^3\}, \quad (2.1)$$

tako da je za vsaki $\chi_1, \chi_2 \in \boldsymbol{\chi}$ preslikava $\chi_2 \circ \chi_1^{-1}: \chi_1(\mathcal{B}) \rightarrow \chi_2(\mathcal{B})$ difeomorfizem.

Preslikavam $\chi \in \boldsymbol{\chi}$ pravimo *konfiguracije* telesa. Odlikovani konfiguraciji χ_R pravimo *referenčna konfiguracija* in območje, ki ga telo zaseda v tej konfiguraciji, bomo označevali z B , torej $B = \chi_R(\mathcal{B})$.

Definicija 2.2. Gibanje telesa \mathcal{B} je gladka družina konfiguracij

$$\{\chi_t: \mathcal{B} \rightarrow \chi_t(\mathcal{B}) \subseteq \mathbb{R}^3, t \in \mathbb{R}\}. \quad (2.2)$$

Tem konfiguracijam pravimo *prostorske konfiguracije*. Območje, ki ga telo zaseda v prostorski konfiguraciji ob času t , označujemo z B_t , torej $B_t = \chi_t(\mathcal{B})$.

Koordinatam telesa v referenčni konfiguraciji pravimo *referenčne koordinate* in jih pišemo z velikimi tiskanimi črkami, npr. $X \in \chi_R(\mathcal{B})$. Koordinatam telesa v prostorski konfiguraciji pravimo *prostorske koordinate* in jih pišemo z malimi črkami, npr. $x \in \chi_t(\mathcal{B})$. Krajevni vektor do točke z referenčnimi koordinatami X bomo označevali z \vec{X} in krajevni vektor do točke s prostorskimi koordinatami x bomo označevali z \vec{x} . Krajevnemu vektorju bomo rekli tudi *položaj* točke in pisali vse funkcije kot funkcije odvisne od položaja in ne direktno od koordinat.

Gladkost gibanja glede na t iz definicije 2.2 pomeni, da je preslikava

$$\begin{aligned}\tilde{x}: \mathbb{R} \times B \subset \mathbb{R} \times \mathbb{R}^3 &\rightarrow \mathbb{R}^3, \\ (t, \vec{X}) &\mapsto \chi_t(\chi_R^{-1}(\vec{X}))\end{aligned}\tag{2.3}$$

gladka kot funkcija iz \mathbb{R}^4 v \mathbb{R}^3 . Preslikava \tilde{x} podaja zvezo med prostorsko in referenčno konfiguracijo:

$$\vec{x} = \tilde{x}(t, \vec{X}).\tag{2.4}$$

Pogosto opustimo strog zapis s preslikavami in pišemo kar $\vec{x} = \vec{x}(t, \vec{X})$. Ker je za vsak t preslikava $\vec{X} \mapsto \vec{x}(t, \vec{X})$ difeomorfizem, lahko funkcijo obrnemo in izrazimo tudi referenčni položaj kot funkcijo prostorskega: $\vec{X} = \vec{X}(t, \vec{x})$.

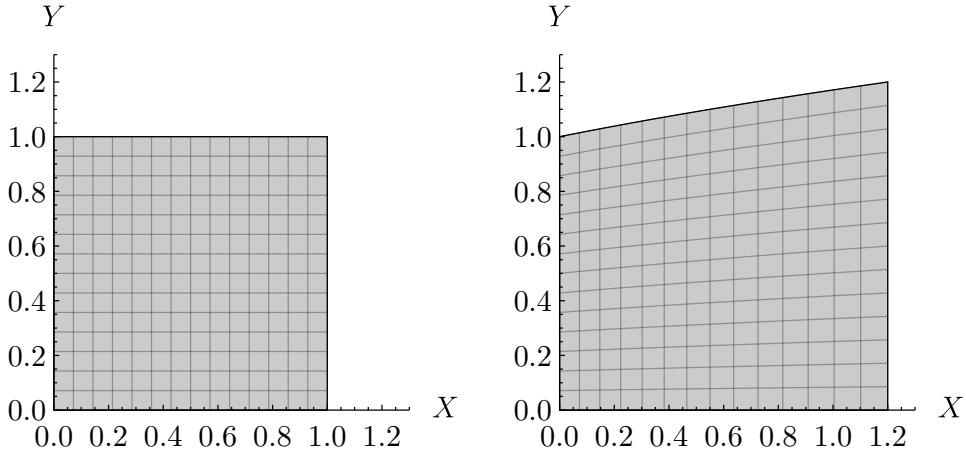
Pogosto za referenčno konfiguracijo vzamemo kar konfiguracijo na začetku gibanja, $\chi_R = \chi_{t=0}$, vendar pa to ni nujno.

Gibanje, kot opisano sedaj, opisuje realno makroskopsko gibanje. Pogoji gladkosti zagotavljajo, da telesa ne morejo kar izginiti in se pojaviti drugje ter da se ne morejo sploščiti ali sekati samih sebe.

Primer 2.3. Pišimo $\vec{X} = (X, Y, Z)$ in $\vec{x} = (x, y, z)$. Naj bo dano gibanje

$$\vec{x}(t, X, Y, Z) = (X + tX^2, Y + tXY, Z),\tag{2.5}$$

za $0 < X, Y, Z < 1$. Ob času $t = 0$ je telo v referenčni konfiguraciji. Telo v referenčni in prostorski konfiguraciji je prikazano na sliki 1. Za vsak $t \geq 0$ je \vec{x} difeomorfizem,



Slika 1: Prerez enotske kocke pri $Z = 0$ v referenčni konfiguraciji (levo) in v prostorski konfiguraciji ob času $t = 0.2$ (desno).

saj velja

$$\det \left(\frac{\partial \vec{x}}{\partial (X, Y, Z)} \right) = (1 + tX)(1 + 2tX) > 0.\tag{2.6}$$

Zato lahko izrazimo obratno preslikavo

$$\vec{X}(t, x, y, z) = \left(\frac{\sqrt{4tx + 1} - 1}{2t}, \frac{y(\sqrt{4tx + 1} - 1)}{2tx}, z \right).\tag{2.7}$$

Vsako količino φ , definirano na telesu \mathcal{B} , kot na primer temperaturo ali hitrost, lahko zapišemo na dva načina: v prostorskem ali v referenčnem položaju. Za funkcijo

$$\begin{aligned}\tilde{\varphi}: B &\rightarrow \varphi(\mathcal{B}), \\ \vec{X} &\mapsto \varphi(\chi_R^{-1}(\vec{X})),\end{aligned}\tag{2.8}$$

pravimo, da je zapisana v referenčnem položaju in pišemo $\tilde{\varphi} = \varphi(\vec{X})$. Podobno za funkcijo

$$\begin{aligned}\hat{\varphi}: B_t &\rightarrow \varphi(\mathcal{B}), \\ \vec{x} &\mapsto \varphi(\chi_t^{-1}(\vec{x})),\end{aligned}\tag{2.9}$$

pravimo, da je zapisana v prostorskem položaju in pišemo $\hat{\varphi} = \varphi(\vec{x})$. Med obema zapisoma lahko prehajamo s pomočjo zveze (2.4) ali njenega inverza kot $\varphi(\vec{x}) = \varphi(\vec{x}(t, \vec{X}))$ ali $\varphi(\vec{X}) = \varphi(\vec{X}(\vec{x}, t))$.

Podoben zapis bomo uporabljali tudi pri diferencialnih operatorjih. Pri odvodih moramo povedati, ali na količino gledamo v prostorskem ali v referenčnem položaju. Z velikimi črkami bomo pisali operatorje, kjer odvajamo glede na referenčni položaj, z malimi pa tiste, kjer odvajamo glede na prostorskega.

Primer 2.4. Naj bo dana količina $\vartheta(t, X, Y, Z) = tX^2 + 2ZY$, ki je zapisana v referenčnem položaju. Če jo zapišemo v prostorskih s pomočjo izpeljane inverzne relacije (2.7) iz primera 2.3, dobimo

$$\vartheta(t, x, y, z) = \frac{2tx^2 + 2yz(\sqrt{4tx+1} - 1) - x\sqrt{4tx+1} + x}{2tx}.$$

Demonstrirajmo še uporabo diferencialnih operatorjev, izračunajmo

$$\begin{aligned}\frac{\partial \vartheta}{\partial Y} &= 2Z, \\ \frac{\partial \vartheta}{\partial y} &= \frac{z(\sqrt{4tx+1} - 1)}{tx}.\end{aligned}$$

Te odvode bi ponovno lahko zapisali kot funkcije referenčnega ali prostorskega položaja. Podobno se obnašajo časovni odvodi.

$$\begin{aligned}\frac{D}{Dt} \vartheta &= X^2, \\ \frac{d}{dt} \vartheta &= -\frac{(-2tx + \sqrt{4tx+1} - 1)(x - 2yz)}{2t^2 x \sqrt{4tx+1}}.\end{aligned}$$

Zapis $\frac{D\vartheta}{Dt}(t, x, y, z)$ predstavlja torej časovni odvod ϑ pri konstantnem referenčnem položaju, zapisan v prostorski konfiguraciji. Izračunamo ga tako, da izraz prevedemo v referenčni položaj, odvajamo po času in prenesemo nazaj v prostorsko konfiguracijo. S simboli lahko to zapišemo kot

$$\left(\frac{D}{Dt} \vartheta \right) (t, \vec{x}) = \left(\frac{D}{Dt} \vartheta(\vec{x}(t, \vec{X})) \right) (\vec{X}(t, \vec{x})).\tag{2.10}$$

Definirajmo še osnovne pojme pomika, hitrosti in pospeška.

Definicija 2.5. Količino

$$\vec{u}(\vec{X}) = \vec{x}(t, \vec{X}) - \vec{X} \quad (2.11)$$

imenujemo *pomik*. Količino

$$\vec{v}(\vec{X}) = \frac{D\vec{x}}{Dt}(t, \vec{X}) \quad (2.12)$$

imenujemo *hitrost*. Količino

$$\vec{a}(\vec{X}) = \frac{D\vec{v}}{Dt}(t, \vec{X}) \quad (2.13)$$

imenujemo *pospešek*.

Trditev 2.6. *Odvod po času v referenčnem koordinatnem sistemu se lahko izračuna v prostorskem koordinatnem sistemu kot*

$$\frac{D\varphi}{Dt}(t, \vec{x}) = \frac{d\varphi}{dt}(t, \vec{x}) + ((\text{grad } \varphi)(t, \vec{x})) \vec{v}(t, \vec{x}). \quad (2.14)$$

Pri tem je φ skalarna, vektorska ali tenzorska količina.

Dokaz. Po definiciji najprej prenesemo φ v referenčni koordinatni sistem z uporabo 2.8. Nato odvajamo po verižnem pravilu

$$\begin{aligned} \frac{D\varphi}{Dt}(t, \vec{x}) &= \frac{D\varphi}{Dt}(t, \vec{x}(t, \vec{X})) = \frac{d\varphi}{dt}(t, \vec{x}) + \frac{\partial \varphi}{\partial \vec{x}} \frac{D\vec{x}}{Dt}(t, \vec{X}) = \\ &= \frac{d\varphi}{dt}(t, \vec{x}) + ((\text{grad } \varphi)(t, \vec{x})) \vec{v}(t, \vec{X}(\vec{x}, t)). \end{aligned} \quad \square$$

Poleg že naštetih količin pa imajo telesa tudi druge fizikalne lastnosti, kot so masa in volumen, ki vplivajo na gibanje. Za modeliranje teh si pomagamo z merami.

Definicija 2.7. Predpostavimo, da imamo na \mathcal{B} definirani dve σ -končni meri, m in V , ki nam predstavlja maso in volumen. Predpostavimo še, da je m absolutno zvezna glede na V , torej, če je volumen nekega podtelesa nič, je tudi njegova masa nič. Od tod po Radon-Nikodymovem izreku sledi, da obstaja merljiva funkcija $\rho = \frac{dm}{dV}$, da velja

$$m(A) = \int_A \rho dV \quad (2.15)$$

za vsako merljivo podmnožico $A \subseteq \mathcal{B}$. Funkciji $\rho: \mathcal{B} \rightarrow [0, \infty)$ pravimo *gostota*.

Opomba 2.8. Ponavadi za V vzamemo kar Lebesgueovo mero na \mathbb{R}^3 , mero m pa podamo tako, da podamo gostoto telesa ρ . Meri m in V s potiskom prek konfiguracij razširimo na referenčni in prostorski položaj.

2.1.1 Aksiomi gibanja

Iz mehanike točke in iz klasične mehanike vemo, da gibanje zadošča nekim zakonom, ki jih bomo za nadaljnje izpeljave privzeli kot aksiome.

Definicija 2.9. Množica $\mathcal{B}' \subseteq \mathcal{B}$ je *podtelo* telesa \mathcal{B} , če je sama telo za isto družino konfiguracij.

Aksiom 1 (Zakon o ohranitvi mase). Za vsako podtelo $\mathcal{B}' \subseteq \mathcal{B}$ in vsako njegovo gibanje velja

$$\frac{D}{Dt} m(B'_t) = 0. \quad (2.16)$$

Masa se med gibanjem niti ne izgublja niti ne nastaja in je vseskozi konstantna.

Druga dva aksioma bosta poleg mase imela opraviti s silami. Na kakšen način lahko sile delujejo na telo? En način so sile na daljavo, kot na primer gravitacija, ki delujejo na vsak košček telesa posebej. Drug način so kontaktne sile, ki nastanejo zaradi stika med telesi in se prenašajo preko površine. Sile v skladu s hipotezo kontinuma opišemo z njihovimi *gostotami* in vedno delujejo na telo v njegovi prostorski konfiguraciji, saj je to konfiguracija, ki jo telo dejansko zavzame med gibanjem. Če je sila konstantna, potem gostota sile predstavlja silo na enoto volumna ali površine, kar pogosto imenujemo *tlak*.

Definicija 2.10. *Volumenska gostota sile* \vec{f} je zvezna družina zveznih funkcij

$$\vec{f}: B_t \rightarrow \mathbb{R}^3. \quad (2.17)$$

Definicija 2.11. *Površinska gostota sile* \vec{t} je zvezna družina zveznih funkcij

$$\vec{t}_t: \partial B_t \rightarrow \mathbb{R}^3. \quad (2.18)$$

Opomba 2.12. Zveznost družine funkcij podobno kot v definiciji 2.2 pomeni, da je funkcija

$$\begin{aligned} \vec{f}: \mathbb{R} \times B &\subseteq \mathbb{R} \times \mathbb{R}^3 \rightarrow \mathbb{R}^3, \\ (t, \vec{X}) &\mapsto \vec{f}_t(\vec{x}(t, \vec{X})) \end{aligned} \quad (2.19)$$

zvezna kot funkcija iz \mathbb{R}^4 v \mathbb{R}^3 . Poenostavljeni pišemo \vec{f} in \vec{t} kar kot funkciji odvisni od t in \vec{x} , torej $\vec{f} = \vec{f}(t, \vec{x})$ in $\vec{t} = \vec{t}(t, \vec{x})$.

Primer 2.13. Gostota gravitacijske sile je enaka $\vec{f} = \rho \vec{g}$, kjer je \vec{g} gravitacijski pospešek in je neodvisna od časa in položaja.

Aksiom 2 (Zakon o ohranitvi gibalne količine). Za vsako podtelo $\mathcal{B}' \subseteq \mathcal{B}$ velja

$$\frac{D}{Dt} \int_{B'_t} \vec{v} dm = \int_{B'_t} \vec{f} dV + \int_{\partial B'_t} \vec{t} dS. \quad (2.20)$$

Spremembra gibalne količine je enaka vsoti vseh sil, ki delujejo na telo.

Aksiom 3 (Zakon o ohranitvi vrtilne količine). Za vsako podtelo $\mathcal{B}' \subseteq \mathcal{B}$ velja

$$\frac{D}{Dt} \int_{B'_t} \vec{x} \times \vec{v} dm = \int_{B'_t} \vec{x} \times \vec{f} dV + \int_{\partial B'_t} \vec{x} \times \vec{t} dS. \quad (2.21)$$

Spremembra vrtilne količine je enaka vsoti vseh zunanjih navorov, ki delujejo na telo.

Opomba 2.14. Pri aksiomu 3 smo predpostavili, da na kontinuum ne delujejo notranji navori, ampak da so vsi navori posledica delovanja zunanjih sil. Drugače povedano, predpostavili smo *nepolarnost* kontinuma.

V vseh treh aksiomih nastopa odvod po času v referenčni konfiguraciji nekega integrala, zapisanega v prostorski konfiguraciji. Tu ne velja običajen izrek o odvajajuju pod integralom, zato moramo izraz pod integralom najprej prenesti v referenčni položaj, zamenjati odvod in integral, nato pa ga prenesti nazaj. Storimo to najprej za aksiom o ohranitvi mase.

Trditev 2.15. *Masa se ohranja* ($\frac{D}{Dt} m(B_t) = 0$) *natanko tedaj, ko je*

$$\frac{D\rho}{Dt} + \rho \operatorname{div} \vec{v} = 0. \quad (2.22)$$

Dokaz. Trditev pokaže direkten račun, kjer bomo zamenjali integralsko spremenljivko iz \vec{x} v \vec{X} in nato nazaj. Pri uvedbi nove spremenljivke se bo v integralu pojavila determinanta $J = \det(F)$ diferenciala prehodne preslikave $F = \frac{\partial \vec{x}}{\partial \vec{X}}$, prav tako pa tudi njen odvod $\frac{DJ}{Dt}$, zato ga izračunajmo vnaprej:

$$\begin{aligned}\frac{DJ}{Dt} &= \text{tr} \left(\text{adj}(F) \frac{DF}{Dt} \right) = \text{tr} \left(\det(F) F^{-1} \frac{\partial^2 \vec{x}}{\partial \vec{X} \partial t} \right) = \det(F) \text{tr} \left(\frac{\partial \vec{v}}{\partial \vec{X}} F^{-1} \right) = \\ &= \det(F) \text{tr} \left(\frac{\partial \vec{v}}{\partial \vec{X}} \frac{\partial \vec{X}}{\partial \vec{x}} \right) = \det(F) \text{tr} (\text{grad } \vec{v}) = \det(F) \text{div } \vec{v}.\end{aligned}$$

Pri tem smo uporabili Jacobijevo formulo za računanje odvoda determinante, ciklično lastnost sledi, formulo za računanje odvoda inverza in verižno pravilo. Sedaj je vse pripravljeno za glavni izračun:

$$\begin{aligned}0 &= \frac{D}{Dt} m(B_t) = \frac{D}{Dt} \int_{B_t} dm = \frac{D}{Dt} \int_{B_t} \rho dV = \frac{D}{Dt} \int_B \rho J dV = \\ &= \int_B \frac{D}{Dt} (\rho J) dV = \int_B \left(\frac{D\rho}{Dt} J + \rho \frac{DJ}{Dt} \right) dV = \\ &= \int_B \left(\frac{D\rho}{Dt} + \rho \text{div } \vec{v} \right) J dV = \int_{B_t} \left(\frac{D\rho}{Dt} + \rho \text{div } \vec{v} \right) dV.\end{aligned}$$

Ker je integral enak nič za vsako telo, mora biti nujno tudi integrand enak nič, in obratno, če je integrand nič, se masa ohranja. \square

Zgornjo trditev bomo s pridom uporabili pri izračunu odvoda integrala splošne količine.

Trditev 2.16. *Naj bo φ neka skalarna, vektorska ali tenzorska količina in privzemimo aksiom o ohranitvi mase. Potem velja*

$$\frac{D}{Dt} \int_{B_t} \varphi dm = \int_{B_t} \frac{D\varphi}{Dt} dm. \quad (2.23)$$

Dokaz. Postopamo enako kot v prejšnji trditvi. Izračunamo

$$\begin{aligned}\frac{D}{Dt} \int_{B_t} \varphi dm &= \frac{D}{Dt} \int_B \varphi \rho J dV = \int_B \frac{D(\varphi \rho J)}{Dt} dV = \\ &= \int_B \left(\frac{D\varphi}{Dt} \rho J + \varphi \left(\underbrace{\frac{D\rho}{Dt} + \rho \text{div } \vec{v}}_{= 0 \text{ po (2.22)}} \right) \right) dV = \\ &= \int_B \frac{D\varphi}{Dt} \rho J dV = \int_{B_t} \frac{D\varphi}{Dt} dm.\end{aligned} \quad \square$$

2.2 Napetostni tenzor

Predstavljammo si, da na telo deluje neka površinska sila z gostoto \vec{t} . Ta sila deluje na zunanjo površino telesa, nato pa se prenaša skozi telo od točke do točke. V telesu si zamislimo neko podtelo. Na površini tega podtelesa zunanji del kontinuma deluje

na podtelo, toda ekvivalentno bi bilo, če bi zunanji del odstranili in na površino preostanka delovali z enakim poljem sil, kot je prej deloval zunanji del.

Ta razmislek ponuja naslednji opis notranjih in površinskih sil telesa. V nekem trenutku t se postavimo v neko točko \vec{x} v telesu in si zamislimo neko podtelo, tako da je izbrana točka na njegovi površini. Tedaj obstaja vektor \vec{t} , ki predstavlja silo, s katero preostali kontinuum deluje v tej točki na izbrano podtelo. Če bi si izbrali drugo podtelo, bi bila v splošnem sila drugačna. Toda, če sta imeli dve podtelesi v tisti točki enako zunanjo enotsko normalo \vec{n} na površino in posledično enako tangentno ravnino v tisti točki, bo vektor \vec{t} še vedno enak, saj bosta lokalno telesi izgledali enaki. Vektor \vec{t} je torej odvisen od izbrane točke, trenutka v času, ko ga opazujemo in normale na izbrano površino podtelesa, ter ničesar drugega; neodvisen je na primer od ukrivljenosti ploskve v tisti točki. Temu razmisleku pravimo *Cauchyjeva hipoteza*.

Aksiom 4 (Cauchyjeva hipoteza). Gostota površinske sile \vec{t} je odvisna samo od položaja, časa in normale na površino (resnično ali namišljeno), kjer jo opazujemo. S simboli lahko zapišemo

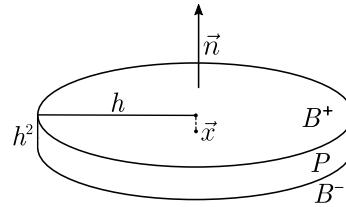
$$\vec{t} = \vec{t}(t, \vec{x}, \vec{n}). \quad (2.24)$$

Iz zgornjega razmisleka se zdi, da bi morala biti sila, s katero prvi del kontinuma deluje na drugega, nasprotno enaka sili, s katero drugi deluje na prvega. Temu se reče tudi tretji Newtonov zakon za mehaniko kontinuma in ga ni potrebno privzeti kot nov aksiom, temveč ga lahko dokažemo iz do sedaj privzetih aksiomov. Dokaza tega in naslednjega izreka sta povzeta po dokazih iz [Hje07, str. 104–107].

Trditev 2.17 (Cauchyjeva recipročna relacija). *Velja*

$$\vec{t}(t, \vec{x}, -\vec{n}) = -\vec{t}(t, \vec{x}, \vec{n}). \quad (2.25)$$

Dokaz. Izberimo trenutek t in točko \vec{x} . Krajše pišimo $\vec{t}_{\vec{n}} = \vec{t}(t, \vec{x}, \vec{n})$. Oglejmo si podtelo v obliki krožnega valja z višino h^2 in radijem h , ki ima v središču točko \vec{x} in je \vec{n} normala na eno izmed osnovnic (slika 2). Tako podtelo zaradi odprtosti \mathcal{B} gotovo obstaja za dovolj majhen h . Označimo osnovnico z normalo \vec{n} z B^+ , nasprotno z B^- in plašč s P . Površina osnovnice je πh^2 , površina plašča je $2\pi h^3$, volumen valja pa je πh^4 .



Slika 2: Valj, uporabljen v dokazu Cauchyjeve recipročne relacije.

Uporabimo za ta valj aksiom o gibalni količini (2.20):

$$\frac{D}{Dt} \int_{B_t} \vec{v} dm = \int_{B_t} \vec{f} dV + \int_{\partial B_t} \vec{t} dS.$$

Po trditvi 2.16 lahko zamenjamo odvod in integral, rob valja pa razpišimo po ploskvah:

$$\int_{B_t} \frac{D\vec{v}}{Dt} \rho dm = \int_{B_t} \vec{f} dV + \int_{B^+} \vec{t}_{\vec{n}} dS + \int_{B^-} \vec{t}_{-\vec{n}} dS + \int_P \vec{t}_{\vec{m}(\vec{x})} dS.$$

Vektor $\vec{m}(\vec{x})$ pri tem označuje normalo na plašč valja. Celotno enačbo pomnožimo s konstantnim vektorjem \vec{c} in na vsakem od integralov, ki so postali integrali skalarnih funkcij, uporabimo izrek o povprečni vrednosti

$$\begin{aligned} V(B)(\rho \frac{D\vec{v}}{Dt})(\vec{x}_1) \cdot \vec{c} &= V(B)\vec{f}(\vec{x}_2) \cdot \vec{c} + S(B^+) \vec{t}_{\vec{n}}(\vec{x}_3) \cdot \vec{c} + \\ &\quad + S(B^-) \vec{t}_{-\vec{n}}(\vec{x}_4) \cdot \vec{c} + S(P) \vec{t}_{\vec{m}(\vec{x})}(\vec{x}_5) \cdot \vec{c}, \end{aligned}$$

kjer so $\vec{x}_1, \vec{x}_2, \vec{x}_3, \vec{x}_4, \vec{x}_5$ neke vmesne točke v ali na površini valja. Vstavimo v enačbo izraze za volumen in površine ploskev in dobimo

$$\pi h^4 (\rho \frac{D\vec{v}}{Dt})(\vec{x}_1) \cdot \vec{c} = [\pi h^4 \vec{f}(\vec{x}_2) + \pi h^2 \vec{t}_{\vec{n}}(\vec{x}_3) + \pi h^2 \vec{t}_{-\vec{n}}(\vec{x}_4) + 2\pi h^3 \vec{t}_{\vec{m}(\vec{x})}(\vec{x}_5)] \cdot \vec{c}.$$

Celoten izraz delimo z πh^2 in pogledamo limito $h \rightarrow 0$: vse točke \vec{x}_i gredo proti \vec{x} , členi, pomnoženi s h^3 ali h^4 , gredo proti 0, in ostane enakost

$$0 = (\vec{t}_{\vec{n}}(\vec{x}) + \vec{t}_{-\vec{n}}(\vec{x})) \cdot \vec{c}.$$

Ker zgornja enakost velja za poljuben vektor \vec{c} , velja tudi

$$0 = (\vec{t}_{\vec{n}}(\vec{x}) + \vec{t}_{-\vec{n}}(\vec{x})),$$

kar je točno to, kar smo želeli pokazati. \square

Izkaže se, da velja še več, in sicer da je vektor \vec{t} linearno odvisen od normale. S pomočjo prejšnje trditve lahko pokažemo naslednji izrek, ki zagotovi obstoj matematičnega objekta, s pomočjo katerega opišemo napetost v materialu.

Izrek 2.18 (Cauchyjev izrek o napetosti). *Obstaja tenzor σ , tako da velja*

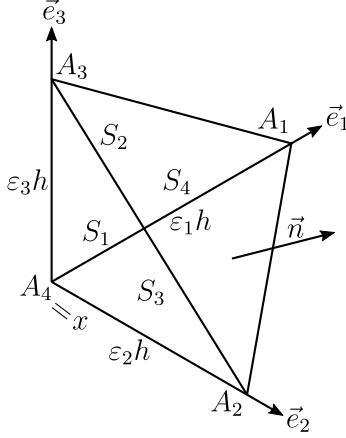
$$\vec{t}(t, \vec{x}, \vec{n}) = \sigma(t, \vec{x}) \vec{n}. \quad (2.26)$$

Tenzor σ se imenuje Cauchyjev napetostni tenzor.

Dokaz. Podobno kot pri Cauchyjevi recipročni relaciji si tokrat izberimo majhno telo z ogliščem v točki \vec{x} . Naj bo to tetraeder s tremi stranicami, vzporednimi koordinatnim osem, in normalo \vec{n} na ploskev, ki jo razpenjajo preostale tri stranice, kot prikazano na sliki 3.

Označimo oglišča tetraedra z $A_1, A_2, A_3, A_4 = \vec{x}$ in ploskve nasproti izbranega oglišča s S_1, S_2, S_3, S_4 . Ploščine teh ploskev označimo z a_1, a_2, a_3, a_4 . Naj bodo pravokotne stranice tetraedra dolge $\varepsilon_1 h, \varepsilon_2 h$ in $\varepsilon_3 h$, kjer razmerje $\varepsilon_1 : \varepsilon_2 : \varepsilon_3$ določimo tako, da je normala na S_4 enaka \vec{n} . Opisana konfiguracija je prikazana na sliki 3. Izračunamo lahko ploščine ploskev $a_1 = \frac{1}{2} \varepsilon_2 \varepsilon_3 h^2$, $a_2 = \frac{1}{2} \varepsilon_1 \varepsilon_3 h^2$ in $a_3 = \frac{1}{2} \varepsilon_1 \varepsilon_2 h^2$. Volumen tetraedra je $V = \frac{1}{6} \varepsilon_1 \varepsilon_2 \varepsilon_3 h^3$. Vektorje v smeri koordinatnih osi označimo z $\vec{e}_1, \vec{e}_2, \vec{e}_3$. Zunanja normala na ploskev S_1 je $-\vec{e}_1$, na ploskev S_2 je $-\vec{e}_2$ in na ploskev S_3 je $-\vec{e}_3$. Normalo na preostalo ploskev, ki bo dolga ravno toliko, kot je ploščina te ploskve, dobimo s pomočjo vektorskega produkta stranic, ki jo oklepata.

$$\begin{aligned} a_4 \vec{n} &= \frac{1}{2} (\varepsilon_1 h \vec{e}_1 - \varepsilon_2 h \vec{e}_2) \times (\varepsilon_3 h \vec{e}_3 - \varepsilon_2 h \vec{e}_2) = \\ &= -\frac{1}{2} \varepsilon_2 \varepsilon_3 h^2 \vec{e}_1 - \frac{1}{2} \varepsilon_1 \varepsilon_3 h^2 \vec{e}_2 - \frac{1}{2} \varepsilon_1 \varepsilon_2 h^2 \vec{e}_3 = \\ &= a_1 \vec{e}_1 + a_2 \vec{e}_2 + a_3 \vec{e}_3. \end{aligned}$$



Slika 3: Tetraeder, uporabljen v dokazu Cauchyjevega izreka o napetosti.

Če izraz pomnožimo z i -tim baznim vektorjem, dobimo $a_i = (\vec{n} \cdot \vec{e}_i)a_4$. Ponovno uporabimo aksiom 2 o ohranitvi gibalne količine, razpišemo integral po površini tetraedra po posameznih ploskvah, pomnožimo izraz s konstantnim vektorjem \vec{c} in nato uporabimo izrek o povprečni vrednosti:

$$\begin{aligned} \frac{D}{Dt} \int_{B_t} \vec{v} dm &= \int_{B_t} \vec{f} dV + \int_{\partial B_t} \vec{t} dS, \\ \int_{B_t} \rho \frac{D\vec{v}}{Dt} dV &= \int_{B_t} \vec{f} dV + \int_{S_1} \vec{t}_{-\vec{e}_1} dS + \int_{S_2} \vec{t}_{-\vec{e}_2} dS + \int_{S_3} \vec{t}_{-\vec{e}_3} dS + \int_{S_4} \vec{t}_{\vec{n}} dS, \\ V(\rho \frac{D\vec{v}}{Dt})(\vec{x}_1) \cdot \vec{c} &= \left[V\vec{f}(\vec{x}_2) + a_1 \vec{t}_{-\vec{e}_1}(\vec{x}_3) + a_2 \vec{t}_{-\vec{e}_2}(\vec{x}_4) + a_3 \vec{t}_{-\vec{e}_3}(\vec{x}_5) + a_4 \vec{t}_{\vec{n}}(\vec{x}_6) \right] \cdot \vec{c}, \end{aligned}$$

pri čemer so \vec{x}_i , kot v prejšnjem dokazu, neke točke v notranjosti ali na površini tetraedra. Sedaj enačbo delimo z a_4 in pošljemo limito $h \rightarrow 0$. Vse točke \vec{x}_i limitirajo proti \vec{x} , člena, ki vsebuje volumen tetraedra, pa se približujeta 0. Z upoštevanjem

$$\lim_{h \rightarrow 0} \frac{a_i}{a_4} = \lim_{h \rightarrow 0} \frac{(\vec{n} \cdot \vec{e}_i)a_4}{a_4} = \vec{n} \cdot \vec{e}_i$$

dobimo

$$0 = \left[\vec{t}_{\vec{n}} + \sum_{i=1}^3 (\vec{n} \cdot \vec{e}_i) \vec{t}_{-\vec{e}_i} \right] \vec{c}.$$

Zgornja enakost velja za vsak vektor \vec{c} ; ko uporabimo še Cauchyjevo recipročno relacijo (2.25), dobimo

$$\vec{t}_{\vec{n}} = \sum_{i=1}^3 (\vec{n} \cdot \vec{e}_i) \vec{t}_{\vec{e}_i}.$$

Ta zveza pove, kako je napetost na poljubni ploskvi povezana z napetostmi na koordinatnih ploskvah. To nam dovoljuje definicijo *Cauchyjevega napetostnega tenzorja* σ

$$\sigma = \sum_{i=1}^3 \vec{t}_{\vec{e}_i} \otimes \vec{e}_i,$$

za katerega res velja

$$\sigma \vec{n} = \sum_{i=1}^3 (\vec{t}_{\vec{e}_i} \otimes \vec{e}_i)(\vec{n}) = \sum_{i=1}^3 (\vec{n} \cdot \vec{e}_i) \vec{t}_{\vec{e}_i} = \vec{t}_{\vec{n}}. \quad \square$$

2.3 Enačbe gibanja

V tem razdelku bomo iz aksiomov izpeljali lokalne enačbe gibanja, tako da jih bomo iz integralske oblike prevedli v diferencialno. Prevedimo najprej aksiom 2 o ohranitvi gibalne količine.

Izrek 2.19 (Cauchyjeva momentna enačba). *Za gibanje kontinuma velja Cauchyjeva momentna enačba*

$$\rho \frac{D\vec{v}}{Dt} = \vec{f} + \operatorname{div} \sigma. \quad (2.27)$$

Opomba 2.20. Tako kot je enačba (2.25) analogija tretjega Newtonovega zakona, je enačba (2.27) analogija drugega Newtonovega zakona: na levi imamo produkt gostote mase in pospeška, na desni pa gostoto vseh sil, ki delujejo na telo.

Dokaz. Za vsako telo s prostorsko konfiguracijo B_t velja

$$\begin{aligned} \frac{D}{Dt} \int_{B_t} \vec{v} dm &= \int_{B_t} \vec{f} dV + \int_{\partial B_t} \vec{t} dS, \\ \int_{B_t} \frac{D\vec{v}}{Dt} \rho dV &= \int_{B_t} \vec{f} dV + \int_{\partial B_t} \sigma \vec{n} dS, \\ \int_{B_t} \frac{D\vec{v}}{Dt} \rho dV &= \int_{B_t} \vec{f} dV + \int_{B_t} \operatorname{div} \sigma dV, \\ 0 &= \int_{B_t} \left(\frac{D\vec{v}}{Dt} \rho - \vec{f} - \operatorname{div} \sigma \right) dV. \end{aligned}$$

V računu smo uporabili Gaussov izrek 1.3 za tenzorje drugega reda in izrek 2.16 o menjavi odvoda in integrala. Ker mora biti zadnji integral enak nič za vsako telo, mora biti tudi integrand enak nič, kar dokaže trditev. \square

Do sedaj še nismo uporabili aksioma 3 o vrtilni količini. Njegova lokalna oblika se prevede na zelo enostavno trditev o simetriji Cauchyjevega napetostnega tenzorja.

Trditev 2.21. *Cauchyjev napetostni tenzor je simetričen:*

$$\sigma^\top = \sigma. \quad (2.28)$$

Dokaz. Začnimo z zakonom o vrtilni količini (2.21) in ga prevedimo v lokalno obliko.

$$\frac{D}{Dt} \int_{B_t} \vec{x} \times \vec{v} dm = \int_{B_t} \vec{x} \times \vec{f} dV + \int_{\partial B_t} \vec{x} \times \vec{t} dS.$$

Posvetimo se zadnjemu členu in ga pomnožimo s konstantnim vektorjem \vec{w} :

$$\begin{aligned} \left(\int_{\partial B_t} \vec{x} \times \vec{t} dS \right) \cdot \vec{w} &= \int_{\partial B_t} \langle \vec{x}, \vec{t}, \vec{w} \rangle dS = \int_{\partial B_t} \langle \vec{w}, \vec{x}, \vec{t} \rangle dS = \\ &= \int_{\partial B_t} (\vec{w} \times \vec{x}) \cdot \vec{t} dS = \int_{\partial B_t} (\vec{w} \times \vec{x}) \cdot \sigma \vec{n} dS = \\ &= \int_{\partial B_t} \sigma^\top (\vec{w} \times \vec{x}) \cdot d\vec{S} = \int_{B_t} \operatorname{div}(\sigma^\top (\vec{w} \times \vec{x})) dV = \\ &= \int_{B_t} [\langle \operatorname{div} \sigma, \vec{w}, \vec{x} \rangle + \sigma : \operatorname{grad}(\vec{w} \times \vec{x})] dV. \end{aligned}$$

Zgoraj smo po vrsti uporabili: definicijo in cikličnost mešanega produkta, definicijo σ^\top , Gaussov izrek in relacijo iz trditve 1.10.

Podobno z \vec{w} pomnožimo tudi zakon o vrtilni količini in vstavimo zgornjo relacijo.

$$\begin{aligned}
0 &= \left(\frac{D}{Dt} \int_{B_t} \vec{x} \times \vec{v} dm - \int_{B_t} \vec{x} \times \vec{f} dV - \int_{\partial B_t} \vec{x} \times \vec{t} dS \right) \cdot \vec{w} = \\
&= \frac{D}{Dt} \int_{B_t} \langle \vec{x}, \vec{v}, \vec{w} \rangle dm - \int_{B_t} \langle \vec{x}, \vec{f}, \vec{w} \rangle dV - \int_{B_t} [\langle \operatorname{div} \sigma, \vec{w}, \vec{x} \rangle + \sigma : \operatorname{grad}(\vec{w} \times \vec{x})] dV = \\
&= \int_{B_t} \langle \rho \frac{D\vec{v}}{Dt}, \vec{w}, \vec{x} \rangle dV - \int_{B_t} \langle \vec{f}, \vec{w}, \vec{x} \rangle dV - \int_{B_t} [\langle \operatorname{div} \sigma, \vec{w}, \vec{x} \rangle + \sigma : \operatorname{grad}(\vec{w} \times \vec{x})] dV = \\
&= \int_{B_t} \left[\left\langle \rho \frac{D\vec{v}}{Dt} - \vec{f} - \operatorname{div} \sigma, \vec{w}, \vec{x} \right\rangle - \sigma : \operatorname{grad}(\vec{w} \times \vec{x}) \right] dV = \\
&= - \int_{B_t} \sigma : \operatorname{grad}(\vec{w} \times \vec{x}) dV.
\end{aligned}$$

Pri tem smo uporabili definicijo in cikličnost mešanega produkta, Cauchyjevo momentno enačbo (2.27) in dejstvo, da je

$$\frac{D}{Dt} \langle \vec{x}, \vec{v}, \vec{w} \rangle = \langle \vec{v}, \vec{v}, \vec{w} \rangle + \langle \vec{x}, \frac{D\vec{v}}{Dt}, \vec{w} \rangle = \langle \frac{D\vec{v}}{Dt}, \vec{w}, \vec{x} \rangle.$$

Ker aksiom o vrtilni količini drži za vsako telo, mora biti

$$\sigma : \operatorname{grad}(\vec{w} \times \vec{x}) = 0$$

za poljuben vektor \vec{w} . Tenzor $\operatorname{grad}(\vec{w} \times \vec{x})$ je antisimetričen in vsak antisimetričen tenzor lahko zapišemo v tej obliki s pomočjo njegovega osnega vektorja. Zgornja relacija je zato ekvivalentna trditvi, da velja

$$\sigma : w = 0$$

za vsak antisimetričen tenzor w . Torej je $\sigma \in \operatorname{Asym}(\mathbb{R}^3)^\perp$ in po trditvi 1.1 mora biti σ simetričen. \square

2.4 Konstitutivne enačbe

Do sedaj izpeljane enačbe veljajo za poljuben kontinuum, naj bo to tekočina, plastična ali elastična trdnina. V tem razdelku si bomo ogledali enačbe, ki definirajo obnašanje našega kontinuma kot elastične trdnine preko posplošitve Hookovega zakona, ki povezuje napetosti v materialu z njegovo deformacijo. Naučili smo se že, kako izražamo napetost, sedaj pa si poglejmo, kako merimo deformacijo teles. Tukaj bomo tudi privzeli, da je referenčna konfiguracija kar prostorska konfiguracija na začetku gibanja.

2.4.1 Mera deformacije

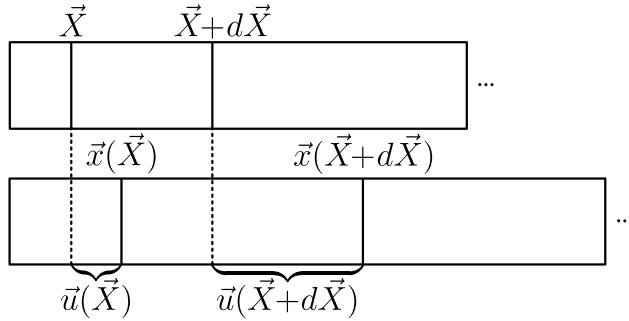
Definicija 2.22 (Gradient deformacije). Količino $F = \frac{\partial \vec{x}}{\partial \vec{X}}(t, \vec{X}) = \operatorname{Grad} \vec{x}$ imenujemo *gradient deformacije*.

Tenzor F je drugega reda in v vsaki točki predstavlja lokalno deformacijo telesa. Privzeli smo že, da je \vec{x} difeomorfizem, torej je F neizrojen, dodatno pa bomo privzeli še, da gibanje \vec{x} ohranja orientacijo, saj so taka vsa gibanja v realnosti. Od tod sledi, da je $\det F > 0$. Fizikalno interpretacijo F dobimo z naslednjim Taylorjevim razvojem:

$$d\vec{x} := \vec{x}(t, \vec{X} + d\vec{X}) - \vec{x}(t, \vec{X}) = F d\vec{X} + O(d\vec{X}^2). \quad (2.29)$$

Tenzor F do prvega reda natančno opiše, kako se vektor $d\vec{X}$ iz referenčne konfiguracije deformira v vektor $d\vec{x}$ v prostorski konfiguraciji. Vendar, ni celotna deformacija, ki jo opisuje tenzor F tako, da bi povzročala napetosti. Hookov zakon v eni dimenziji pravi, da je sila sorazmerna raztezku. Če imamo opravka s togim premikom $\vec{X} \mapsto Q\vec{X} + a$, ta premik ne bo povzročil nobene napetosti, saj se bo telo samo premaknilo, ne pa raztegnilo.

Primer 2.23. Naj bo dana tanka palica kot na sliki 4, ki jo raztegnemo vzdolž njene nosilke. Premik v točki \vec{X} je $\vec{u}(\vec{X}) = \vec{x} - \vec{X}$, v točki $\vec{X} + d\vec{X}$ pa $\vec{u}(\vec{X} + d\vec{X})$.



Slika 4: Razteg tanke palice vzdolž njene osi.

Dejanski relativni raztezek delčka palice dolžine $d\vec{X}$ je

$$\begin{aligned} \frac{d\vec{x} - d\vec{X}}{d\vec{X}} &= \frac{\vec{x}(\vec{X} + d\vec{X}) - \vec{x}(\vec{X}) - (\vec{X} + d\vec{X} - \vec{X})}{d\vec{X}} = \\ &= \frac{\vec{u}(\vec{X} + d\vec{X}) - \vec{u}(\vec{X})}{d\vec{X}} = \vec{u}'(\vec{X}) + O(d\vec{X}). \end{aligned} \quad (2.30)$$

Do prvega reda je relativni raztezek v točki \vec{X} enak kar $\vec{u}'(\vec{X})$.

Preverimo še, da ta mera raztezka zadošča nekaterim intuitivnim zahtevam. Za togi premik $\vec{X} \mapsto \vec{X} + c$ velja $\vec{u}(\vec{X}) = c$ in $\vec{u}'(\vec{X}) \equiv 0$, kot pričakovano. Za enakomerni razteg $\vec{X} \mapsto a\vec{X}$ je $\vec{u}(\vec{X}) = a\vec{X} - \vec{X}$ in $\vec{u}'(\vec{X}) = a - 1$. Tudi to ustrezza pričakovanjem, saj za $a = 1$ ni raztezka, za $a < 1$ je to skrčitev in je raztezek negativen, za $a > 1$ pa je raztezek pozitiven.

Posplošitev mere deformacije med točkama \vec{X} in $\vec{X} + d\vec{X}$ v višje dimenzije bi bila

$$\varepsilon_1 = \frac{\|d\vec{x}\| - \|d\vec{X}\|}{\|d\vec{X}\|}. \quad (2.31)$$

Toda veliko lažje je računati s kvadrami norm kot z normami vektorjev, zato je bolj primerna Cauchyjeva mera

$$\varepsilon_2 = \frac{\|d\vec{x}\|^2 - \|d\vec{X}\|^2}{\|d\vec{X}\|^2}. \quad (2.32)$$

Ker velja $\varepsilon_2 = (1 + \varepsilon_1)^2 - 1 = 2\varepsilon_1 + \varepsilon_1^2$, je za majhne pomike $\varepsilon_2 \approx 2\varepsilon_1$. Meri sta za majhne pomike torej ekvivalentni.

Izračunajmo infinitezimalno aproksimacijo mere ε_2 .

$$\begin{aligned}\varepsilon_2 &= \frac{\|d\vec{x}\|^2 - \|d\vec{X}\|^2}{\|d\vec{X}\|^2} = \frac{\|d\vec{x}\|^2}{\|d\vec{X}\|^2} - 1 = \frac{\|Fd\vec{X} + O(d\vec{X}^2)\|^2}{\|d\vec{X}\|^2} - 1 = \\ &= \frac{d\vec{X}^\top F^\top F d\vec{X} + O(d\vec{X}^3)}{\|d\vec{X}\|^2} - 1 = \frac{d\vec{X}}{\|d\vec{X}\|} \underbrace{(F^\top F - I)}_{2E} \frac{d\vec{X}}{\|d\vec{X}\|} + O(d\vec{X})\end{aligned}\quad (2.33)$$

V limiti $d\vec{X} \rightarrow 0$ lahko mero ε_2 predstavimo s tenzorjem $F^\top F - I$, mero ε_1 pa s tenzorjem E .

Definicija 2.24 (Deformacijski tenzor). Količina

$$E = \frac{1}{2}(F^\top F - I) \quad (2.34)$$

se imenuje (Cauchy-Greenov) deformacijski tenzor.

Trditev 2.25. *Deformacijski tenzor E lahko izrazimo z gradientom pomika kot*

$$E = \frac{1}{2}(\text{Grad } \vec{u} + \text{Grad } \vec{u}^\top + \text{Grad } \vec{u}^\top \text{Grad } \vec{u}). \quad (2.35)$$

Dokaz. Trditev pokaže preprost račun. Spomnimo se, da je pomik definiran kot $\vec{u}(\vec{X}) = \vec{x}(\vec{X}) - \vec{X}$ in je gradient pomika enak

$$\text{Grad } \vec{u}(\vec{X}) = \frac{\partial \vec{u}}{\partial \vec{X}} = \frac{\partial \vec{x}}{\partial \vec{X}} - I = F - I.$$

Izračunamo:

$$\begin{aligned}\frac{1}{2}(\text{Grad } \vec{u} + \text{Grad } \vec{u}^\top + \text{Grad } \vec{u}^\top \text{Grad } \vec{u}) &= \\ &= \frac{1}{2}(F - I + F^\top - I + (F - I)^\top(F - I)) = \\ &= \frac{1}{2}(F + F^\top - 2I + F^\top F - F - F^\top + I) = \\ &= \frac{1}{2}(F^\top F - I) = \\ &= E.\end{aligned}$$

□

Deformacijo F lahko s pomočjo polarnega razcepa zapišemo kot $F = RU$, kjer je R ortogonalna in U pozitivno definitna matrika. Tenzor R predstavlja rotacijo, U pa razteg in strig. Predstavljamo si lahko, da deformacijo F izvedemo tako, da najprej telo v koordinatnem sistemu, v katerem je U diagonalna, raztegnemo, nato pa zavrtimo v končno lego.

Rotacija R ne vpliva na obrabo in deformacijo materiala, saj je pomik tog in ne povzroča notranjih napetosti. Prava mera deformacije, ki vpliva na napetost v materialu, torej ne vsebuje nobenih rotacij. Za mero deformacije bi lahko vzeli kar razliko $U - I$, toda polarni razcep matrike je težko izračunati. Poglejmo si, kakšno zvezo ima zgoraj definirani deformacijski tenzor E z U :

$$E = \frac{1}{2}(F^\top F - I) = \frac{1}{2}(U^\top R^\top RU - I) = \frac{1}{2}(U^2 - I). \quad (2.36)$$

Vidimo, da E meri razliko $U^2 - I$, torej nam pove, kako se naše gibanje razlikuje od identitete, ki predstavlja gibanje brez deformacij. Kot pri enodimensionalnem primeru si oglejmo, kako se E obnaša pri enostavnih deformacijah.

Primer 2.26. Pokažimo, da je za toge deformacije $E = 0$. Naj bo deformacija toga, torej oblike $\vec{X} \mapsto Q\vec{X} + c$ z ortogonalno konstantno matriko Q in konstantnim c . Tedaj je

$$E = \frac{1}{2}(F^\top F - I) = \frac{1}{2}(Q^\top Q - I) = \frac{1}{2}(I - I) = 0. \quad (2.37)$$

Oglejmo si še enostavni razteg v smeri osi, dan kot $\vec{X} \mapsto \text{diag}(\lambda_1, \lambda_2, \lambda_3)\vec{X}$. V tem primeru velja $F = \text{diag}(\lambda_1, \lambda_2, \lambda_3)$ in

$$E = \text{diag}\left(\frac{\lambda_1^2 - 1}{2}, \frac{\lambda_2^2 - 1}{2}, \frac{\lambda_3^2 - 1}{2}\right). \quad (2.38)$$

Če so $\lambda_i = 1$, je E res 0, sicer pa so v njegovih diagonalnih komponentah zapisani raztezki vzdolž posameznih osi.

Opomba 2.27. Velja tudi obrat trditve, da je za toge transformacije E enak nič, in posledično ekvivalenca, da je $E = 0$ natanko tedaj, ko je transformacija toga. Ta ekvivalenca je dokazana za primer infinitezimalnega deformacijskega tenzorja v trditvi 2.30.

V teoriji linearne elastičnosti se bomo ukvarjali z majhnimi pomiki in majhnimi gradienti pomikov. Zato poenostavimo deformacijski tenzor z geometrijsko linearizacijo: zanemarimo člen $\text{Grad } \vec{u}^\top \text{Grad } \vec{u}$.

Definicija 2.28 (Infinitezimalni deformacijski tenzor). Količino

$$\varepsilon = \frac{1}{2}(\text{Grad } \vec{u} + \text{Grad } \vec{u}^\top) \quad (2.39)$$

imenujemo *infinitezimalni deformacijski tenzor*.

Opomba 2.29. Infinitezimalni deformacijski tenzor $\varepsilon = \frac{1}{2}(\text{Grad } \vec{u} + \text{Grad } \vec{u}^\top)$ smo dobili kot geometrijsko linearizacijo deformacijskega tenzorja (2.35) tako da smo v izrazu

$$E = \frac{1}{2} \left(\text{Grad } \vec{u} + \text{Grad } \vec{u}^\top + \overbrace{\text{Grad } \vec{u}^\top \text{Grad } \vec{u}} \right) \quad (2.40)$$

zanemarili kvadratni člen $\text{Grad } \vec{u}^\top \text{Grad } \vec{u}$.

Pokažimo še naslednjo karakterizacijo, ki pove, da se tudi ε za toge pomike obnaša kot pričakovano. Dokaz je povzet po [Gur82, str. 56].

Trditev 2.30. *Infinitezimalni deformacijski tenzor ε je ničeln natanko tedaj, ko je*

$$\vec{u} = \vec{a} + \vec{b} \times \vec{X}, \quad (2.41)$$

za konstantna vektorja \vec{a} in \vec{b} .

Dokaz. Iz $\varepsilon = 0$ direktno sledi $\text{Grad } u = -\text{Grad } u^\top$, torej je $\text{Grad } u$ antisimetričen. Naj bo Ω odprta konveksna množica $V \setminus \mathcal{B}$. Izberimo poljubni točki $\vec{X}_1, \vec{X}_2 \in \Omega$ in daljico med njima enakomerno parametriziramo, tako da je $\gamma(1) = \vec{X}_1, \gamma(0) = \vec{X}_2, \dot{\gamma}(t) = \vec{X}_1 - \vec{X}_2$. Integrirajmo $\text{Grad } \vec{u}$ po γ :

$$\vec{u}(\vec{X}_1) - \vec{u}(\vec{X}_2) = \int_0^1 (\text{Grad } \vec{u}(\gamma(t))) \dot{\gamma}(t) dt = \int_0^1 \text{Grad } \vec{u}(\gamma(t)) (\vec{X}_1 - \vec{X}_2) dt.$$

Če zgornjo enakost skalarno pomnožimo z $\vec{X}_1 - \vec{X}_2$, zaradi antisimetričnosti $\text{Grad } \vec{u}$ dobimo

$$(\vec{X}_1 - \vec{X}_2) \cdot (\vec{u}(\vec{X}_1) - \vec{u}(\vec{X}_2)) = \int_0^1 \underbrace{(\vec{X}_1 - \vec{X}_2) \cdot \text{Grad } \vec{u}(\gamma(t))(\vec{X}_1 - \vec{X}_2)}_{=0} dt = 0.$$

Videli smo torej, da za vsaka \vec{X}_1 in \vec{X}_2 velja $(\vec{X}_1 - \vec{X}_2) \cdot (\vec{u}(\vec{X}_1) - \vec{u}(\vec{X}_2)) = 0$. Odvajajmo ta izraz po \vec{X}_1 :

$$\vec{u}(\vec{X}_1) - \vec{u}(\vec{X}_2) + \text{Grad } \vec{u}(\vec{X}_1)^T (\vec{X}_1 - \vec{X}_2) = 0$$

in če ga odvajamo še po \vec{X}_2 , dobimo

$$-\text{Grad } \vec{u}(\vec{X}_2) - \text{Grad } \vec{u}(\vec{X}_1)^T = 0.$$

Upoštevamo še antisimetričnost $\text{Grad } \vec{u}$ in dobimo, da je

$$\text{Grad } \vec{u}(\vec{X}_2) = \text{Grad } \vec{u}(\vec{X}_1).$$

Ker lahko telo \mathcal{B} pokrijemo z odprtimi konveksnimi množicami (na primer kroglami), je $\text{Grad } \vec{u}$ konstanten. Pomik \vec{u} je torej oblike $\vec{u} = \vec{a} + (\text{Grad } \vec{u})\vec{X}$. Ker je $\text{Grad } \vec{u}$ antisimetričen, ga lahko zapišemo kot vektorski produkt z njegovim osnim vektorjem in dobimo $\vec{u} = \vec{a} + \vec{b} \times \vec{X}$. \square

2.4.2 Hookov zakon

Potrebujemo še relacijo med deformacijo in napetostjo, ki jo v splošnem zapišemo kot $\sigma = f(\varepsilon)$, predpostavimo torej, da je napetost odvisna samo od deformacije. V teoriji linearne elastičnosti predpostavimo, da je zveza f med napetostjo in deformacijo linearna in jo imenujemo Hookov zakon, saj je posplošitev običajnega Hookovega zakona za vzmet.

Aksiom 5 (Hookov zakon). Napetost je linearno odvisna od deformacije, preko tenzorja četrtega reda C :

$$\sigma = C : \varepsilon. \quad (2.42)$$

Tenzor C se imenuje *tenzor elastičnosti* ali *togostni tenzor* (*angl. stiffness tensor*) in ima v splošnem $3^4 = 81$ prostih parametrov.

Opomba 2.31. Aksiom 5 se po komponentah glasi $\sigma_{ij} = C_{ijkl}\varepsilon_{kl}$.

Na srečo v splošnem C nima 81 prostih komponent. Iz trditve 2.21 o simetričnosti σ sledi, da lahko prosto zamenjamo indeksa i in j , zato velja $C_{ijkl} = C_{jikl}$. Podobno iz simetričnosti ε sledi, da je $C_{ijkl} = C_{ijlk}$. S tem smo C reducirali na $6^2 = 36$ komponent.

Dodatno lahko zmanjšamo število neodvisnih komponent C , če predpostavimo, da zveza med deformacijo in napetostjo izvira iz nekega potenciala $U = U(\varepsilon)$, ki mu pravimo *gostota energije deformacije*. V tem primeru je zveza med σ in ε dana z

$$\sigma = \frac{\partial U(\varepsilon)}{\partial \varepsilon}. \quad (2.43)$$

Materialom, pri katerih je zvezo med napetostjo in deformacijo možno opisati z gostoto energije deformacije, pravimo *hiperelastični materiali*. Če predpostavimo hiperelastičnost obravnavanega materiala, vidimo, da je

$$\frac{\partial U}{\partial \varepsilon_{ij}} = \sigma_{ij} = C_{ijkl} \varepsilon_{kl}. \quad (2.44)$$

Odvajamo še enkrat po ε_{kl} in dobimo

$$\frac{\partial U}{\partial \varepsilon_{ij} \varepsilon_{kl}} = C_{ijkl}. \quad (2.45)$$

Vrstni red drugih odvodov ni pomemben, zato je $C_{ijkl} = C_{klij}$. Velja tudi obratno: če želimo, da je linearne elastičen material hiperelastičen, mora veljati $C_{ijkl} = C_{klij}$. Od sedaj naprej predpostavimo, da so vsi materiali hiperelastični, s čimer smo C reducirali na 21 komponent. Gostota energije deformacije je v tem primeru dana kar s kvadratno formo

$$U = \frac{1}{2} \varepsilon : C : \varepsilon. \quad (2.46)$$

Tenzor C se dodatno poenostavi, če predpostavimo, da je material *izotropen*, torej "enak v vse smeri". Izotropni tenzorji so pomembni objekti v mehaniki kontinuuma in so dobro raziskani. V dveh dimenzijah je edini (linearne neodvisen) izotropen tenzor identiteta δ_{ij} , v treh dimenzijah je to permutacijski tenzor ε_{ijk} , splošen izotropen tenzor četrtega reda pa je oblike

$$C_{ijkl} = \lambda \delta_{ij} \delta_{kl} + \mu \delta_{ik} \delta_{jl} + \kappa \delta_{il} \delta_{jk}, \quad (2.47)$$

kjer so λ, μ in κ splošni skalarji. V [KF75] so karakterizirani vsi izotropni tenzorji do reda 8, število izotropnih tenzorjev dimenzije n pa je navedeno kot vnos št. A005043 v spletni enciklopediji celoštivljskih zaporedij [Slo91].

Vidimo, da je vsak izotropen linearne elastičen material nujno hiperelastičen, saj je simetrija $C_{ijkl} = C_{klij}$ izpolnjena že v definiciji (2.47). Ostalima dvema simetrijam je zadoščeno le, če za izotropen C velja $\kappa = \lambda$. V tem primeru zvezo $\sigma = C : \varepsilon$ za tak tenzor zapišemo kot

$$\sigma_{ij} = C_{ijkl} \varepsilon_{kl} = \lambda \delta_{ij} \delta_{kl} \varepsilon_{kl} + \mu (\delta_{ik} \delta_{jl} + \delta_{il} \delta_{jk}) \varepsilon_{kl}, \quad (2.48)$$

kar lahko brezkoordinatno zapišemo kot

$$\sigma = \lambda (\operatorname{tr} \varepsilon) I + 2\mu \varepsilon. \quad (2.49)$$

S tem smo preostalih 21 prostih parametrov reducirali na dva parametra, λ in μ , ki ju imenujemo Laméjevi konstanti [Sla12, str. 211]. Običajno so materiali tudi homogeni, torej snovne konstante niso krajevno odvisne. Navkljub privzetim predpostavkam lahko samo z dvema konstantama dobro opišemo kar nekaj materialov, kot na primer železo, aluminij in ostale kovine, zlitine, kot npr. jeklo, ter tudi steklo. Primer anizotropnega materiala, ki ga na tak način ne moremo opisati, je les, saj zradi letnic ni enak v vseh smereh. V strojniških priročnikih najdemo snovne lastnosti bolj pogosto opisane s parametrom E in ν , ki jima pravimo Youngov (prožnostni)

modul in Poissonovo razmerje. Zveza med njima in Laméjevima parametroma je dana z

$$E = \frac{\mu(3\lambda + 2\mu)}{\lambda + \mu}, \quad \nu = \frac{\lambda}{2(\lambda + \mu)}. \quad (2.50)$$

Pogosto najdemo tudi druge snovne parametre, kot na primer strižni modul ali stisljivost. Tabela pretvorb med različnimi parametri je na voljo v [Sla12, tabela 5.1, str. 215]. Teoretična omejitev za parametre je dana s tem, da zahtevamo pozitivno definitnost energije U kot kvadratne forme, ki jo bomo potrebovali tudi v izreku 2.39 o enoličnosti in obstoju rešitev. Energija je pozitivno definitna, če je $\lambda, \mu > 0$ ali pa je, ekvivalentno, $E > 0$ in $-1 < \nu < \frac{1}{2}$. Poissonovo razmerje predstavlja razmerje med tem, koliko se telo pri raztegu v eni dimenziji skrči v drugi. Materiali, ki imajo Poissonovo razmerje negativno, se ob raztegu v eni dimenziji raztegnejo tudi v drugi, se imenujejo *auksetični* materiali (*angl. auxetic materials*). Eden prvih auksetičnih materialov je bil sintetiziran leta 1987 [Lak87], raziskave na tem področju pa so aktivne še danes.

V praksi se izkaže, da pozitivni definitnosti brez težav zadostimo. Vrednosti parametrov za nekaj pogostih materialov so podane v tabeli 1 in jih lahko najdemo v primerem fizikalnem ali strojniškem priročniku, npr. v [Cam03, str. 11].

material	E [GPa]	ν
jeklo	189 – 216	0.30
aluminij	68 – 82	0.33
steklo	61 – 110	0.18 – 0.30
beton	25 – 38	0.10 - 0.20

Tabela 1: Vrednosti elastičnih parametrov za pogoste materiale.

2.5 Navierova enačba

Sedaj imamo vse pripravljeno za opis teorije linearne elastičnosti:

- (1) mera deformacije: $\varepsilon = \frac{1}{2}(\text{Grad } \vec{u} + \text{Grad } \vec{u}^T)$,
- (2) konstitutivna zveza: $\sigma = C : \varepsilon$ in
- (3) gibalna enačba: $\rho \frac{D\vec{v}}{Dt} = \vec{f} + \text{div } \sigma$.

Naredimo še nekaj poenostavitev. Ker so gradienti pomikov majhni, za poljubno količino φ velja

$$\text{Grad } \varphi = \frac{\partial \varphi}{\partial \vec{X}} = \frac{\partial \varphi}{\partial \vec{x}} \frac{\partial \vec{x}}{\partial \vec{X}} = \text{grad } \varphi (I + \text{Grad } u) \approx \text{grad } \varphi. \quad (2.51)$$

Zaradi majhnih gradientov pomika je vseeno, ali uporabljam odvode glede na prostorski ali glede na referenčni položaj. Enak argument lahko uporabimo tudi za divergenco in običajne odvode.

Cauchyjevo gibalno enačbo lahko namesto v prostorski zapišemo tudi v referenčni konfiguraciji. Pri tem lahko zaradi majhnih gradientov pomikov namesto div pišemo Div in Cauchyjev napetostni tenzor enačimo z napetostnim tenzorjem, zapisanim v

referenčnem sistemu. Opustimo tudi strogo ločevanje referenčnega in prostorskega položaja, saj bodo od sedaj naprej enačbe vedno zapisane v referenčni konfiguraciji. Odvode po času zato pišimo kar s piko, za operatorje pa uporabljam običajne male črke ali pa kar ∇ .

S temi poenostavtvami dobimo *linearizirano enačbo gibanja*, ki pravi

$$\rho \ddot{\vec{u}}(t, \vec{X}) = \vec{f}(t, \vec{X}) + \operatorname{div} \sigma(t, \vec{X}). \quad (2.52)$$

Pripravljeno imamo vse, da napišemo končno enačbo, ki opisuje deformacijo materiala.

Izrek 2.32 (Navierova enačba). Če so gradienti pomikov v linearno elastičnem izotropnem homogenem nepolarnem kontinuumu majhni, potem za njih velja Navierova enačba

$$\rho \ddot{\vec{u}} = \vec{f} + (\lambda + \mu) \operatorname{grad} \operatorname{div} \vec{u} + \mu \Delta \vec{u}. \quad (2.53)$$

Dokaz. Od linearne enačbe gibanja (2.52) do Navierove enačbe nas loči le še izračun $\operatorname{div} \sigma$. Za primerjavo naredimo izračun koordinatno in nekoordinatno. Začnimo s koordinatnim:

$$\begin{aligned} \varepsilon_{ij} &= \frac{1}{2}(\operatorname{grad} u)_{ij} + \frac{1}{2}(\operatorname{grad} u^\top)_{ij} = \frac{1}{2}u_{i,j} + \frac{1}{2}u_{j,i} \\ \sigma_{ij} &= \lambda \varepsilon_{kk} \delta_{ij} + 2\mu \varepsilon_{ij} = \lambda \frac{1}{2}(u_{k,k} + u_{k,k}) \delta_{ij} + 2\mu \left(\frac{1}{2}u_{i,j} + \frac{1}{2}u_{j,i} \right) = \\ &= \lambda u_{k,k} \delta_{ij} + \mu(u_{i,j} + u_{j,i}) \\ (\operatorname{div} \sigma)_i &= \sigma_{ij,j} = \lambda u_{k,kj} \delta_{ij} + \mu(u_{i,jj} + u_{j,ij}) = \lambda u_{k,ki} + \mu(u_{i,kk} + u_{k,ki}) = \\ &= (\lambda + \mu)u_{k,ki} + \mu u_{i,kk} = (\lambda + \mu)(\operatorname{grad}(u_{k,k}))_i + \mu(\Delta u)_i = \\ &= ((\lambda + \mu) \operatorname{grad} \operatorname{div} u + \mu \Delta u)_i. \end{aligned}$$

Nekoordinatni dokaz pa potrebuje nekaj dodatnih relacij iz razdelka 1.2 glede diferencialnih operatorjev, ki jih uporabimo v spodnjem izračunu:

$$\begin{aligned} \operatorname{div} \sigma &= \lambda \operatorname{div} \operatorname{tr}(\varepsilon) I + 2\mu \operatorname{div} \varepsilon = \\ &= \lambda \frac{1}{2}(\operatorname{div}((\operatorname{tr} \operatorname{grad} u) I) + \operatorname{div}((\operatorname{tr} \operatorname{grad} u^\top) I)) + \mu \operatorname{div}(\operatorname{grad} u + \operatorname{grad} u^\top) = \\ &= \frac{\lambda}{2}(\operatorname{div}((\operatorname{div} u) I) + \operatorname{div}((\operatorname{div} u) I)) + \mu \Delta u + \mu \operatorname{grad} \operatorname{div} u = \\ &= \lambda \operatorname{grad} \operatorname{div} u + \mu \operatorname{grad} \operatorname{div} u + \Delta u. \end{aligned}$$

Izračun da enak rezultat kot prej in izrek je ponovno dokazan. \square

Opomba 2.33. V stacionarnem stanju $\dot{\vec{u}} = 0$, se Navierova enačba poenostavi v

$$(\lambda + \mu) \operatorname{grad} \operatorname{div} \vec{u} + \mu \Delta \vec{u} + \vec{f} = 0 \quad (2.54)$$

in jo imenujemo *stacionarna Navierova enačba*.

Opomba 2.34. Bolj splošno obliko Navierove enačbe, ki dopušča tudi bolj splošen C ali prostorsko odvisnost parametrov λ in μ , dobimo tako, da preprosto vstavimo definicijo σ v linearizirano Cauchyjevo momentno enačbo. S tem dobimo enačbo

$$\rho \ddot{\vec{u}} = \frac{1}{2} \operatorname{div}(C : (\operatorname{grad} \vec{u} + \operatorname{grad} \vec{u}^\top)) + \vec{f}. \quad (2.55)$$

Opomba 2.35. Teorija linearne elastičnosti se imenuje *linearna* zaradi dveh linearizacij, narejenih v izpeljavi Navierove enačbe. Prva je linearizacija deformacijskega tenzorja, izpostavljena v opombi 2.29, kjer v deformacijskem tenzorju E zanemarimo kvadratni člen in E zamenjamo z infinitezimalnim deformacijskim tenzorjem ε . Druga linearizacija pa je sam Hookov zakon (2.42), kjer predpostavimo linearno odvisnost napetosti od deformacije.

2.6 Obstoj in enoličnost rešitve

Navierova enačba je linearna parcialna diferencialna enačba drugega reda. Obstoj in enoličnost rešitve bomo pokazali za stacionarno Navierovo enačbo (2.54) z mešanimi robnimi pogoji, pri katerih je na nekem delu roba predpisani pomik, na drugem delu pa površinska sila.

Enačba je linearna, zato je ideja dokaza obstoja in enoličnosti, da rešitev enačbe zapišemo kot linearen funkcional na primerem prostoru in nato uporabimo Riezsov reprezentacijski izrek 1.19, ki nam bo dal obstoj in enoličnost rešitve. Ideja dokaza je vzeta iz [LC09, izrek 3.17.1, str. 232]. Iz izkušenj vemo, da bo rešitev obstajala in bo enolična kvečemu v šibkem smislu, zato prevedimo enačbo v šibko obliko in definirajmo šibko rešitev. Začeli bomo bolj splošno in prevedli stacionarno linearizirano enačbo gibanja (2.52) v šibko obliko, nato pa nadaljevali do stacionarne Navierove enačbe. Šibko obliko enačbe dobimo tako, da jo pomnožimo s splošno funkcijo \vec{v} , integriramo in preko integracije *per partes* ali drugih izrekov znižamo red odvoda na funkciji, ki jo iščemo.

Trditev 2.36. Šibka oblika linearne stacionarne enačbe gibanja $\operatorname{div} \sigma + \vec{f} = 0$ je

$$\int_{\Omega} \sigma : \varepsilon(\vec{v}) dV - \int_{\partial\Omega} \vec{t} \cdot \vec{v} dS - \int_{\Omega} \vec{f} \cdot \vec{v} dV = 0. \quad (2.56)$$

Dokaz. Pomnožimo enačbo $\operatorname{div} \sigma + \vec{f} = 0$ skalarno z $\vec{v} \in [L^2(\Omega)]^3$ in integrirajmo po Ω ter poskušajmo prenести odvode iz σ na \vec{v} . Dobimo

$$\int_{\Omega} (\vec{v} \cdot \operatorname{div} \sigma + \vec{v} \cdot \vec{f}) dV = 0.$$

Sedaj obrnemo relacijo iz trditve 1.10 ter upoštevamo simetričnost σ in dobimo $\vec{v} \cdot \operatorname{div} \sigma = \operatorname{div}(\sigma \vec{v}) - \sigma : \operatorname{grad} \vec{v}$. Ker je σ simetričen in pravokoten na antisimetrične tenzorje, je vseeno, če pri $\operatorname{grad} \vec{v}$ upoštevamo samo simetrični del, $\varepsilon(\vec{v}) = \frac{1}{2}(\operatorname{grad} \vec{v} + \operatorname{grad} \vec{v}^T)$. Nadaljujemo z računom in dobimo

$$\begin{aligned} & \int_{\Omega} (\operatorname{div}(\sigma \vec{v}) - \sigma : \varepsilon(\vec{v})) dV + \int_{\Omega} \vec{v} \cdot \vec{f} dV = 0, \\ & \int_{\partial\Omega} \sigma \vec{v} \cdot \vec{n} dS - \int_{\Omega} \sigma : \varepsilon(\vec{v}) dV + \int_{\Omega} \vec{v} \cdot \vec{f} dV = 0, \\ & \int_{\partial\Omega} \vec{v} \cdot \vec{t} dS - \int_{\Omega} \sigma : \varepsilon(\vec{v}) dV + \int_{\Omega} \vec{v} \cdot \vec{f} dV = 0. \end{aligned} \quad \square$$

Pri Navierovi enačbi upoštevamo še relacijo $\sigma = C : \varepsilon$, s pomočjo katere definiramo šibko rešitev. Najprej pa si natančno oglejmo problem in prostor, v katerem bomo rešitve iskali. Problem zastavimo za splošno stacionarno obliko Navierove

enačbe (2.55), pri čemer predpostavimo le pozitivno definitnost C . Najti želimo \vec{u} , ki zadošča

$$\begin{aligned} \operatorname{div}(C : \varepsilon(\vec{u})) + \vec{f} &= 0 && \text{na } \Omega \\ \vec{u} &= 0 && \text{na } S \subseteq \partial\Omega \\ \sigma \vec{n} = (C : \varepsilon(\vec{u})) \vec{n} &= \vec{t} && \text{na } \partial\Omega \setminus S, \end{aligned} \quad (2.57)$$

pri čemer je Ω povezana omejena odprta množica z odsekoma gladkim robom, S nek kos $\partial\Omega$ z odsekoma gladkim robom in \vec{n} enotska zunanja normala za $\partial\Omega$. Klasično rešitve iščemo v podprostoru dvakrat zvezno odvedljivih funkcij $C^2(\Omega, \mathbb{R}^3)$, za katere velja $\vec{u}|_S = 0$. Označimo ta prostor s C_S^2 in na njem definirajmo bilinearno formo

$$\langle \vec{u}, \vec{v} \rangle_E = \int_{\Omega} \varepsilon(\vec{u}) : C : \varepsilon(\vec{v}) dV. \quad (2.58)$$

Trditve 2.37. *Bilinearna forma (2.58) določa skalarni produkt na C_S^2 .*

Dokaz. Simetričnost in bilinearnost sta očitni iz lastnosti integrala. Iz pozitivne definitnosti integrala in tenzorja C sledi, da je $\varepsilon(\vec{u}) = 0$. Od tod po trditvi 2.30 sledi, da je \vec{u} oblike $\vec{u} = \vec{a} + \vec{b} \times \vec{x}$. Zaradi ničelnosti na delu roba mora biti $\vec{u} \equiv 0$. \square

Prostor C_S^2 ni poln, podobno kot prostor zveznih funkcij ni poln v L^p . Označimo njegovo napolnitev v normi $\|\cdot\|_E$ z E . Sedaj imamo vse pripravljeno za definicijo šibke rešitve problema 2.57.

Definicija 2.38. Rešitev $\vec{u} \in E$ je šibka rešitev problema 2.57, če za vsako funkcijo $\vec{v} \in E$ velja

$$\int_{\Omega} \varepsilon(\vec{u}) : C : \varepsilon(\vec{v}) dV - \int_{\partial\Omega \setminus S} \vec{t} \cdot \vec{v} dS - \int_{\Omega} \vec{f} \cdot \vec{v} dV = 0. \quad (2.59)$$

S pomočjo definicije 2.38 lahko zapišemo izrek o obstoju in enoličnosti rešitev.

Izrek 2.39. *Naj bo $\vec{f} \in [L^{\frac{6}{5}}(\Omega)]^3$ in $\vec{t} \in [L^{\frac{4}{5}}(\Omega)]^3$. Potem ima problem 2.57 natanko eno šibko rešitev v prostoru E .*

Dokaz. Kornova neenakost 1.13 pravi, da je

$$\int_{\Omega} (|\vec{u}|^2 + \|\operatorname{grad} \vec{u}\|^2) dV \leq c_1 \int_{\Omega} \varepsilon(\vec{u}) : C : \varepsilon(\vec{u}) dV.$$

Drugače prebrano zgornja neenakost pravi, da je $\|u\|_{[H^1(\Omega)]^3}^2 \leq c_1 \|u\|_E^2$. Dano imamo torej zvezno vložitev E v $[H^1(\Omega)]^3$. Na vsaki komponenti \vec{u} lahko sedaj uporabimo poseben primer (1.15) izreka o vložitvi prostorov Soboljeva 1.14 in dobimo, da vsaka komponenta \vec{u} in $|\vec{u}|$ ležijo v L^6 . Po posebnem primeru (1.16) izreku o sledi 1.16 dobimo, da na robu $\partial\Omega$ komponente in norma \vec{u} ležijo v L^4 . Dobimo verigi zveznih vložitev

$$\begin{aligned} E &\hookrightarrow [H^1(\Omega)]^3 \hookrightarrow [L^6(\Omega)]^3, \\ E &\hookrightarrow [H^1(\Omega)]^3 \hookrightarrow [L^4(\partial\Omega)]^3, \end{aligned}$$

ki prek norm na prvem in zadnjem prostoru implicirata neenakosti

$$\begin{aligned}\|\vec{u}\|_{L^6} &= \left(\int_{\Omega} |\vec{u}|^6 dV \right)^{\frac{1}{6}} \leq c_2 \|\vec{u}\|_E \\ \|\vec{u}\|_{L^4} &= \left(\int_{\partial\Omega \setminus S} |\vec{u}|^4 dV \right)^{\frac{1}{4}} \leq c_3 \|\vec{u}\|_E.\end{aligned}$$

Na šibko obliko Navierove enačbe iz definicije šibke rešitve 2.38 lahko gledamo kot na enačbo oblike

$$\langle \vec{v}, \vec{u} \rangle_E = F(\vec{v}),$$

kjer je F linearen funkcional na E , ki slika kot

$$F(\vec{v}) = \int_{\partial\Omega \setminus S} \vec{t} \cdot \vec{v} dS + \int_{\Omega} \vec{f} \cdot \vec{v} dV.$$

Prostor E je poln in opremljen s skalarnim produktom, torej je Hilbertov. Če pokažemo še, da je F zvezan, lahko zanj uporabimo Riezsov reprezentacijski izrek 1.19. Ekvivalentno je, če pokažemo omejenost F . Z uporabo trikotniške, Cauchy-Schwarzeve, Hölderjeve in zgoraj izpeljanih neenakosti dobimo

$$\begin{aligned}\left| \int_{\Omega} \vec{f} \cdot \vec{v} dV \right| &\leq \int_{\Omega} |\vec{f} \cdot \vec{v}| dV \leq \int_{\Omega} |\vec{f}| |\vec{v}| dV \leq \\ &\leq \underbrace{\left(\int_{\Omega} |\vec{f}|^{\frac{6}{5}} dV \right)^{\frac{5}{6}}}_{< \infty, \ker \vec{f} \in L^{\frac{6}{5}}(\Omega)} \underbrace{\left(\int_{\Omega} |\vec{v}|^6 dV \right)^{\frac{1}{6}}}_{\leq c_2 \|\vec{v}\|_E} \leq \\ &\leq c_1 c_2 \|\vec{v}\|_E\end{aligned}$$

in

$$\begin{aligned}\left| \int_{\partial\Omega \setminus S} \vec{t} \cdot \vec{v} dS \right| &\leq \int_{\partial\Omega \setminus S} |\vec{t} \cdot \vec{v}| dS \leq \int_{\partial\Omega \setminus S} |\vec{t}| |\vec{v}| dS \leq \\ &\leq \underbrace{\left(\int_{\partial\Omega \setminus S} |\vec{t}|^{\frac{4}{3}} dS \right)^{\frac{3}{4}}}_{< \infty, \ker \vec{t} \in L^{\frac{4}{3}}(\partial\Omega \setminus S)} \underbrace{\left(\int_{\partial\Omega \setminus S} |\vec{v}|^4 dS \right)^{\frac{1}{4}}}_{\leq c_4 \|\vec{v}\|_E} \leq \\ &\leq c_3 c_4 \|\vec{v}\|_E.\end{aligned}$$

Funkcional F je po trikotniški neenakosti omejen in zato zvezen. Po izreku 1.19 obstaja enolično določen $\vec{u}^* \in E$, tako da je

$$F(v) = \langle \vec{v}, \vec{u}^* \rangle_E$$

za vsak $\vec{v} \in E$. Šibka oblika enačbe se torej glasi

$$\langle \vec{v}, \vec{u} \rangle_E = F(\vec{v}) = \langle \vec{v}, \vec{u}^* \rangle_E.$$

Če prenesemo vse na eno stran, dobimo, da za vsak $\vec{v} \in E$ velja

$$\langle \vec{v}, \vec{u} - \vec{u}^* \rangle_E = 0.$$

Ker je $\vec{u} - \vec{u}^*$ pravokoten na vse $\vec{v} \in E$, ne preostane drugega, kot da je $\vec{u} = \vec{u}^*$ enolična rešitev v smislu definicije 2.38. \square

Samo enoličnost lahko dokažemo tudi na bolj elementaren način, ki morda nudi boljši vpogled v to, kako robni pogoji določijo enoličnost rešitve v notranjosti, hkrati pa ne potrebuje tako strogih predpostavk na \vec{f} in \vec{t} .

Izrek 2.40 (Kirchoff). *Rešitev problema 2.57 je enolična v E , če obstaja.*

Dokaz. Denimo, da imamo dve rešitvi \vec{u}_1 in \vec{u}_2 z enakimi robnimi pogoji. Ker sta tako \vec{u}_1 kot \vec{u}_2 šibki rešitvi, zadoščata

$$\begin{aligned}\int_{\Omega} \varepsilon(\vec{u}_1) : C : \varepsilon(\vec{v}) dV - \int_{\partial\Omega \setminus S} \vec{t} \cdot \vec{v} dS - \int_{\Omega} \vec{f} \cdot \vec{v} = 0, \\ \int_{\Omega} \varepsilon(\vec{u}_2) : C : \varepsilon(\vec{v}) dV - \int_{\partial\Omega \setminus S} \vec{t} \cdot \vec{v} dS - \int_{\Omega} \vec{f} \cdot \vec{v} = 0.\end{aligned}$$

Razlika $\vec{w} = \vec{u}_1 - \vec{u}_2$ torej zadošča

$$\int_{\Omega} \varepsilon(\vec{w}) : C : \varepsilon(\vec{v}) dV = 0.$$

Funkcija \vec{w} je v prostoru E , saj je na S enaka 0, zato zgornja enačba velja tudi za $\vec{w} = \vec{v}$. Zaradi pozitivne definitnosti C sledi $\varepsilon(\vec{w}) = 0$, od koder kot prej zaradi robnih pogojev sledi $\vec{w} = 0$ ozziroma $\vec{u}_1 = \vec{u}_2$. \square

Opomba 2.41. Iz enoličnosti v E sledi tudi enoličnost v klasičnem smislu, saj je prostor klasičnih rešitev podprostор v E .

2.7 Priprava na numerično reševanje

Za numerično reševanje se postavimo v kartezični koordinatni sistem. Komponente tenzorja σ razpišimo:

$$\sigma = \begin{bmatrix} \sigma_{xx} & \sigma_{xy} & \sigma_{xz} \\ \sigma_{xy} & \sigma_{yy} & \sigma_{yz} \\ \sigma_{xz} & \sigma_{yz} & \sigma_{zz} \end{bmatrix}, \quad (2.60)$$

pri čemer smo že upoštevali simetrijo. Reševati tridimenzionalne enačbe je težje kot dvodimenzionalne, zato si oglejmo dve poenostavitevi.

2.7.1 Poenostavitev na dve dimenziji

Tridimenzionalne probleme se zavoljo lažje formulacije, manjše računske zahtevnosti in lažje implementacije, pogosto prevede na nižjedimenzionalne. Spodaj sta na kratko opisani dve najbolj pogosti poenostavitevi. Bolj podroben opis je dan v [Sla12, str. 260–278].

Ravninska deformacija: Pri tej poenostavitevi predpostavimo, da nimamo pomikov v eni izmed koordinatnih smeri, torej brez škode za splošnost $u_3 = 0$ in preostali dve komponenti sta neodvisni od tretje koordinate. Posledično so $\varepsilon_{xz}, \varepsilon_{yz}$ in ε_{zz} enaki 0. Ta poenostavitev je primerna za telesa, ki imajo eno dimenzijo mnogo večjo od drugih in se vzdolž nje ne spreminja, torej za (ne nujno krožne) valje. Dobimo enačbo, ki je enaka tisti v treh dimenzijah, le da ima eno komponento manj.

Ravninska napetost: Pri tej poenostavitev predpostavimo, da napetost nima nečelnih komponent v smeri ene izmed koordinatnih osi, torej, po primerni rotaciji koordinatnega sistema, da so komponente σ_{xz} , σ_{yz} in σ_{zz} enake 0. Ta poenostavitev je primerna za telesa, ki so “tanke plošče”, torej imajo eno dimenzijo veliko manjšo od ostalih dveh. Poleg tega morajo biti vse obremenitve vzdolž plošče. Ta poenostavitev da enako enačbo kot v treh dimenzijah, le da je potrebno uporabiti druge materialne konstante: namesto λ v zvezi $\sigma = \lambda \operatorname{tr} \varepsilon I + 2\mu\varepsilon$ moramo uporabiti $\hat{\lambda} = \frac{2\mu\lambda}{\lambda+2\mu}$. Pretvorba drugih parametrov je dana v [Sla12, str. 276, tabela 7.1].

Pri obeh poenostavitevah se iz pogojev izpelje, da v Navierovi enačbi ne nastopa tretja komponenta \vec{u} . V dveh dimenzijah pišemo prvo komponento pomika z u in drugo z v , komponente položaja pa označujemo z x in y . Če si razpišemo izraze za deformacijski tenzor in napetostni tenzor za dvodimenzionalni primer, dobimo:

$$\varepsilon = \begin{bmatrix} \frac{\partial u}{\partial x} & \frac{1}{2}\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) \\ \frac{1}{2}\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) & \frac{\partial v}{\partial y} \end{bmatrix}, \quad (2.61)$$

$$\sigma = \begin{bmatrix} \lambda \frac{\partial v}{\partial y} + (\lambda + 2\mu) \frac{\partial u}{\partial x} & \mu \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \\ \mu \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) & \lambda \frac{\partial u}{\partial x} + (\lambda + 2\mu) \frac{\partial v}{\partial y} \end{bmatrix}. \quad (2.62)$$

2.7.2 Robni pogoji

Pri reševanju robnih problemov bomo imeli tri vrste robnih pogojev. Dirichletove robne pogoje $\vec{u}|_{\partial\Omega} = \vec{u}_0$ bomo uporabili, ko bomo poznali pomike na robu. Najpogosteje bo pogoj oblike $\vec{u} = 0$, ko kos robu materiala držimo pritrjen. Pri drugi vrsti robnih pogojev poznamo napetosti na robovih. Najpogosteje bomo podali kar napetost na robu $\vec{t} = \vec{t}_0$, kjer je \vec{t}_0 neka znana vrednost, npr. 0, če je ta rob prost. Tretja vrsta pogojev bodo simetrijski pogoji, ko bomo problem reducirali preko neke osi simetrije, na osi pa bomo zahtevali kompatibilnostne ali simetrijske pogoje.

Oglejmo si bolj natančno pogoje, podane z napetostjo. Napetost na robu izračunamo preko napetostnega tenzorja, $\vec{t} = \sigma \vec{n}$, kjer je \vec{n} zunanjna normala na rob. Zveza $\vec{t}_0 = \sigma \vec{n}$, zapisana v kartezičnih koordinatah, da dva pogoja

$$\begin{bmatrix} t_{0x} \\ t_{0y} \end{bmatrix} = \begin{bmatrix} \lambda \frac{\partial v}{\partial y} + (\lambda + 2\mu) \frac{\partial u}{\partial x} & \mu \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \\ \mu \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) & \lambda \frac{\partial u}{\partial x} + (\lambda + 2\mu) \frac{\partial v}{\partial y} \end{bmatrix} \begin{bmatrix} n_x \\ n_y \end{bmatrix}. \quad (2.63)$$

3 Numerična metoda

V 20. stoletju je skupaj z razvojem računalnikov začel svojo pot tudi razvoj numeričnih metod za reševanje parcialnih diferencialnih enačb. Do danes je bilo razvitih veliko metod za numerično reševanje parcialnih diferencialnih enačb. Dva pomembna razreda metod se ločita glede na obliko, v kateri rešujemo parcialno diferencialno enačbo: šibki (*angl. weak form*) ali krepki (*angl. strong form*). Najznamenitejši in zelo uspešen predstavnik prve skupine je metoda končnih elementov (MKE) (*angl. finite element method (FEM)*) [Koz08, str. 340], kjer problem najprej prevedemo v šibko obliko, nato pa rešitev poiščemo kot linearno kombinacijo baznih funkcij iz izbranega prostora. Najbolj poznan predstavnik metod, ki rešujejo problem v krepki obliki, pa je metoda končnih differenc (MKD) (*angl. finite difference method (FDM)*) [Koz08, str. 296], pri kateri direktno diskretiziramo operator, ki nastopa v enačbi.

Poleg tega se metode delijo tudi glede na tip diskretizacije domene, ki ga potrebujejo. Metoda končnih elementov potrebuje *mrežo*, nad katero deluje, tj. triangulacijo notranjosti domene, ki inducira tudi mrežo na robu. Metoda robnih elementov potrebuje mrežo samo na robu domene. Metoda končnih differenc je običajno formulirana na pravokotni mreži. Obstajajo pa tudi metode, ki mreže ne potrebujejo; imenujemo jih *brezmrežne metode* (*angl. meshless methods*) [LL02]. Metoda, predstavljena v tem razdelku bo reševala enačbo v krepki obliki in bo brezmrežna.

3.1 Izpeljava

Izpeljavo začnimo z osvežitvijo spomina na metodo končnih differenc, ki bo služila kot motivacija.

3.1.1 Ideja in motivacija

Primer 3.1. Rešujemo enodimensionalno Poissonovo enačbo. Izpeljava metode končnih differenc ne bo povsem običajna in tudi ne najkrajša možna, ampak bo narejena tako, da jo bomo lahko posplošili v brezmrežno metodo.

Rešujemo problem z mešanimi robnimi pogoji

$$\begin{aligned} u''(x) &= f(x) \quad \text{na } (a, b), \\ u(a) &= A, \\ u'(b) &= B, \end{aligned} \tag{3.1}$$

katerega rešitev poznamo v kvadraturah

$$u(x) = \int_a^x \left(\int_b^\eta f(\xi) d\xi \right) d\eta + B(x - a) + A. \tag{3.2}$$

Numeričnega reševanja se lotimo tako, da interval $[a, b]$ diskretiziramo na N enakih delov dolžine $h = \frac{b-a}{N}$ z delilnimi točkami $x_i = a + ih$, za $i = 0, \dots, N$. Za vsako od teh točk uvedemo neznanko u_i , ki predstavlja neznano funkcionalno vrednost v točki x_i . S pomočjo vrednosti u_{i-1}, u_i in u_{i+1} želimo aproksimirati $u''(x_i)$. To nam bo dalo zvezo med spremenljivkami in ko jo uporabimo za vse notranje točke

ter upoštevamo še robne pogoje, dobimo sistem linearnih enačb, katerega rešitev da aproksimacijo funkcije u .

Funkcijo u v okolini x_i aproksimiramo z interpolacijskim polinomom, njene odvode pa z odvodi interpolacijskega polinoma. Za izračun interpolacijskega polinoma zapišimo nastavek

$$\hat{u}(x) = \alpha_0 + \alpha_1 x + \alpha_2 x^2 = [1 \ x \ x^2] \begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \alpha_2 \end{bmatrix} = \mathbf{b}(x)^\top \boldsymbol{\alpha} \quad (3.3)$$

in zahtevajmo, da velja

$$\hat{u}(x_{i-1}) = u_{i-1}, \quad (3.4)$$

$$\hat{u}(x_i) = u_i, \quad (3.5)$$

$$\hat{u}(x_{i+1}) = u_{i+1}. \quad (3.6)$$

Levo stran zapišemo po definiciji (3.3) in dobimo sistem enačb za $\boldsymbol{\alpha}$

$$\alpha_0 + \alpha_1(x_i - h) + \alpha_2(x_i - h)^2 = u_{i-1}, \quad (3.7)$$

$$\alpha_0 + \alpha_1 x_i + \alpha_2 x_i^2 = u_i, \quad (3.8)$$

$$\alpha_0 + \alpha_1(x_i + h) + \alpha_2(x_i + h)^2 = u_{i+1}, \quad (3.9)$$

ozziroma v matrični obliki

$$\begin{bmatrix} 1 & x_i - h & (x_i - h)^2 \\ 1 & x_i & x_i^2 \\ 1 & x_i + h & (x_i + h)^2 \end{bmatrix} \begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \alpha_2 \end{bmatrix} = \begin{bmatrix} u_{i-1} \\ u_i \\ u_{i+1} \end{bmatrix}. \quad (3.10)$$

Sistem krajše zapišemo kar kot $B\boldsymbol{\alpha} = \mathbf{u}$. Sistem rešimo in dobimo

$$\alpha_0 = \frac{2h^2 u_i + h(u_{i-1} - u_{i+1})x_i + (u_{i-1} - 2u_i + u_{i+1})x_i^2}{2h^2}, \quad (3.11)$$

$$\alpha_1 = \frac{h(u_{i+1} - u_{i-1}) - 2(u_{i-1} - 2u_i + u_{i+1})x_i}{2h^2}, \quad (3.12)$$

$$\alpha_2 = \frac{u_{i-1} - 2u_i + u_{i+1}}{2h^2}. \quad (3.13)$$

Interpolacijski polinom skozi točke (x_i, u_i) lahko sedaj zapišemo kot

$$\begin{aligned} \hat{u}(x) &= [1 \ x \ x^2] \begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \alpha_2 \end{bmatrix} = \\ &= u_i + \frac{u_{i+1} - u_{i-1}}{2h}(x - x_i) + \frac{u_{i-1} - 2u_i + u_{i+1}}{2h^2}(x - x_i)^2. \end{aligned} \quad (3.14)$$

Členi u_j nastopajo linearno, zato lahko (3.14) napišemo tudi v obliki

$$\hat{u}(x) = \begin{bmatrix} \frac{(x_i - x)(h + x_i - x)}{2h^2} & \frac{(h + x - x_i)(h + x_i - x)}{h^2} & \frac{(x - x_i)(h + x - x_i)}{2h^2} \end{bmatrix} \begin{bmatrix} u_{i-1} \\ u_i \\ u_{i+1} \end{bmatrix} = \boldsymbol{\varphi}(x)\mathbf{u}. \quad (3.15)$$

S tem smo ločili podatke, ki se nanašajo na vrednost funkcije, od podatkov, ki se nanašajo na pozicije točk. V primerih, ko bomo večkrat potrebovali vrednost interpolacijskega polinoma v neki točki x^* za različne nabore funkcijskih vrednosti \mathbf{u} (vendar še vedno izmerjene v istih točkah), je ugodno izračunati vrednosti $\varphi(x^*)$ vnaprej in nato vrednosti interpolacijskega polinoma dobimo le s skalarnim produkтом $\hat{u}(x^*) = \varphi(x^*)\mathbf{u}$. Pri tem ni potrebno, da je x^* ena izmed točk x_i , ampak je lahko poljubna točka v domeni.

Z odvajanjem \hat{u} aproksimiramo u' in u'' . Odvode izračunamo v točki x_i in dobimo znani formuli za aproksimacijo odvodov s končnimi diferencami:

$$\hat{u}'(x_i) = \varphi'(x_i)\mathbf{u} = \begin{bmatrix} -\frac{1}{2h} & 0 & \frac{1}{2h} \end{bmatrix} \begin{bmatrix} u_{i-1} \\ u_i \\ u_{i+1} \end{bmatrix}, \quad (3.16)$$

$$\hat{u}''(x_i) = \varphi''(x_i)\mathbf{u} = \begin{bmatrix} \frac{1}{h^2} & -\frac{2}{h^2} & \frac{1}{h^2} \end{bmatrix} \begin{bmatrix} u_{i-1} \\ u_i \\ u_{i+1} \end{bmatrix}. \quad (3.17)$$

Zgornji aproksimaciji lahko uporabimo za reševanje problema (3.1). Namesto enakosti

$$u''(x_i) = f(x_i), \quad (3.18)$$

ki jo narekuje diferencialna enačba, za vsako točko x_i v notranjosti zapišemo njeni aproksimacijo

$$\hat{u}''(x_i) = \begin{bmatrix} \frac{1}{h^2} & -\frac{2}{h^2} & \frac{1}{h^2} \end{bmatrix} \begin{bmatrix} u_{i-1} \\ u_i \\ u_{i+1} \end{bmatrix} = f(x_i). \quad (3.19)$$

Dirichletov pogoj na levem robu zapišemo preprosto kot $u_0 = A$, za Neumannovega na desnem robu pa lahko uporabimo npr. enostransko diferenco na treh točkah

$$\begin{bmatrix} \frac{1}{2h} & -\frac{2}{h} & \frac{3}{2h} \end{bmatrix} \begin{bmatrix} u_{N-2} \\ u_{N-1} \\ u_N \end{bmatrix} = B, \quad (3.20)$$

ki bi jo izpeljali na enak način kot diferenco (3.16).

Vse te enakosti zložimo v sistem enačb in ga zapišimo v matrični obliki

$$\frac{1}{h^2} \begin{bmatrix} 1 & & & & \\ -1 & 2 & 1 & & \\ & -1 & 2 & 1 & \\ & & \ddots & \ddots & \ddots \\ & & & -1 & 2 & 1 \\ & & & h/2 & -2h & 3h/2 \end{bmatrix} \begin{bmatrix} u_0 \\ u_1 \\ u_2 \\ \vdots \\ u_{N-1} \\ u_N \end{bmatrix} = \begin{bmatrix} A \\ f(x_1) \\ f(x_2) \\ \vdots \\ f(x_{N-1}) \\ B \end{bmatrix}. \quad (3.21)$$

Rešitev tega sistema imenujemo numerična rešitev problema (3.1).

3.1.2 Splošna izpeljava

Postavimo se sedaj v splošnejši okvir. Rešujmo parcialno diferencialno enačbo

$$\mathcal{L}u = f \text{ na } \Omega, \quad (3.22)$$

$$\mathcal{R}u = g \text{ na } \partial\Omega,$$

kjer je $\Omega \subseteq \mathbb{R}^d$ omejena domena, torej omejena povezana odprta množica z odsekoma gladkim robom, u neznana skalarna funkcija, $\mathcal{L}: C^r(\mathbb{R}^d) \rightarrow C(\mathbb{R})$ eliptičen linearen parcialni diferencialni operator reda r in $\mathcal{R}u$ robni pogoji, pri katerih je problem enolično rešljiv.

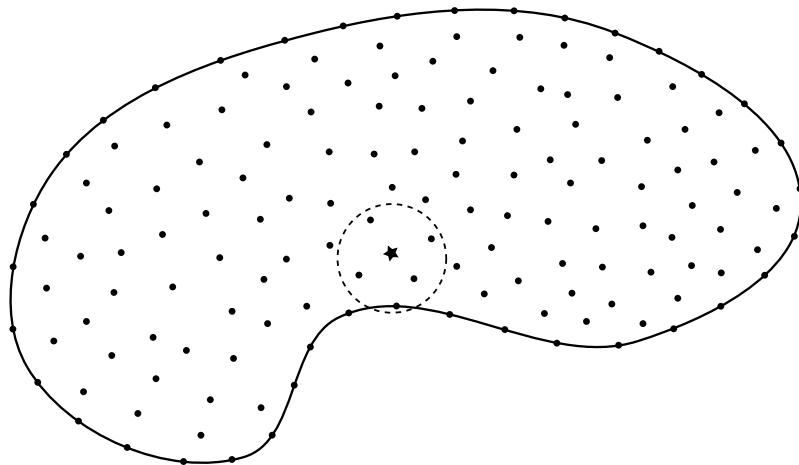
Poščimo primera diskretno obliko zgornjega zveznega problema. Izberimo N točk v zaprtju domene, $x_1, \dots, x_N \in \bar{\Omega}$, od teh naj jih Q leži v notranjosti Ω in $N - Q$ na robu. Množico vseh diskretizacijskih točk označimo z

$$\mathcal{X} = \{x_1, \dots, x_N\}. \quad (3.23)$$

Podobno kot pri končnih diferencah bomo v teh točkah aproksimirali vrednost funkcije u . Za vsako točko $p \in \bar{\Omega}$, bomo izbrali n točk iz \mathcal{X} , ki bodo sestavljale *soseščino* (*angl. support*) točke p . Število n imenujemo velikost *soseščine*. Označimo z $\mathcal{N}(p)$ *soseščino* točke p in z $\mathcal{I}(p) = (i_1, \dots, i_n)$ nabor indeksov, za katere so izbrani x_{i_j} v *soseščini* p . Velja torej

$$\mathcal{N}(p) = \{x_i; i \in \mathcal{I}(p)\}. \quad (3.24)$$

Običajno bo $n \ll N$, npr. $n = 9$ in $N = 10^6$. Primer domene Ω , izbrane točke p in njene *soseščine* je prikazan na sliki 5.



Slika 5: Primer domene z diskretnim opisom notranjosti in roba, skupaj z izbrano točko in njeno *soseščino*.

V okolini točke p aproksimirajmo u z elementi iz nekega končno dimenzionalnega prostora funkcij $\mathcal{B} = \text{Lin}\{b_1, \dots, b_m\}$. Funkcijam $b_i: \mathbb{R}^d \rightarrow \mathbb{R}$ pravimo *bazne funkcije*, številu m pa moč baze. Ni nujno, da so funkcije b_i definirane globalno, saj so lahko odvisne od izbrane točke p in njene *soseščine*. Aproksimacijo \hat{u} za u lahko torej zapišemo kot

$$u \approx \hat{u} = \sum_{i=1}^m \alpha_i b_i = \mathbf{b}^\top \boldsymbol{\alpha}, \quad (3.25)$$

pri čemer smo z $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^m$ označili vektor neznanih koeficientov in z $\mathbf{b} = (b_i)_{i=1}^m$ funkcijo $\mathbf{b}: \mathbb{R}^d \rightarrow \mathbb{R}^m$, katere komponente so bazne funkcije b_i .

Če bi poznali vrednosti $u(x_i)$ za $i \in \mathcal{I}(p)$, potem bi za aproksimiranko \hat{u} v najboljšem primeru zahtevali $\hat{u}(x_i) = u(x_i)$, za vsak $i \in \mathcal{I}(p)$. Ker funkcijskih vrednosti $u(x_i)$ ne poznamo, uvedimo spremenljivke u_i za vsako točko v domeni, ki

predstavljajo neznane prave vrednosti in nadalujmo s simbolnim računanjem. Če zahteve za interpolacijo po vrsticah zapišemo v sistem linearnih enačb, dobimo

$$\begin{bmatrix} b_1(x_{i_1}) & \cdots & b_m(x_{i_1}) \\ \vdots & \ddots & \vdots \\ b_1(x_{i_n}) & \cdots & b_m(x_{i_n}) \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_m \end{bmatrix} = \begin{bmatrix} u_{i_1} \\ \vdots \\ u_{i_n} \end{bmatrix}. \quad (3.26)$$

Na krajše sistem zapišemo kot $B\boldsymbol{\alpha} = \tilde{\mathbf{u}}$. Odvisno od n, m in b_i je ta sistem lahko poddoločen, predoločen ali običajen. Vprašanje obrnljivosti matrike B v primeru $n = m$ je težko in je odvisno od izbire funkcij in lege točk. Tudi če so funkcije b_i linearne neodvisne, obstajajo primeri že v dveh dimenzijah, ko zastavljen interpolacijski problem ni korekten [Koz08, str. 79, izrek 2.2] in zagotavljanje korektnosti v splošnem je težek problem.

V vsakem primeru lahko definiramo neke vrste rešitev sistema (3.26), za katero zahtevamo, da minimizira napako v smislu utežene diskretne 2-norme, torej, da minimizira

$$\|u - \hat{u}\|_{2, \mathcal{N}(p), \mathbf{w}} = \sum_{i \in \mathcal{I}(p)} w(p - x_i)(u_i - \hat{u}(x_i))^2, \quad (3.27)$$

pri čemer je $w: \mathbb{R}^d \rightarrow \mathbb{R}$ nenegativna funkcija, ki jo imenujemo *utež*, \mathbf{w} pa je vektor, sestavljen iz vrednosti te funkcije v točkah v sosedstvu. Če je takih rešitev $\boldsymbol{\alpha}$ več, izberimo tisto, za katero je $\|\boldsymbol{\alpha}\|$ najmanjša. Zgornjo minimizacijo lahko prevedemo na minimiziranje diskretne 2-norme $\|WB\boldsymbol{\alpha} - W\tilde{\mathbf{u}}\|_{2, \mathcal{N}(p)}$, kjer je W diagonalna matrika korenov uteži za posamezne točke, $W = \text{diag}(\sqrt{w(x_{i_1} - p)}, \dots, \sqrt{w(x_{i_n} - p)})$. Tak sistem lahko, ne glede na njegovo določenost, rešimo s pomočjo Moore-Penroseovega psevdoinverza, ki ga izračunamo s pomočjo singularnega razcepa matrike WB . Tako lahko izrazimo

$$\boldsymbol{\alpha} = (WB)^+ W \tilde{\mathbf{u}}, \quad (3.28)$$

kjer $+$ označuje Moore-Penroseov psevdoinverz. Izraz za $\boldsymbol{\alpha}$ lahko vstavimo nazaj v definicijo \hat{u} (3.25) in dobimo

$$\hat{u} = \mathbf{b}^\top \boldsymbol{\alpha} = \mathbf{b}^\top (WB)^+ W \tilde{\mathbf{u}}. \quad (3.29)$$

Sedaj lahko za izbrano točko p izračunamo

$$\hat{u}(p) = \underbrace{\mathbf{b}(p)^\top (WB)^+ W}_{\boldsymbol{\varphi}_p} \tilde{\mathbf{u}}. \quad (3.30)$$

Izračunljivi kos $\boldsymbol{\varphi}_p$ je v praksi vrstica velikosti n , matematično pa je linearen funkcional $\boldsymbol{\varphi}_p \in (\mathbb{R}^n)^*$, ki naboru funkcijskih vrednosti v sosedstvu $\mathcal{N}(p)$ privedi aproksimacijo za funkcijsko vrednost v točki p .

Podobno kot pri deljenih diferencah odvode funkcije u aproksimiramo z odvodi interpolacijskega polinoma skozi točke v sosedstvu, bomo tudi v našem primeru aproksimirali odvode funkcije u z odvodi \hat{u} ,

$$(\mathcal{L}u)(p) \approx (\mathcal{L}\hat{u})(p) = (\mathcal{L}\mathbf{b})(p)^\top (WB)^+ W \tilde{\mathbf{u}}, \quad (3.31)$$

kjer podobno kot v primeru 3.1 definiramo

$$\boldsymbol{\varphi}_{\mathcal{L}, p} = (\mathcal{L}\mathbf{b})(p)^\top (WB)^+ W. \quad (3.32)$$

Funkcional $\varphi_{\mathcal{L},p}$ je aproksimacija operatorja \mathcal{L} v točki p . Pogosto ga imenujemo tudi *funkcija oblike* (angl. shape function), saj v sebi nosi podatke o lokalni obliki domene in izboru okoliških točk ter seveda o obnašanju \mathcal{L} v tej okolici. Funkcional $\varphi_{\mathcal{L},p}$ je možno izračunati tudi, če funkcijskih vrednosti $\tilde{\mathbf{u}}$ v okolici p ne poznamo. Kasneje samo s skalarnim produktom $\varphi_{\mathcal{L},p} \tilde{\mathbf{u}}$ dobimo aproksimacijo za $(\mathcal{L}u)(p)$, lahko pa $\varphi_{\mathcal{L},p}$ izkoristimo tudi za zapis linearne enačbe

$$\varphi_{\mathcal{L},p} \tilde{\mathbf{u}} = f(p), \quad (3.33)$$

ki je direktna aproksimacija diferencialne enačbe (3.22) v točki p ,

$$(\mathcal{L}u)(p) = f(p). \quad (3.34)$$

Tu ni potrebno, da je $p \in \mathcal{N}(p)$, temveč je lahko katerakoli točka v domeni, čeprav bo najpogosteje tudi sama ena izmed diskretizacijskih točk.

Operatorja \mathcal{L} in \mathcal{R} aproksimiramo po celi domeni: za vsako diskretizacijsko točko x_i v notranjosti izračunamo $\varphi_{\mathcal{L},x_i}$, za vsako točko x_j na robu pa $\varphi_{\mathcal{R},x_j}$. Nato za vsak $x_i \in \mathcal{X}$ napišemo enačbo (3.34) in dobimo sistem enačb

$$\varphi_{\mathcal{L},x_i} \tilde{\mathbf{u}} = f(x_i), \text{ za vsak } i, \text{ tak, da je } x_i \in \Omega, \quad (3.35)$$

$$\varphi_{\mathcal{R},x_i} \tilde{\mathbf{u}} = g(x_i), \text{ za vsak } i, \text{ tak, da je } x_i \in \partial\Omega. \quad (3.36)$$

Te enačbe lahko zapišemo v matrični sistem

$$A\mathbf{u} = \mathbf{f}, \quad (3.37)$$

kjer ima matrika A v vrsticah zapisane funkcije oblike, tako da so neničelnii elementi na tistih mestih, ki se pomnožijo z neznankami, ki ustrezajo sosedom x_i . Natančneje, elementi matrike A so

$$A_{k,i_j} = (\varphi_{\mathcal{L},p})_j, \text{ za vsak } k, \text{ tak, da je } x_k \in \Omega \text{ in za vsak } i_j \in \mathcal{I}(x_k), \quad (3.38)$$

$$A_{k,i_j} = (\varphi_{\mathcal{R},p})_j, \text{ za vsak } k, \text{ tak, da je } x_k \in \partial\Omega \text{ in za vsak } i_j \in \mathcal{I}(x_k). \quad (3.39)$$

Vektor $\mathbf{u} = (u_i)_{i=1}^N$ je vektor neznanih funkcijskih vrednosti, ki ga iščemo, v vektorju \mathbf{f} pa so zapisani robni pogoji

$$\mathbf{f}_k = \begin{cases} f(x_k); & x_k \in \Omega \\ g(x_k); & x_k \in \partial\Omega \end{cases}, \text{ za } k = 1, \dots, N. \quad (3.40)$$

Matrika A je razpršena, dimenzij $N \times N$ in v vsaki vrstici ima kvečjemu n neničelnih elementov, torej je skupno število neničelnih elementov

$$\text{nnz}(A) \leq nN. \quad (3.41)$$

Enakost je lahko dosežena, lahko pa je tudi stroga, saj so kakšni koeficienti v $\varphi_{\mathcal{L},x_i}$ lahko tudi 0, kot se to zgodi pri Dirichletovih robnih pogojih.

Rešitev sistema (3.37) da aproksimacijo $u(x_i) \approx u_i$, kar vzamemo za numerično rešitev problema (3.22).

3.2 Posebni primeri

V metodi iz razdelka 3.1.2 lahko v posebnih primerih prepoznamo druge znane metode. Za začetek pokažimo enostavno trditev, ki bo metodo v določenih primerih poenostavila.

Trditev 3.2. Če je $m = n$ in je matrika B iz sistema (3.26) obrnljiva, potem je izbira uteži nepomembna.

Dokaz. Spomnimo se, da je

$$\varphi_{\mathcal{L},p} = (\mathcal{L}\mathbf{b})(p)^T (WB)^+ W.$$

Matrika W je diagonalna s samimi pozitivnimi števili na diagonali in je torej obrnljiva. Po predpostavki je obrnljiva tudi matrika B , torej je obrnljiv tudi produkt WB in velja

$$(WB)^+ = (WB)^{-1} = B^{-1}W^{-1}.$$

Od tod sledi, da je

$$\varphi_{\mathcal{L},p} = (\mathcal{L}\mathbf{b})(p)^T B^{-1} W^{-1} W = (\mathcal{L}\mathbf{b})(p)^T B^{-1},$$

kar je neodvisno od W . \square

Opomba 3.3. Čeprav trditev 3.2 velja v eksaktni aritmetiki, v praksi ne velja nujno. Če so izbrane uteži zelo majhne ali zelo različnih velikosti, lahko to povzroči nepotrebne numerične nestabilnosti. Prav tako lahko zaradi izbire uteži pri izračunu psevdoinverza v vrstici 16 v algoritmu 2 SVD razcep, uporabljen v ozadju funkcije PINV, odreže različno število singularnih vrednosti, kar seveda vpliva na rezultate.

Če je operator \mathcal{L} enostaven, kot je bil $\frac{\partial^2}{\partial x^2}$ v primeru 3.1, lahko enačbo za funkcije oblike izračunamo direktno. Kaj pa lahko storimo, če je operator bolj zapleten? Odgovor na to nudi naslednja trditev.

Trditev 3.4. Preslikava iz vektorskega prostora linearnih parcialnih diferencialnih operatorjev, ki operatorju \mathcal{L} priredi funkcijo oblike $\varphi_{\mathcal{L},p} \in (\mathbb{R}^n)^*$ v neki točki p , je homomorfizem. Za $\mathcal{L} = \alpha\mathcal{L}_1 + \beta\mathcal{L}_2$ velja torej

$$\varphi_{\alpha\mathcal{L}_1 + \beta\mathcal{L}_2, p} = \alpha\varphi_{\mathcal{L}_1, p} + \beta\varphi_{\mathcal{L}_2, p}. \quad (3.42)$$

Dokaz. Naj bo $\mathcal{L} = \alpha\mathcal{L}_1 + \beta\mathcal{L}_2$. Izračunajmo

$$\begin{aligned} \varphi_{\alpha\mathcal{L}_1 + \beta\mathcal{L}_2, p} &= ((\alpha\mathcal{L}_1 + \beta\mathcal{L}_2)\mathbf{b})(p)(WB)^+ W = \\ &= (\alpha(\mathcal{L}_1\mathbf{b})(p) + \beta(\mathcal{L}_2\mathbf{b})(p))(WB)^+ W = \\ &= (\alpha(\mathcal{L}_1\mathbf{b})(p) + \beta(\mathcal{L}_2\mathbf{b})(p))(WB)^+ W = \\ &= \alpha(\mathcal{L}_1\mathbf{b})(p)(WB)^+ W + \beta(\mathcal{L}_2\mathbf{b})(p)(WB)^+ W = \\ &= \alpha\varphi_{\mathcal{L}_1, p} + \beta\varphi_{\mathcal{L}_2, p}. \end{aligned} \quad \square$$

Zgornja trditev pove, da je za popis vseh linearnih operatorjev do reda r dovolj izračunati le funkcije oblike za elementarne odvode D^ω za multiindeks $|\omega| \leq r$. Vsak drug operator $\mathcal{L} = \sum_{|\omega| \leq r} a_\omega D^\omega$ lahko po linearnosti v vsaki točki aproksimiramo z elementarnimi aproksimacijami, tako da jih primerno pomnožimo in seštejemo. V tem primeru ni potrebno, da so a_ω zgoraj konstante; lahko so tudi funkcije p

in v vsaki točki uporabimo drugače uteženo linearno kombinacijo D^ω . Poseben primer trditve 3.4, ki ga bomo pogosto uporabljali, se nanaša na Laplaceov operator, katerega funkcijo oblike bomo računali kot $\varphi_{\Delta,p} = \varphi_{\frac{\partial^2}{\partial x^2},p} + \varphi_{\frac{\partial^2}{\partial y^2},p}$.

Že v primeru 3.1 smo videli, da se za Laplaceov robni problem na intervalu naša metoda ujema z metodo končnih diferenc. Pokažimo to tudi na primeru dvodimen-zionalne Poissonove enačbe.

Trditev 3.5. *Metoda iz razdelka 3.1.2 se za reševanje Poissonove enačbe na enakomerni pravokotni mreži z razmakom h ujema z metodo končnih diferenc pri $\mathbf{b} = \{1, x, y, x^2, y^2\}$, $w \equiv 1$ in $n = 5$.*

Dokaz. Kot ponavadi označimo točke na mreži s koordinatami (x_i, y_j) , tako da je $x_{i+1} = x_i + h$ in $y_{j+1} = y_j + h$. Pokazati moramo, da se funkcija oblike v vsaki točki x ujema z aproksimacijo končnih diferenc, ki pravi

$$(\Delta u)(x_i, y_j) = \frac{u_{i+1,j} + u_{i,j+1} + u_{i-1,j} + u_{i,j-1} - 4u_{i,j}}{h^2}, \quad (3.43)$$

pri čemer spremenljivka $u_{i,j}$ pripada koordinati (x_i, y_j) . Če se aproksimaciji Lapla-ceovega operatorja ujemata, potem se ujemata tudi aproksimaciji rešitve, saj sistem pri obeh metodah gradimo na enak način. Ker je operator krajevno neodvisen, so tudi funkcije oblik odvisne samo od medsebojne lege točk in torej lahko brez škode za splošnost predpostavimo, da računamo funkcijo oblike za točko $p = (0, 0)$. Naj-bližjih n sosedov je tako $\mathcal{N}(p) = \{(0, 0), (0, h), (0, -h), (h, 0), (-h, 0)\}$. Matrika B iz sistema (3.26), ki ima po stolcih izračunane bazne funkcije v vseh sosedih, je

$$B = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & -h & 0 & h^2 \\ 1 & 0 & h & 0 & h^2 \\ 1 & h & 0 & h^2 & 0 \\ 1 & -h & 0 & h^2 & 0 \end{bmatrix} \quad (3.44)$$

in njen psevdoinverz, ki je kar njen inverz, je

$$B^+ = B^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{2h} & -\frac{1}{2h} \\ 0 & -\frac{1}{2h} & \frac{1}{2h} & 0 & 0 \\ -\frac{1}{h^2} & 0 & 0 & \frac{1}{2h^2} & \frac{1}{2h^2} \\ -\frac{1}{h^2} & \frac{1}{2h^2} & \frac{1}{2h^2} & 0 & 0 \end{bmatrix}. \quad (3.45)$$

Pri tem smo že upoštevali $w \equiv 1$. Vektor izračunanih vrednosti operatorja na baznih funkcijah v točki x je

$$(\Delta \mathbf{b})(x) = [0 \ 0 \ 0 \ 2 \ 2]^\top. \quad (3.46)$$

Od tod dobimo po zvezi (3.32)

$$\boldsymbol{\varphi}_\Delta = (\Delta \mathbf{b})(x)^\top B^+ = \left[-\frac{4}{h^2} \quad \frac{1}{h^2} \quad \frac{1}{h^2} \quad \frac{1}{h^2} \quad \frac{1}{h^2} \right], \quad (3.47)$$

kar je ravno iskana aproksimacija. \square

Opomba 3.6. Metodo za izračun funkcije oblike je zelo elegantno implementirati v Wolfram Mathematici [Wol17], kar olajša simbolno raziskovanje takih aproksimacij. Enačbe (3.44–3.47) so izračunane s programom 1.

```

1 (* podatki *)
2 p = {0, 0};
3 sosedci = {p, {0, -h}, {0, h}, {h, 0}, {-h, 0}};
4 bazne = {(1 &), (#1 &), (#2 &), (#1^2 &), (#2^2 &)};
5 operator = Function[f, Function[{x, y}, Derivative[2, 0][f][x, y] + Derivative[0, 2][f][x, y]]];
6 (* izračun *)
7 B = Table[Table[b @@ s, {b, bazne}], {s, sosedci}];
8 Lb = Table[operator[b] @@ p, {b, bazne}];
9 phi = Lb.PseudoInverse[B] // Simplify

```

Program 1: Računanje funkcij oblike na pravokotni mreži.

Opomba 3.7. Tudi za izbiro $n = 9$ in $\mathbf{b} = \{1, x, x^2, y, xy, x^2y, y^2, y^2x, y^2x^2\}$ dobimo enako aproksimacijo kot v trditvi 3.5, le da uporabimo več sosedov. Če jih uredimo po oddaljenosti od $(0, 0)$, dobimo aproksimacijo

$$\varphi_{\Delta} = \left[-\frac{4}{h^2} \quad \frac{1}{h^2} \quad \frac{1}{h^2} \quad \frac{1}{h^2} \quad \frac{1}{h^2} \quad 0 \quad 0 \quad 0 \quad 0 \right]. \quad (3.48)$$

Če pa bi izbrali $n = 25$ točk in bazne funkcije $\mathbf{b} = \{x^i y^j, 0 \leq i, j \leq 4\}$, bi dobili aproksimacijo četrtega reda.

Primer 3.8. Naredimo še en primer, kjer za bazne funkcije vzamemo monome. Vzemimo tokrat $n = 9$ in za bazne funkcije vse monome skupne stopnje manj kot 3, torej $\mathbf{b} = \{1, x, y, x^2, y^2, xy\}$. V tem primeru je matrika B velikosti 9×6 in s pomočjo programa 1 izračunamo

$$\varphi_{\Delta} = \left[-\frac{4}{3h^2} \quad -\frac{1}{3h^2} \quad -\frac{1}{3h^2} \quad -\frac{1}{3h^2} \quad -\frac{1}{3h^2} \quad \frac{2}{3h^2} \quad \frac{2}{3h^2} \quad \frac{2}{3h^2} \right]. \quad (3.49)$$

Vidimo, da tokrat v aproksimaciji upoštevamo vseh 9 sosedov, za razliko od aproksimacije v opombi 3.7. Naravno sledi vprašanje, ali je ta aproksimacija drugačnega reda? Izkaže se, da ne, saj z razvojem aproksimacij v Taylorjevo vrsto dobimo

$$\begin{aligned} \frac{-4u(0,0) + u(0,h) + u(0,-h) + u(h,0) + u(-h,0)}{h^2} &= \\ &= \Delta u + \frac{h^2}{12} \left(\frac{\partial^4 u}{\partial x^4}(0,0) + \frac{\partial^4 u}{\partial y^4}(0,0) \right) + O(h^4) \end{aligned} \quad (3.50)$$

in

$$\begin{aligned} \frac{-1}{3h^2} (4u(0,0) + u(0,h) + u(0,-h) + u(h,0) + u(-h,0)) + \\ + \frac{2}{3h^2} (u(h,h) + 2u(h,-h) + 2u(-h,h) + 2u(-h,-h)) &= \\ &= \Delta u + \frac{h^2}{12} \left(\frac{\partial^4 u}{\partial x^4}(0,0) + \frac{\partial^4 u}{\partial x^2 y^2}(0,0) + \frac{\partial^4 u}{\partial y^4}(0,0) \right) + O(h^4). \end{aligned} \quad (3.51)$$

Obe aproksimaciji sta torej drugega reda, razlikujeta se le pri izražavi napake.

Primer 3.9. Poglejmo še, kaj se zgodi, če za bazne funkcije vzamemo radialne bazne funkcije in $n = m$, torej postavimo po eno radialno funkcijo na vsakega izmed sosedov. Opisana metoda tako postane kolokacijska metoda z lokalnimi radialnimi baznimi funkcijami (*angl. local radial basis function collocation method*) [KŠ11].

Če za $n = 9$ sosedov točke $(0, 0)$ izberemo točke

$$\mathcal{N}(p) = \{(0,0), (0,h), (0,-h), (h,0), (-h,0), (h,h), (h,-h), (-h,h), (-h,-h)\} \quad (3.52)$$

in za \mathbf{b} vzamemo

$$\mathbf{b} = \{x \mapsto \exp((x - s)^2 / \sigma^2); s \in \mathcal{N}(p)\}, \quad (3.53)$$

potem ponovno s pomočjo programa 1 izračunamo aproksimacijo

$$\varphi_{\Delta} = \frac{h^2 e^{(h/\sigma)^2}}{\sinh((h/\sigma)^2)^2 \sigma^4} \left[-\frac{4+4(\sigma/h)^2 \sinh((h/\sigma)^2)^2}{e^{(h/\sigma)^2}} \quad 1 \quad 1 \quad 1 \quad 1 \quad 0 \quad 0 \quad 0 \quad 0 \right]. \quad (3.54)$$

Tokrat vidimo, da se precej razlikuje od aproksimacije s končnimi diferencami. Toda, še vedno je drugega reda, saj za napako velja

$$\varphi_{\Delta} \cdot \tilde{\mathbf{u}} - \Delta u(0, 0) = h^2 \left(\frac{2u(0, 0)}{\sigma^2} - \frac{\Delta u(0, 0)}{\sigma^2} + \frac{1}{12} \left(\frac{\partial^4 u}{\partial x^4}(0, 0) + \frac{\partial^4 u}{\partial y^4}(0, 0) \right) \right) + O(h^4). \quad (3.55)$$

Vidimo, da sta aproksimacija operatorja in napaka odvisni od tega, kakšen σ si izberemo v baznih funkcijah.

Trditev 3.10. *Aproksimacija Laplaceovega operatorja z devetimi Gaussovimi baznimi funkcijami konvergira proti aproksimaciji z monomi, ko gre $\sigma \rightarrow \infty$.*

Dokaz. Obe aproksimaciji smo izračunali že v primeru 3.9 in opombi 3.7. Po krajšem izračunu limite izraza (3.54), ko gre $\sigma \rightarrow \infty$, dobimo izraz (3.48). \square

3.3 Algoritem

V izpeljavi metode v razdelku 3.1.2 je nekaj podrobnosti ostalo neizdelanih, kot na primer, kako diskretiziramo domeno ali kako poiščemo sosedne. Ti detajli so opisani v naslednjih podrazdelkih, celotna metoda pa je opisana s pomočjo psevdokode kot algoritom 1.

3.3.1 Diskretizacija

Brezmrežne metode se pogosto oglašuje kot metode, pri katerih je manj težav z diskretizacijo domene in da lahko kar naključno porazdelimo diskretizacijske točke v domeno. Temu na žalost ni tako, saj sta natančnost in stabilnost metode odvisni od razporeditve točk [Kos16, slika 8], [AAN12] in je za kvalitetno rešitev potrebno poskrbeti da so točke čim bolj enakomerno razporejene.

V poljubni d -dimenzionalni domeni Ω je težko dobro razporediti točke. Uvedimo dve količini, ki merita, kako dobra je diskretizacija domene. Za dano domeno Ω in množico diskretizacijskih točk \mathcal{X} definirajmo

$$h_{\Omega}(\mathcal{X}) = \max_{p \in \Omega} \min_{x \in \mathcal{X}} \|p - x\|, \quad (3.56)$$

$$S(\mathcal{X}) = \min_{\substack{x, y \in \mathcal{X} \\ x \neq y}} \|x - y\|. \quad (3.57)$$

Količina h pove, kolikšna je maksimalna razdalja do najbližje točke kjer koli v domeni in intuitivno opisuje, kako dobro z diskretizacijskimi točkami pokrijemo domeno. Največji krog, ki ga lahko narišemo v domeni, ne da bi v notranjosti vseboval kakšno diskretizacijsko točko, je ravno radija $h_{\Omega}(\mathcal{X})$. Količina S pove, kolikšna je najmanjša razdalja med dvema sosednjima točkama in karakterizira stabilnost metode. Dobra diskretizacija želi pri danem številu točk maksimizirati S in minimizirati h .

Algoritem 1 Brezmrežna metoda za reševanje PDE iz razdelka 3.1.2.

Vhod: Parcialna diferencialna enačba, opisana v (3.22). Parametri metode:

- N ... celotno število diskretizacijskih točk
 Q ... število diskretizacijskih točk v notranjosti Ω
 n ... število sosedov, ki jih ima vsaka točka
 m ... število baznih funkcij
 b ... seznam baznih funkcij dolžine m
 w ... utež

Izhod: Skalarne polje u , ki aproksimira rešitev enačbe (3.22).

```
1: function REŠI( $\Omega, \mathcal{L}, f, \mathcal{R}, g, N, Q, n, m, b, w$ )
2:    $\mathcal{X} \leftarrow \text{DISKRETIJIRAJ}(\Omega, N, Q)$   $\triangleright \mathcal{X}$  postane seznam  $N$  točk, brez škode za
      splošnost naj leži prvih  $Q$  točk v  $\Omega$  in preostalih  $N - Q$  na  $\partial\Omega$ .
3:    $s \leftarrow \text{SOSEDI}(\mathcal{X}, n)$   $\triangleright s$  je seznam dolžine  $N$ , pri čemer je  $s[i]$  seznam
      indeksov elementov v  $\mathcal{X}$ , ki so sosedji  $\mathcal{X}[i]$ , vključno z  $i$ .
4:    $\varphi \leftarrow \text{prazen seznam dolžine } N$ 
5:   for  $i \leftarrow 1$  to  $Q$  do  $\triangleright$  Izračunamo funkcije oblik v notranjosti.
6:      $\varphi[i] \leftarrow \text{FUNKCIJA_OBLIKE}(\mathcal{L}, \mathcal{X}[i], \mathcal{X}, s[i], n, m, b, w)$   $\triangleright$  Glej algoritem 2.
7:   end for
8:   for  $i \leftarrow Q + 1$  to  $N$  do  $\triangleright$  Izračunamo funkcije oblik na robu.
9:      $\varphi[i] \leftarrow \text{FUNKCIJA_OBLIKE}(\mathcal{R}, \mathcal{X}[i], \mathcal{X}, s[i], n, m, b, w)$   $\triangleright$  Glej algoritem 2.
10:  end for
11:   $A \leftarrow \text{prazna razpršena } N \times N \text{ matrika}$ 
12:  for  $i \leftarrow 1$  to  $N$  do  $\triangleright$  Aproksimiramo enačbo.
13:    for  $j \leftarrow 1$  to  $n$  do
14:       $A[i, s[j]] \leftarrow \varphi[i][j]$ 
15:    end for
16:  end for
17:   $r \leftarrow \text{prazen vektor dolžine } N$ 
18:  for  $i \leftarrow 1$  to  $Q$  do  $\triangleright$  Izračunamo desno stran v notranjosti.
19:     $r[i] \leftarrow f(\mathcal{X}[i])$ 
20:  end for
21:  for  $i \leftarrow Q + 1$  to  $N$  do  $\triangleright$  Izračunamo robne pogoje.
22:     $r[i] \leftarrow g(\mathcal{X}[i])$ 
23:  end for
24:   $u \leftarrow \text{REŠIRAZPRŠENSISTEM}(A, r)$ 
25:  return  $u$ 
26: end function
```

Pri domenah lepih oblik kot so kvadri in krogle, lahko uporabimo znane algoritme za regularne diskretizacije. Tako lahko za diskretizacijo kvadra uporabimo kar enakomerno diskretizacijo. Za čim bolj enakomerno diskretizacijo notranjosti kroga ali površine sfere lahko uporabimo npr. Fibonaccijeve mreže, kot predlagano v [HN04] ali v [Gon10]. Z nekaj osnovnimi operacijami kot so unija in razlika množic lahko opišemo že dokaj zanimive oblike (slika 6).

Pogosto pa nimamo opravka z domenami pravilnih oblik in želimo splošnejši algoritmom, ki potrebuje samo implementacijo operacije $p \in \Omega$. Ker je domena omejena, je vsebovana v nekem d -dimenzionalnem kvadru in lahko diskretizacijo Ω konstrui-

Algoritem 2 Izračun funkcije oblike za operator \mathcal{L} .

Vhod: Parametri metode:

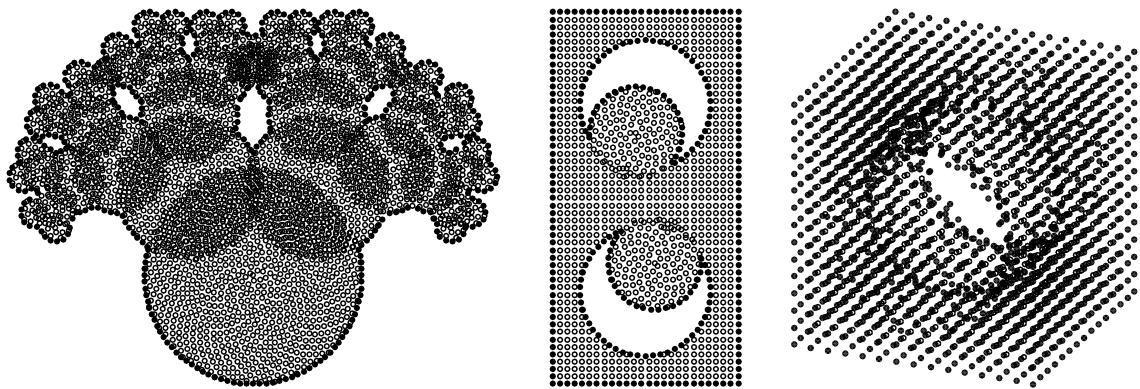
- \mathcal{L} ... parcialen diferencialen operator
- p ... točka, v kateri aproksimiramo operator
- \mathcal{X} ... seznam diskretizacijskih točk
- I ... množica indeksov točk v sosedstvu p
- n ... število sosedov, ki jih ima vsaka točka
- m ... število baznih funkcij
- b ... seznam baznih funkcij dolžine m
- w ... utež

Izhod: Funkcional, ki aproksimira operator \mathcal{L} v točki p .

```

1: function FUNKCIJA_OBLIKE( $\mathcal{L}, p, \mathcal{X}, I, n, m, b, w$ )
2:    $W \leftarrow$  prazen vektor dolžine  $n$ 
3:   for  $i \leftarrow 1$  to  $n$  do
4:      $W[i] \leftarrow \sqrt{w(\mathcal{X}[I[i]] - p)}$ 
5:   end for
6:    $B \leftarrow$  prazna matrika velikosti  $n \times m$ 
7:   for  $i \leftarrow 1$  to  $n$  do
8:     for  $j \leftarrow 1$  to  $m$  do
9:        $B[i, j] \leftarrow W[i] \cdot b[j](\mathcal{X}[I[i]])$ 
10:    end for
11:   end for
12:    $\ell \leftarrow$  prazen vektor dolžine  $m$ 
13:   for  $j \leftarrow 1$  to  $m$  do
14:      $\ell[j] \leftarrow (\mathcal{L}(b[j]))(p)$ 
15:   end for
16:    $\varphi \leftarrow (\ell \cdot \text{PINV}(B)) \odot W$        $\triangleright$  Direktna analogija enačbe (3.32),  $\odot$  označuje
17:   return  $\varphi$                                  $\triangleright$  Hadamardov produkt.
18: end function

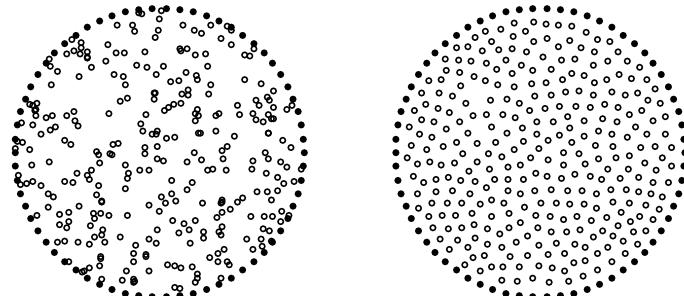
```



Slika 6: Primeri bolj zapletenih domen in njihovih diskretizacij.

ramo tako, da vzamemo enakomerno diskretizacijo tega kvadra ter odstranimo vse točke, ki niso v domeni. Druga metoda je, da naključno izbiramo točke v kvadru, in sprejmemmo tiste, ki pristanejo v notranjosti Ω . Še boljšo diskretizacijo dobimo, če namesto psevdonaključnih števil pri izbiranju točk uporabimo kvazinaključna števila, ki imajo manjšo diskrepanco [MC94] in so zato bolj primerna. Kljub njihovi

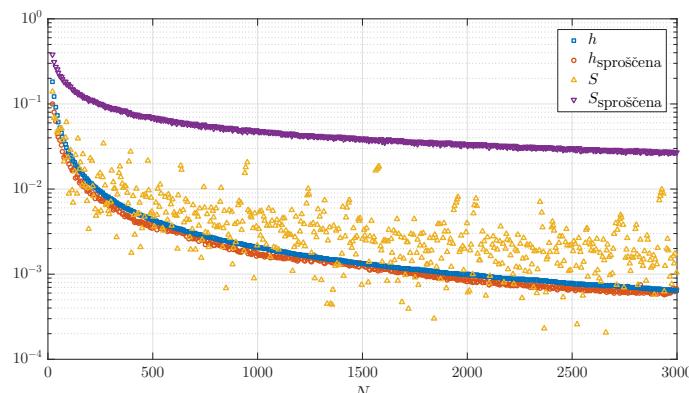
splošnosti so naključne diskretizacije precej slabe, kot lahko vidimo na enakomerni naključni diskretizaciji kroga na sliki 7 (levo).



Slika 7: Primerjava domene z naključno diskretizacijo (levo) in domene po izvedbi 10 iteracij algoritma 3 s parametri $F_0 = 10^{-2}$, $s = 6$, $\alpha = 3$ (desno).

Vendar ni še vse izgubljeno, saj lahko zgoraj opisane enostavne postopke za diskretizacijo uporabimo, da dobimo začetni približek za boljšo diskretizacijo. Pri tem, da bi ta približek izboljšali in točke v domeni porazdelili čim bolj enakomerno, si lahko pomagamo s preprostim iterativnim postopkom, za katerega se izkaže, da v praksi dobro deluje. Navdih jemljemo iz fizike in si zamislimo, da je vsaka točka naboj, ki se odbija od sosednjih točk. Na vsako točko tako deluje sila, porojena z nekim potencialom, ki jo sili stran od drugih točk, v prazen prostor. Točko lahko malo premaknemo v smeri sile, ponovno izračunamo medsebojne sile in postopek ponavljamo. Pri tem točkam na robu ne dovolimo premikanja. Če kljub temu kakšna točka iz notranjosti zleze iz domene, jo postavimo nazaj na naključno mesto v domenu. Zapis zgornjega postopka je v psevdokodi podan kot algoritem 3. Postopek se imenuje *sprostitev* (angl. relaxation) domene [Kos16, razdelek 2.2], kajti na začetku so med točkami napetosti, ki jih s premikanjem minimizirajo. Rezultati opisanega postopka na primeru izboljšanja naključne diskretizacije kroga so prikazani na sliki 7 (desno).

Da algoritem izboljša kvaliteto aproksimacije se vidi tudi, če izračunamo parameter h in S . Na sliki 8 vidimo primerjavo med kvaliteto diskretizacije s Fibonaccijevo mrežo in njeno 30-kratno izboljšavo. Vidimo, da se je h nekoliko izboljšal, za ceno večje variance, S pa se je precej izboljšal in izkazuje celo lepše obnašanje kot prej.



Slika 8: Spremembra h in S po 30 iteracijah algoritma 3 s parametri $F_0 = 10^{-2}$, $s = 6$, $\alpha = 3$ v enotskem krogu z začetno Fibonaccijevo mrežo, v odvisnosti od N .

Algoritem 3 Algoritem za izboljšanje kvalitete diskretizacije domene.

Vhod: Parametri metode:

- Ω ... domena
- N ... število diskretizacijskih točk
- Q ... število diskretizacijskih točk v notranjosti Ω
- \mathcal{X} ... seznam diskretizacijskih točk, prvih Q je v notranjosti.
- I ... število iteracij
- s ... število sosedov, ki jih upoštevamo pri delovanju sile
- F_0 ... delež sile, ki vpliva na premik
- α ... eksponent v sili

Izhod: Nov seznam diskretizacijskih točk.

```

1: function IZBOLJŠAJ( $\Omega, N, Q, \mathcal{X}, I, s, F_0, \alpha$ )
2:    $r_\chi \leftarrow \left( \frac{\text{VOLUMEN}(\Omega)}{N} \right)^{\frac{1}{\text{DIMENZIJA}(\Omega)}}$      $\triangleright$  Približek povprečne razdalje med točkami.
3:   for  $i \leftarrow 1$  to  $I$  do
4:     for  $j \leftarrow 1$  to  $Q$  do
5:        $\vec{F} \leftarrow \vec{0}$ 
6:       for each  $y$  in {najbližjih  $s$  sosedov  $\mathcal{X}[i]$ } do
7:          $\vec{r} \leftarrow \frac{\mathcal{X}[i] - y}{r_\chi}$            $\triangleright$  Brezdimenzijski vektor razdalje.
8:          $\vec{F} \leftarrow \vec{F} + \frac{\vec{r}}{\|\vec{r}\|^\alpha}$ 
9:       end for
10:       $\mathcal{X}[i] \leftarrow \mathcal{X}[i] + F_0 \vec{F}$ .
11:      if  $\mathcal{X}[i] \notin \Omega$  then
12:         $\mathcal{X}[i] \leftarrow$  naključna pozicija znotraj  $\Omega$ 
13:      end if
14:    end for
15:  end for
16:  return  $\mathcal{X}$ 
17: end function

```

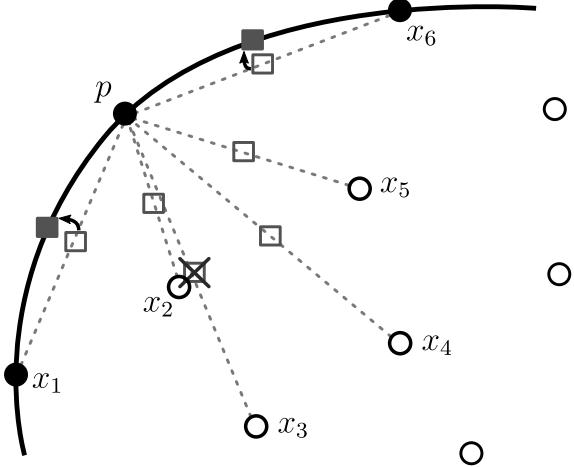
3.3.2 Goščenje diskretizacije

Pri reševanju problemov z divje spreminjajočimi reštvami, singularnostmi ali zelo majhnimi lokalnimi fenomeni si pogosto pomagamo z lokalnim zgoščevanjem krajevne diskretizacije na delu domene. Pri metodah, ki temeljijo na triangulaciji domene, lahko trikotnike v območju interesa delimo in tako dobimo finejšo triangulacijo. Pri brezmrežnih metodah želimo doseči enak učinek, zato potrebujemo način, da gostimo točke na nekem območju. Postopek bomo imenovali *goščenje* (*angl. refinement*).

Izberimo točko p , okoli katere bomo gostili diskretizacijo. Pri opisu postopka si pomagajmo z ilustracijo na sliki 9. Poiščimo najbližjih ℓ sosedov točke p , pri čemer ne štejemo p , in jih označimo z $L = \{x_1, \dots, x_\ell\}$. Za vsakega izmed sosedov dodamo novo točko na polovici daljice med x_i in p , označeno s sivim kvadratkom. Kandidati za nove diskretizacijske točke so torej

$$\mathcal{X}' = \left\{ \frac{p + x}{2}; x \in L \right\}. \quad (3.58)$$

Pri tem pazimo, da v primeru, ko sta tako x kot p točki na robu, tudi njuno središče



Slika 9: Algoritem za goščenje točk v domeni.

okličemo za robno točko in jo projiciramo na rob domene, kot prikazano na primeru na sliki 9 z dvema potemnjenima kvadratkomoma. Če je katera izmed točk v \mathcal{X}' izven $\bar{\Omega}$, jo odstranimo iz množice kandidatov. Poleg tega se lahko zgodi, da so nekatere nove točke preblizu skupaj ali pa preblizu starim točkam, kar bi povzročilo nestabilnosti pri računanju funkcij oblike. Zato v parih točk, ki so bolj skupaj kot dovoljena razdalja r_{\min} zavrhemo novejšo točko, če je ta preblizu kakšni stari. Za razdaljo r_{\min} ponavadi vzamemo $r_{\min} = fr_c$, kjer je r_c razdalja do najbližnjega soseda (3.61) in f pozitiven faktor, ponavadi manjši od $1/2$.

Zgornji postopek je lokalne narave in njegovo izvajanje je, zaradi iskanja sosedov, le logaritemsko odvisno od števila točk v domeni. Če želimo na nekem območju še gostejšo diskretizacijo, postopek preprosto ponovimo večkrat zapored. Primer večkrat zgoščene domene je prikazan na sliki 26 na strani 65.

3.3.3 Iskanje najbližjih sosedov

Tako v algoritmu 1 na vrstici 3, kot v algoritmu 3 na vrstici 6 je potrebno poiskati najti nekaj najbližjih sosedov dane točke. Problem iskanja najbližjih sosedov (*angl. nearest neighbour search*) je znana in dobro raziskana tema z razvitim veliko podatkovnimi strukturami, ki podpirajo grajenje, iskanje, vstavljanje in brisanje v logaritemskem času. Večina jih temelji na delitvi prostora na različne hierarhično urejene podprostore, ki jih potem hranimo v drevesni strukturi, kar omogoča logaritemski dostop. Med bolj znanimi strukturami so *k-d tree* [Moo91], *ball tree* [Omo89] in *cover tree* [BKL06]. V članku [KF07] je narejena empirična primerjava med zgornjimi tremi strukturami, ki pokaže, da se v primeru nizkih dimenzij (kar gotovo vsebuje $d \leq 3$) najbolje obnese *k-d* drevo. Poleg tega *k-d* drevo deluje najbolje tudi, ko imajo točke, v katerih bomo iskali sosede, podobno porazdelitev kot točke, iz katerih naredimo drevo, kar v našem primeru drži. Če vključimo še dejstvo, da je *k-d* drevo tudi najpopularnejša rešitev za problem najbližjih sosedov in ima na voljo veliko prosto dostopnih implementacij, je to dovolj argumentov, da jo uporabimo tudi mi. Specifična uporabljeni implementacija je predstavljena v [MA98], ki za shrambo N točk porabi $O(N)$ prostora in odgovarja na poizvedbe o n najbližjih sosedih v $O(n \log N)$ časa. S pomočjo tega postane implementacija funkcije SOSEDI iz algoritma 1 trivialna, pri algoritmu 3 pa si na vsaki iteraciji na novo zgradimo

drevo in ga uporabimo za iskanje s najbližjih sosedov.

3.3.4 Reševanje razpršenega sistema

Za reševanje razpršenih sistemov poznamo veliko različnih metod, ki se v grobem delijo na direktne [Dav06] in iterativne [Saa03]. Pri običajnih, polnih matrikah bi za reševanje splošnega sistema linearnih enačb uporabili LU razcep in razcepili matriko A kot $A = LU$. Toda, tudi če je matrika A razpršena, v splošnem L in U nista nujno in sta lahko celo polni, kar povzroči, da nam zmanjka pomnilnika. Direktne metode, kot na primer SuperLU [Li05], poskušajo s preuređitvijo stolpcev v razpršeni matriki A minimizirati število novih neničelnih elementov v matrikah L in U . Po drugi strani iterativne metode aproksimirajo pravilno rešitev sistema $Ax = b$ z zaporedjem približkov $\{x^{(r)}\}$, ki naj bi konvergirali k x . Te metode običajno ne zahtevajo veliko pomnilnika, je pa natančnost približka odvisna od števila porabljenih iteracij, ali drugače, z višanjem želene natančnosti raste tudi računska zahtevnost.

V primerih, predstavljenih v tem delu, uporabljamo knjižnico Eigen [GJ⁺10], ki nudi elegantno in hitro delo z matrikami v C++. Vgrajenih ima več direktnih in iterativnih metod za reševanje sistemov linearnih enačb.¹ Pri sistemih tipa (3.37) se je v praksi najbolje obnesel BiCGSTAB [Vor92] z ILUT predpogojevanjem [Saa94]. BiCGSTAB v primeru uporabe pravilne shrambe matrike v pomnilniku podpira tudi paralelizacijo, poleg tega pa je konvergenca v praksi dovolj hitra. Predpogojevanje z uporabo nepopolnega LU razcepa z dvojnim pragom (*angl. dual threshold incomplete LU factorization (ILUT)*) omogoča natančnejšo kontrolo nad porabljenim spominom in hitrostjo konvergencije z uporabo dveh parametrov p in τ . Med LU faktorizacijo izpustimo vsak element, ki je manjši kot $\tau \cdot e$, kjer je e povprečna absolutna vrednost elementov v trenutni vrstici. Nato obdržimo le največjih f elementov v matrikah L in U , kjer je p uporabljen kot razmerje med f in začetnim številom neničelnih elementov. V grobem tako parameter p kontrolira, kolikokrat več spomina dovolimo za hranjenje predpogojenke, parameter τ pa, kako natančna bo LU faktorizacija.

3.3.5 Časovna zahtevnost

Pri analizi časovne zahtevnosti metode bomo predpostavili, da so vse evalvacije funkcij f , g , w in b_j , $\mathcal{L}b_j$, $\mathcal{R}b_j$ izvedene v $O(1)$. Prav tako bomo predpostavili, da je dimenzija problema majhna in konstantna, zato ne bo nastopala v analizah.

Trditev 3.11. *Pričakovana časovna zahtevnost algoritma 1 je*

$$O(INs \log^2 N + (N + n) \log N + m^2 n N) + T, \quad (3.59)$$

kjer je T čas, porabljen za reševanje razpršenega sistema enačb.

Dokaz. Pri funkciji DISKRETIZIRAJ predpostavimo, da uporabljamo enostavno diskretizacijo, ki deluje v $O(N)$ časa, skupaj z I iteracijami izboljšave na s sosedih, ki traja $O(INs \log^2 N)$ časa, saj na vsaki iteraciji konstruiramo drevo in iščemo s sosedov vsake točke.

Izvajanje funkcije SOSEDI potrebuje $O((N + n) \log N)$ časa za konstrukcijo drevesa in iskanje sosedov.

¹https://eigen.tuxfamily.org/dox/group__TopicSparseSystems.html [ogled 17. 6. 2017]

Izvajanje funkcije FUNKCIJA_OBLIKE traja $O(n + nm + m + m^2n + nm + n) = O(m^2n)$ časa, kjer je prevladujoči faktor $O(m^2n)$ rezultat računanja SVD razcepa z dvostranskim Jacobijevim SVD razcepom iz knjižnice Eigen.² Funkcija oblike se izračuna za vsako točko v domeni, kar doprinese $O(m^2nN)$ k skupni zahtevnosti.

Pri sestavljanju razpršene matrike lahko v naprej rezerviramo prostor za n elementov v vsaki vrstici, tako da sestavljanje matrike stane $O(nN)$ časa, sestavljanje robnih pogojev pa $O(N)$ časa. Na koncu še rešimo razpršen sistem linearnih enačb, kjer pa je čas zelo odvisen od problema, iteracijske metode in predpogojenke.

Skupna časovna zahtevnost je tako $O(INS \log^2 N + (N + n) \log N + m^2nN) + T$, kjer je T čas, porabljen za reševanje razpršenega sistema enačb. \square

3.3.6 Prostorska zahtevnost

Podobno kot pri časovni zahtevnosti bomo predpostavili, da so vse evalvacije funkcij f, g, w in $b_j, \mathcal{L}b_j, \mathcal{R}b_j$ izvedene v $O(1)$ prostora. Prav tako bomo predpostavili, da je dimenzija problema majhna in konstantna, zato ne bo nastopala v analizah.

Trditev 3.12. *Prostorska zahtevnost algoritma 1 je $O(nN) + P$, kjer je P prostor, potreben za reševanje sistema linearnih enačb.*

Dokaz. Pri funkciji DISKRETIZIRAJ potrebujemo $O(N)$ spomina za hrambo N diskretizacijskih točk. Če izvajamo še dodatne izboljšave diskretizacije, nas to stane $O(n)$ dodatnega prostora.

Za izvajanje funkcije SOSEDI porabimo $O(N)$ prostora za hranjenje drevesa in $O(nN)$ prostora za hranjenje indeksov n sosedov.

Izvajanje funkcije FUNKCIJA_OBLIKE nas stane $O(n + nm + m + n) = O(nm)$ prostora za vsak klic, za hrambo N funkcij oblike pa potrebujemo $O(nN)$ prostora.

Razpršena matrika prav tako potrebuje $O(nN)$ prostora za neničelne elemente. Nato moramo rešiti le še sistem linearnih enačb, kar po predpostavki stane P prostora. Ker je $m = O(N)$, je prevladujoči faktor $O(nN)$ in skupna prostorska zahtevnost je $O(nN) + P$. \square

3.4 Pogoste vrednosti parametrov

Metoda je bila do sedaj formulirana zelo splošno, v praksi pa se uporablja nekaj ustaljenih kombinacij. Videli bomo, da razdalje pogosto merimo v večkratnikih r_χ , kot v vrstici 2 v algoritmu 3.

Definicija 3.13. Količino r_χ , pripisano diskretizaciji \mathcal{X} domene Ω , izračunano z

$$r_\chi(\Omega, \mathcal{X}) = \left(\frac{\text{VOLUMEN}(\Omega)}{|\mathcal{X}|} \right)^{\frac{1}{\text{DIMENZIJA}(\Omega)}}, \quad (3.60)$$

imenujemo *karakteristična razdalja*.

Ideja definicije je, da r_χ predstavlja približno povprečno razdaljo med diskretizacijskimi točkami, če bi bile razporejene enakomerno. Celoten volumen domene razdelimo na $N = |\mathcal{X}|$ delov, tako da vsaki točki pripada enak kos volumna $v = \frac{\text{VOLUMEN}(\Omega)}{N}$. Če bi bil ta kos neka hiperkocka v d dimenzijah, potem je $\sqrt[d]{v}$ dolžina njene stranice in to je ocena za medsebojno razdaljo med točkami. V eni dimenziji r_χ do faktorja

²https://eigen.tuxfamily.org/dox/classEigen_1_1JacobiSVD.html [ogled 16. 6. 2017]

$\frac{N}{N-1}$ natančno ustreza razdalji med enakomerno razporejenimi točkami. Drugo pogosto merilo za lokalno razdaljo bo kar razdalja do najbližjega (različnega) soseda, ki jo imenujmo r_c ,

$$r_c(p) = \min_{\substack{x \in \mathcal{N}(p) \\ x \neq p}} \|x - p\|. \quad (3.61)$$

To je težje izračunati, če nimamo že zgrajenega drevesa, zato se ponavadi na enostavnnejših primerih poslužujemo kar r_χ . Kadar pa to ni dobra aproksimacija razdalje med vozlišči za celo domeno, kot na primer ob goščenju mreže, uporabimo r_c .

Pogoste vrednosti za število sosedov so $n = 3, 5, 7$ v eni dimenziji, $n = 5, 9, 13, 25$ v dveh dimenzijah in $n = 7$ ali 27 v treh dimenzijah. Za regularno porazdeljene točke v domeni to pomeni, da so vozlišča izbrana simetrično, pri neregularnih pa izbira n ni tako pomembna, dokler je dovolj visoka, da dobro popiše operator, ki ga aproksimiramo. Višji n ponavadi pomeni večjo stabilnost, morda višji red in počasnejše izvajanje. V praksi izberemo najmanjši n , ki zagotovi zadovoljive rezultate.

Za utež se pogosto uporablja konstanta $w \equiv 1$, če ne želimo uteženih najmanjših kvadratov, sicer pa pogosto vzamemo za utež Gaussovo funkcijo

$$w(x) = \exp\left(-\left(\frac{x}{\sigma_w}\right)^2\right), \quad (3.62)$$

kjer σ_w imenujemo parameter oblike (*angl. shape parameter*) uteži. Velja opozoriti, da je ta funkcija praktično nič, če smo več kot $3\sigma_w$ stran od središča ($w(3\sigma_w) \approx 0.0001234$). Če na primer izberemo $\sigma_w = r_\chi$, se vsi sosedji, ki so dlje kot r_χ stran, v aproksimaciji ne upoštevajo, ne glede na to, kako velik n izberemo. Ponavadi se za σ_w začne izbirati vrednosti okoli r_χ , je pa potrebno optimalno vrednost določiti za vsak posamezen problem posebej.

Pogosta izbira baznih funkcij so prostori \mathbb{P}_ℓ monomov skupne stopnje manj ali enako ℓ . Pri tem moramo seveda paziti, da izberemo dovolj visok n . Druga pogosta izbira je, da vzamemo $m = n$ in za bazne funkcije izberemo radialne bazne funkcije s centri v sosedih [KS11].

Definicija 3.14. Funkcija ψ_c se imenuje radialna bazna funkcija s centrom v točki c , če je odvisna samo od razdalje do centra, torej

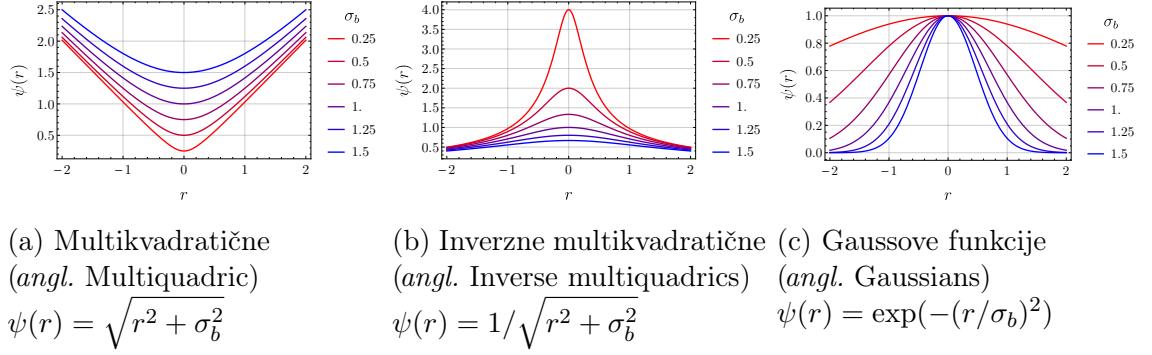
$$\psi_c(x) = \tilde{\psi}(\|x - c\|), \quad (3.63)$$

za vsak x za neko funkcijo ψ . Kasneje kar identificiramo ψ_c in $\tilde{\psi}$ in pišemo $\psi_c = \psi_c(r)$.

Na sliki 10 so naštete najpogostejše uporabljeni radialni bazni funkciji, povzete po [Sch95, str. 5]. Ko si izberemo neko bazno funkcijo ψ , potem za bazne funkcije okoli točke p vzamemo

$$\mathbf{b} = \{\psi_s; s \in \mathcal{N}(p)\}. \quad (3.64)$$

Pri izbiri parametra oblike σ_b moramo biti do neke mere pazljivi. Izbrati ga moramo dovolj velikega, da se funkcije “prekrivajo”, torej tako, da funkcija nima vrednosti blizu nič pri vseh sosedih razen pri sebi. Hkrati mora biti parameter dovolj majhen, da sistem (3.26) ni preveč nestabilen. Bolj natančna analiza izbire parametra oblike za radialne bazne funkcije in za utež je narejena na sliki 16 med numeričnimi zgledi na strani 56.



Slika 10: Najpogosteje uporabljene radialne bazne funkcije s parametrom oblike σ_b .

Pogosto se k bazi radialnih baznih funkcij doda še monome nizkih stopenj za boljšo aproksimacijo funkcij, ki so blizu konstantam. Več o kvaliteti, stabilnosti in redu konvergencije aproksimacije z radialnimi baznimi funkcijami si lahko bralec prebere v [Buh00].

4 Osnovni numerični zgledi

V tem razdelku si bomo ogledali obnašanje metode na čedalje bolj zapletenih numeričnih zgledih, da si zgradimo intuicijo o njenem delovanju in da se pripravimo na reševanje bolj zapletenih problemov. V nadalnjem besedilu in na vseh grafih bo metoda, opisana v razdelku 3, označena s kratico MLSM (*angl.* Meshless Local Strong Form Method). Če ni drugače navedeno, so vse meritve narejene na računalniku z eno centralno procesno enoto Intel(R) Core(TM) i7-4700MQ CPU @2.40GHz s štirimi jedri, od katerih vsako podpira dve niti izvajanja, in 16 GiB DDR3 pomnilnika. Vsi programi so prevedeni s prevajalnikom g++ verzije 7.1.7 in stikali -std=c++14 -O3 -DNDEBUG na operacijskem sistemu Manjaro Linux.

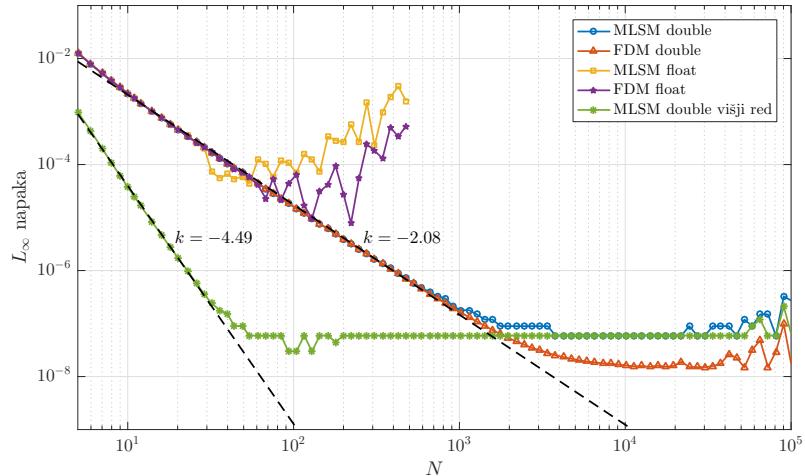
4.1 Enodimensionalni robni problem

Oglejmo si najprej preprost primer reševanja enodimensionalne Poissonove enačbe, ki smo ga uporabili že za motivacijo izpeljave numerične metode. Rešujmo problem

$$\begin{aligned} u''(x) &= \sin(x), \quad x \in (0, 1), \\ u(0) &= 0, \\ u'(1) &= 0, \end{aligned} \tag{4.1}$$

ki ima analitično rešitev $u(x) = \cos(1)x - \sin(x)$. Na sliki 11 je prikazana konvergenca metode končnih differenc in MLSM. Metoda MLSM je uporabljen s parametri, pri katerih velja trditev 3.5, torej $n = m = 3$, $w = 1$, $\mathbf{b} = \{1, x, x^2\}$. Pri obeh metodah uporabimo dvojno in enojno natančnost računanja s plavajočo vejico. Za primerjavo problem rešimo tudi z višjim redom $n = m = 5$, $w = 1$, $\mathbf{b} = \{1, x, x^2, x^3, x^4\}$. V obeh primerih uporabimo direktno metodo SuperLU za reševanje sistema enačb. Za vsakega od teh primerov izračunamo diskretno L_∞ napako v diskretizacijskih točkah:

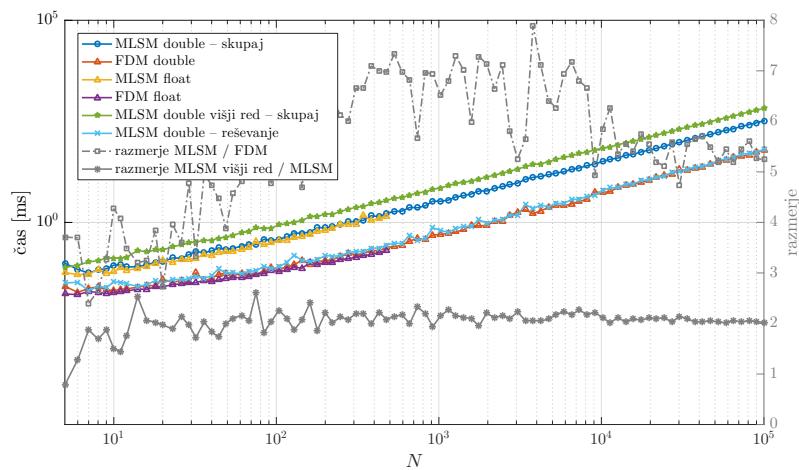
$$L_\infty = \max_{x \in X} |\hat{u}(x) - u(x)|. \tag{4.2}$$



Slika 11: Napaka FDM in MLSM metode v primerjavi s pravilno rešitvijo.

Na grafu (slika 11) vidimo, da metodi tudi v praksi sovpadata, dokler ne preideta v območje nestabilnosti. Pri enojni natančnosti se to zgodi precej hitro, pri

$N = 30$, pri dvojni natančnosti pa obe metodi konvergirata linearno do približno $N = 1000$. Nato se obe metodi približujeta vsaka svoji največji možni natančnosti, med $N = 10^4$ in $N = 10^5$ diskretizacijskimi točkami pa se začnejo pojavljati numerične nestabilnosti. Manjšo končno natančnost MLSM lahko pripisemo numeričnim napakam pri računanju aproksimacije drugih odvodov. Pri FDM namreč v matriko direktno zapišemo koeficiente $\frac{1}{h^2}$ in $-\frac{2}{h^2}$, pri MLSM pa se ti izračunajo numerično. Iz naklona premice vidimo, da sta obe metodi tudi v praksi reda 2. Če za računanje aproksimacije odvoda uporabimo 5 sosedov namesto 3, po pričakovanjih dobimo metodo, ki konvergira z višjim redom, ima pa enako končno natančnost kot prej.



Slika 12: Primerjava časa izvajanja MLSM in FDM metod.

Časovna primerjava obeh metod je prikazana na sliki 12. MLSM je približno konstantno 5.5-krat počasnejši kot FDM za velike ($\geq 10^4$) N , kot prikazano z razmerjem, merjenim na desni osi. To je tudi pričakovano, saj MLSM išče sosede in gradi funkcije oblike, medtem ko so pri FDM uporabljene aproksimacije znane vnaprej. Če pri MLSM posebej merimo samo čas, ki ga porabimo za sestavljanje matrike in reševanje sistema, vidimo, da se skoraj ujema z časom, porabljenim pri FDM. Majhno razliko gre pripisati razlike v načinu grajenja matrike, saj pri FDM natanko vemo število in pozicije neničelnih elementov vnaprej, medtem ko za MLSM to v splošnem ne velja. MLSM višjega reda je pričakovano počasnejši, saj sta n in m poleg osnovnega faktorja N glavna parametra v časovni zahtevnosti (3.59). Iz razmerja vidimo, da je MLSM višjega reda konsistentno dvakrat počasnejša. Toda, če pogledamo koliko časa potrebujemo, da dosežemo natančnost 10^{-6} , z grafov 11 in 12 odčitamo, da MLSM višjega reda potrebuje le $N = 22$ točk, za kar potrebujemo 0.3 ms, pri običajnem redu pa potrebujemo $N = 400$ točk, kar nanese 1.5 ms. Implementacija deljenih diferenc višjega reda je dovolj zapletena, da se zanjo malokrat odločimo, saj je treba uporabljati enostranske difference pri robu. Pri MLSM se te izračunajo same; vse, kar je potrebno, je, da n iz 3 spremenimo na 5. Če uporabljamo enojno natančnost namesto dvojne, se to zgolj malenkostno pozna na času, ki ga porabimo za reševanje, zato se enojne natančnosti zaradi njene zgodnje nestabilnosti ne splača uporabljati.

4.2 Poissonova enačba

Oglejmo si obnašanje numerične rešitve klasične Poissonove enačbe na kvadratu:

$$\begin{aligned}\Delta u &= 1, \quad (x, y) \in (0, 1) \times (0, 1), \\ u(x, 0) &= u(x, 1) = u(0, y) = u(1, y) = 0.\end{aligned}\tag{4.3}$$

Analitično rešitev lahko dobimo s pomočjo separacije spremenljivk in se glasi

$$u(x, y) = -8 \sum_{\substack{k=1 \\ k \text{ lih}}}^{\infty} \frac{\sin(k\pi x) \sinh \frac{k\pi(1-y)}{2} \sinh \frac{k\pi y}{2}}{\cosh(\frac{k\pi}{2}) k^3 \pi^3}. \tag{4.4}$$

Za primerjavo z numeričnimi reštvami bomo uporabili končno vsoto, zato ocenimo ostanek.

Trditev 4.1. Za ostanek vrste (4.4) velja

$$\left| -8 \sum_{\substack{k=\ell \\ k \text{ lih}}}^{\infty} \frac{\sin(k\pi x) \sinh \frac{k\pi(1-y)}{2} \sinh \frac{k\pi y}{2}}{\cosh(\frac{k\pi}{2}) k^3 \pi^3} \right| < -\frac{\psi''(\ell/2)}{4\pi^3}, \tag{4.5}$$

kjer je $\psi(x) = \frac{d}{dx} \log \Gamma(x)$ digama funkcija.

Dokaz. Ocenujmo vsak člen vrste posebej. Funkcijo sin ocenimo z 1, funkcija $\sinh \frac{k\pi(1-y)}{2} \sinh \frac{k\pi y}{2}$ ima maksimum na sredini intervala in jo lahko ocenimo z njeno vrednostjo v $y = \frac{1}{2}$. Tako moramo oceniti še stevilsko vrsto

$$-8 \sum_{\substack{k=\ell \\ k \text{ lih}}}^{\infty} \frac{\left(\sinh \frac{k\pi}{4}\right)^2}{\cosh(\frac{k\pi}{2}) k^3 \pi^3}.$$

Uporabimo neenakost $\frac{\left(\sinh \frac{k\pi}{4}\right)^2}{\cosh(\frac{k\pi}{2})} < \frac{1}{2}$ in dobimo oceno

$$\left| -8 \sum_{\substack{k=\ell \\ k \text{ lih}}}^{\infty} \frac{\sin(k\pi x) \sinh \frac{k\pi(1-y)}{2} \sinh \frac{k\pi y}{2}}{\cosh(\frac{k\pi}{2}) k^3 \pi^3} \right| < \frac{4}{\pi^3} \sum_{\substack{k=\ell \\ k \text{ lih}}}^{\infty} \frac{1}{k^3} = -\frac{\psi''(\ell/2)}{4\pi^3},$$

kjer je ψ digama funkcija (in posledično ψ'' poligama funkcija). \square

Za primer $\ell = 1$ v prejšnji trditvi dobimo oceno

$$|u(x, y)| \leq \frac{7\zeta(3)}{2\pi^2} \approx 0.1356, \tag{4.6}$$

kar drži; v resnici je najbolj ekstremna vrednost $u(1/2, 1/2) \approx -0.0736$. Iz zgornje trditve lahko izračunamo, da je rep vrste (4.4) za $\ell = 59$ manjši kot 10^{-5} , za $\ell = 181$ pa manjši kot 10^{-6} . Za izračun analitične rešitve ob primerjavi z numerično je uporabljen $\ell = 181$.

Opomba 4.2. Računanje izraza

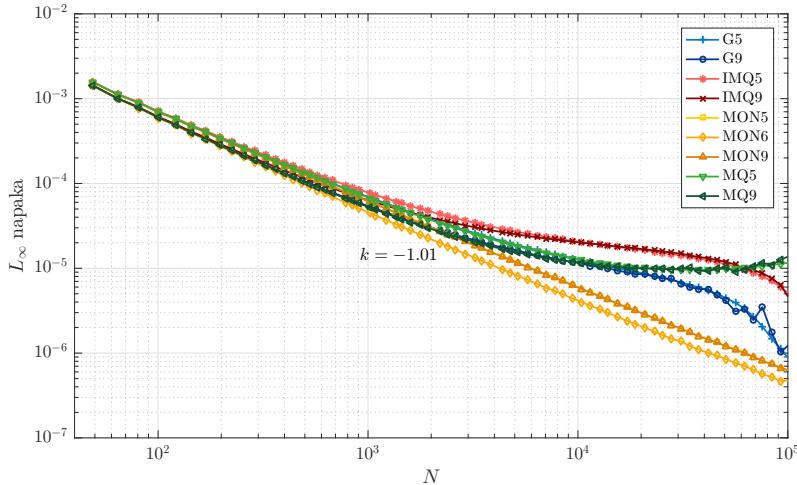
$$\frac{\sinh \frac{k\pi(1-y)}{2} \sinh \frac{k\pi y}{2}}{\cosh(\frac{k\pi}{2})} \quad (4.7)$$

lahko povzroči prekoračitev (*angl. overflow*), ko k raste, kajti tako števec kot imenovalec se približuje neskončnosti. Kljub temu ima kvocient končno limito, zato je za numeričen izračun bolje uporabiti obliko

$$\frac{1}{2} \left(1 - \frac{\exp(-k\pi y) + \exp(-k\pi(1-y))}{1 + \exp(-k\pi)} \right), \quad (4.8)$$

kjer vedno nastopa negativen eksponent.

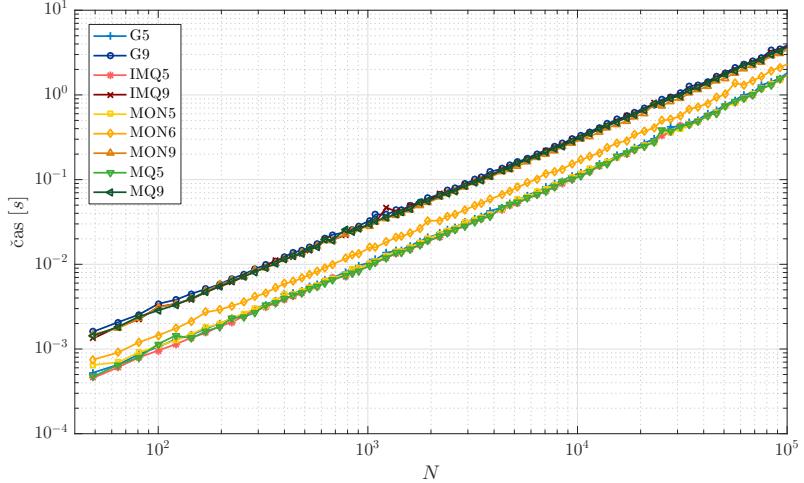
Rešimo problem (4.3) numerično, s štirimi različnimi nabori baznih funkcij: monomi (MON), Gaussovimi funkcijami (G), multikvadratičnimi funkcijami (MQ) in inverznimi multikvadratičnimi funkcijami (IMQ). Za parametre vedno vzemimo $n = m = 9$, kar sovpada s primeri iz razdelka 3.2. Pri monomih dodajmo tudi primer $n = 9$ in $m = 6$. Za parameter oblike pri radialnih baznih funkcijah vzemimo $\sigma_b = 100 r_\chi$. Za utež vzemimo Gaussovo utež z $\sigma_w = \frac{3}{4} r_\chi$. V vseh primerih je za reševanje sistema enačb uporabljen direktna metoda SuperLU.³ Primerjava napak je prikazana na sliki 13.



Slika 13: Konvergenca MLSM za različne parametre pri reševanju problema (4.3).

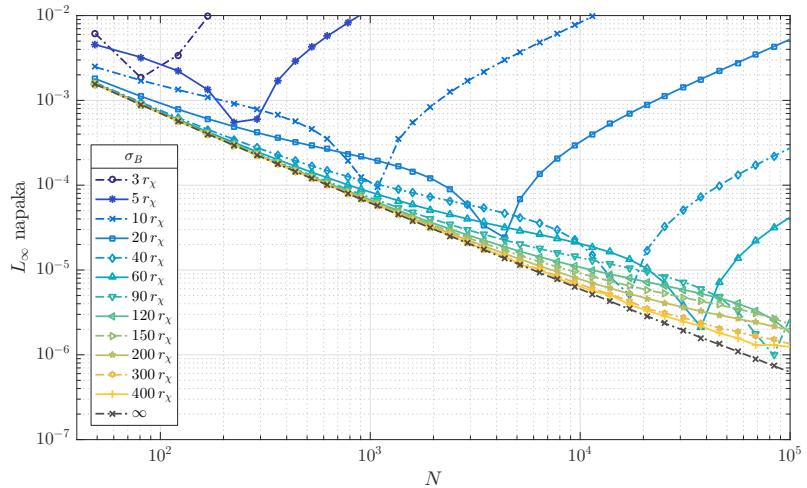
Vidimo, da monomi za $n = 5$ in $n = 9$ konvergirajo linearno z redom 2 (glede na \sqrt{N}), kar vemo tudi iz teorije metode končnih diferenc. Radialne bazne funkcije se na začetku ujemajo z monomi, nato pa pokažejo slabše konvergenčne lastnosti in nelinearno obnašanje, ki je drugačno pri vsakem razredu funkcij. Funkcije, ki uporabljajo več sosedov, začnejo z malenkost nižjo napako, kar pa (morda presenetljivo) velja tudi za monome z $n = 6$. S slike 14, ki prikazuje čas računanja, vidimo, da se uporaba baznih funkcij z $n = 9$ ne splača, saj minimalna pridobljena natančnost ne odtehta časa izvajanja. Vidimo, da se čas izvajanja loči na tri pasove glede na n , ki vsi rastejo linearno z N s konstantnim razmerjem.

³Iterativni BiCGSTAB algoritem je vrnil enake rezultate.



Slika 14: Čas izvajanja za različne izbore parametrov pri reševanju problema (4.3).

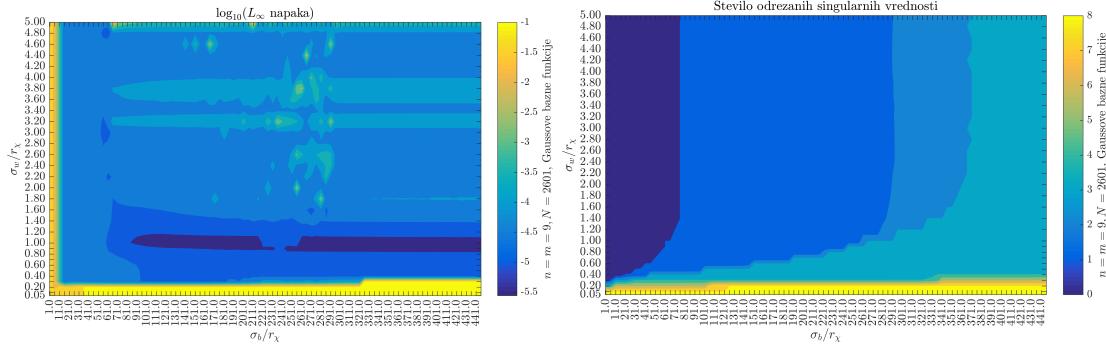
Po trditvi 3.10 aproksimacija z Gaussovimi funkcijami konvergira proti aproksimaciji z monomi. Slika 15 to dejstvo potrjuje. Konvergenčne krivulje za Gaussove funkcije se pri povečevanju parametra oblike čedalje bolj približujejo konvergenčni krivulji monomov. Če imamo pri radialnih baznih funkcijah težavo s konvergenco, lahko poizkusimo povečati parameter oblike. Seveda ga ne moremo povečevati prek vseh meja, saj s tem sistem (3.26) za izračun funkcije oblike z večanjem parametra oblike postaja čedalje bolj občutljiv.



Slika 15: Konvergenčne krivulje za Gaussove funkcije pri čedalje večjem σ in za monome ($\sigma = \infty$). Uporabljeni parametri so $n = m = 5$ in Gaussova utež s parametrom $\sigma_w = \frac{3}{4}r_\chi$.

Da si izboljšamo intuicijo o pomenu parametrov oblike pri uporabi radialnih baznih funkcij in uteži, naredimo še eno analizo, ki bo pomagala pri izbiri parametrov pri bolj zapletenih problemih. Problem (4.3) rešimo za različne izbire parametrov oblike baznih funkcij (σ_b) in Gaussove funkcije uteži (σ_w) ter primerjajmo natančnost rešitve. Graf napake je prikazan na sliki 16a.

Trditev 3.2 pravi, da mora biti v primeru $n = m$ in obrnljive matrike B aproksimacija operatorja (in posledično tudi napaka) neodvisna od izbire uteži. Opomba po



(a) Napaka v odvisnosti od σ_b in σ_w .

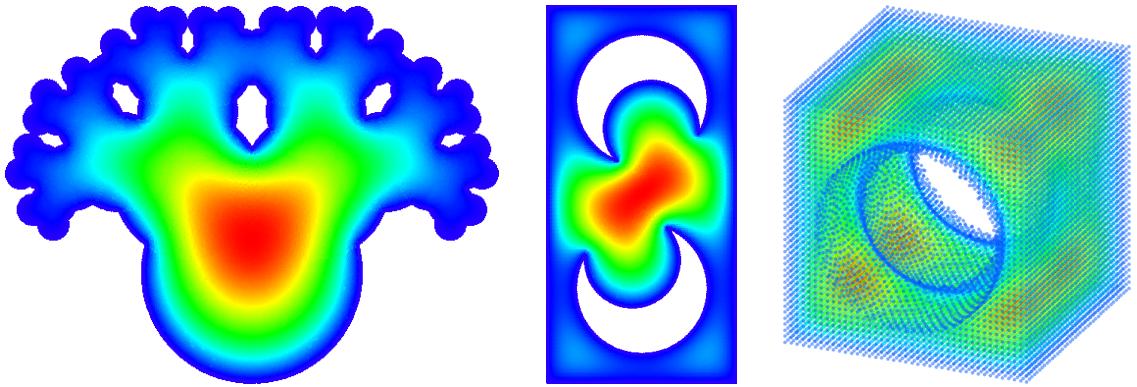
(b) Število odrezanih singularnih vrednosti v odvisnosti od σ_b in σ_w .

Slika 16: Analiza reševanja problema (4.3) z Gaussovimi funkcijami in Gaussovo utežjo pri $n = m = 9$ in $N = 2601$.

trditvi pravi, da numerično to velja, če le utež ni premajhna, da bi bila sama razlog za numerične nestabilnosti. Slika 16a to potrdi; trditev velja v območju, kjer je matrika tudi numerično obrnljiva. Če je utež zelo majhna, manjša od $0.3 r_\chi$, potem se nekatere enačbe v sistemu (3.26) pomnožijo s približno $\exp(-2/0.3^2) \approx 2.23 \cdot 10^{-10}$, kar privede do velike nestabilnosti. Do numerične neobrnljivosti matrike pa lahko pripelje tudi izbira prevelikega parametra σ_b , kajti za $\sigma_b = 70 r_\chi$ so vsi elementi matrike B med 1.000041 in 1. Boljši vpogled v obrnljivost matrike B nam da graf na sliki 16b, ki predstavlja število odrezanih singularnih vrednosti pri SVD razcepu, s pomočjo katerega izračunamo psevdoinverz matrike WB . Aproksimacija je res neodvisna od uteži na predelu, kjer nismo odrezali nobene singularne vrednosti, torej približno na območju $[0, 70] \times [0.3, \infty)$, drugje pa lahko utež vpliva na izračun psevdoinverza in število odrezanih singularnih vrednosti.

Kvaliteta aproksimacije je za σ_b blizu 0 zelo slaba, saj imajo bazne funkcije ne-ničelno vrednost samo v svojem centru. Nato napaka pada z večanjem σ_b , dokler ne pridemo v območje numerične neobrnljivosti matrike B . Tam pride v igro regularizacija v SVD razcepu, ki nam lahko tudi pomaga, kar vidimo v pasu nizke napake okoli $\sigma_w = 1$. Na sliki 16b vidimo, da se z večanjem σ_b reže čedalje več singularnih vrednosti in sčasoma bi aproksimacija spet postala nestabilna. Podobno sliko dobimo pri različnih N , n in ostalih izbirah baznih funkcij. Običajno zato izberemo sorazmerno velik parameter σ_b , okoli $150 r_\chi$, za obliko uteži pa vzamemo približno r_χ .

V dosedanjih analizah smo videli, da se MLSM metoda obnaša enako ali slabše kot metoda končnih differenc na šolskih primerih. Njena prednost leži v splošnosti, saj se jo z lahkoto prilagodi na drugačne domene, v višje dimenzije in na druge operatorje. Na sliki 17 so prikazane rešitve Poissonove enačbe $\Delta u = 1$ s homogenimi robnimi pogoji na bolj zanimivih domenah in v višjih dimenzijah. V dveh dimenzijah so bili za bazne funkcije uporabljeni monomi $\{1, x, y, x^2, y^2\}$ in $n = 5$, v treh pa 9 Gaussovih baznih funkcij z $\sigma_b = 50 r_\chi$ in Gaussovo utežjo s $\sigma_w = r_\chi$ na devetih točkah. V vseh primerih je bilo potrebno poleg parametrov spremeniti le definicijo domene, preostala programska koda je ostala enaka.



Slika 17: Reševanje Poissonove enačbe $\Delta u = 1$ s homogenimi robnimi pogoji na zanimivejših domenah.

4.3 Vpet nosilec

Problem vpetega nosilca (*angl. cantilever beam*) je standarden problem v linearni elastomehaniki in eden osnovnih testnih primerov za raziskovanje obnašanja numeričnih metod. Imejmo tanek nosilec dolžine L in višine D , vpet na desnem koncu, ki ga na levem koncu navzdol potiska sila P na enoto širine. Idealizirana Timoshenkova teorija nosilcev nudi analitično rešitev za ta problem v primeru predpostavke o ravnni napetosti. Napetosti v nosilcu, ki ga postavimo na območje $[0, L] \times [-D/2, D/2]$, so izpeljane v [Sla12, str. 284–289, enačba 7.4.55] in se izražajo kot

$$\sigma_{xx} = \frac{Px}{I}, \quad \sigma_{yy} = 0, \quad \sigma_{xy} = \frac{P}{2I} \left(\frac{D^2}{4} - y^2 \right), \quad (4.9)$$

pomiki pa kot

$$u = \frac{Py(3D^2(\nu + 1) - 4(3L^2 + (\nu + 2)y^2 - 3x^2))}{24EI}, \quad (4.10)$$

$$v = -\frac{P(3D^2(\nu + 1)(L - x) + 4(L - x)^2(2L + x) + 12\nu xy^2)}{24EI}, \quad (4.11)$$

pri čemer je $I = \frac{1}{12}D^3$ vztrajnostni moment preseka nosilca za vrtenje okoli središča, E Youngov modul in ν Poissonovo razmerje. Rešimo primer tudi numerično. Rešimo Navierovo enačbo (2.54) na $\Omega = [0, L] \times [-D/2, D/2]$ s predpisanim pomikom pri $x = L$, danim z analitično rešitvijo (4.10–4.11). Zgornji in spodnji rob naj bosta prosta ($\vec{t} = 0$), na levem robu pa vzemimo navpično parabolično razporejeno silo $\vec{t} = \sigma_{xy}\vec{j}$, dano z analitično rešitvijo (4.9).

Z razliko primera pri izpeljavi metode v razdelku 3.1.2, kjer je bila neznana količina skalarno polje, je tokrat neznana količina vektorsko polje. V tem primeru je najenostavnejši pristop k rešitvi, da obravnavamo enačbo za vektorsko polje kot posamezne sklopljene enačbe za komponente polja, ki so skalarne funkcije. Oglejmo si postopek na primeru dvodimensionalne Navierove enačbe. Vektorsko polje razpišemo po komponentah, za primer dveh dimenzij zapišemo $\vec{u} = (u, v)$. Vektorska oblika stacionarne Navierove enačbe

$$(\lambda + \mu)\nabla(\nabla \cdot \vec{u}) + \mu\nabla^2\vec{u} + \vec{f} = 0 \quad (4.12)$$

se razpiše v sistem dveh enačb

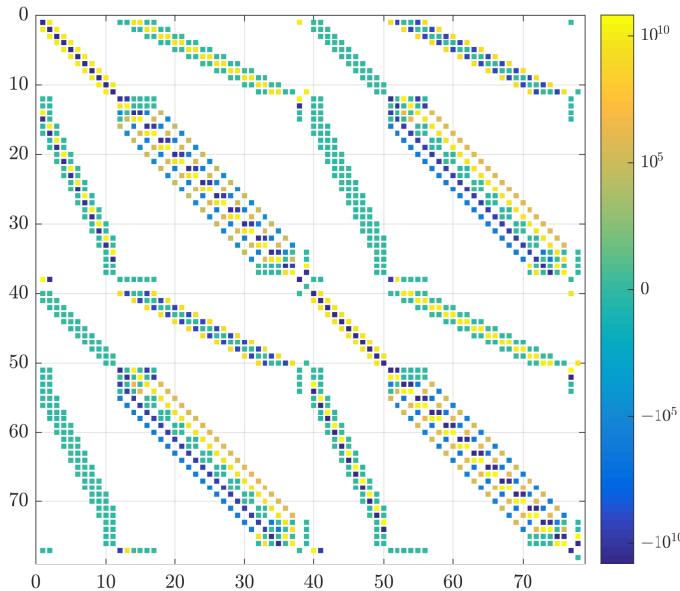
$$(\lambda + \mu) \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 v}{\partial xy} \right) + \mu \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) + f_1 = 0, \quad (4.13)$$

$$(\lambda + \mu) \left(\frac{\partial^2 u}{\partial xy} + \frac{\partial^2 v}{\partial y^2} \right) + \mu \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) + f_2 = 0. \quad (4.14)$$

Enačbe so sklopljene preko operatorja grad div . Če domeno diskretiziramo z N točkami, bomo namesto sistema $N \times N$ za neznano skalarno polje, sedaj reševali sistem $2N \times 2N$ za obe skalarni polji hkrati. Neznanke, ki ustrezajo vektorju \vec{u} , predstavimo numerično kot vektor \mathbf{u} dolžine $2N$, katerega prvih N komponent predstavlja u , preostalih N pa predstavlja v . Enako storimo z \vec{f} in robnimi pogoji. Enačbo $A\mathbf{u} = \mathbf{f}$ lahko sedaj zapišemo po blokih

$$\begin{bmatrix} U1 & V1 \\ U2 & V2 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}. \quad (4.15)$$

Vsako izmed komponent Navierove enačbe napišemo v eno izmed (bločnih) vrstic. Funkcijo oblike za člen $\frac{\partial^2 v}{\partial xy}$ napišemo v blok $V1$, saj nastopa v prvi enačbi in se nahaja v drugi izmed enačb. Podobno aproksimacijo za $\frac{\partial^2 u}{\partial y^2}$ iz (4.13) napišemo v blok $U1$. Če katere elemente napišemo na mesta, kjer so že kakšni neničelni elementi, jih med seboj seštejemo. Primer matrike sistema (4.15) je prikazan na sliki 18. Vidimo, da ima bločno strukturo, kot opisano v (4.15), pri čemer se dodatno znotraj bloků vidijo še različni vzorci na območjih, ki pripadajo notranjim točkam ali točkam na robu.

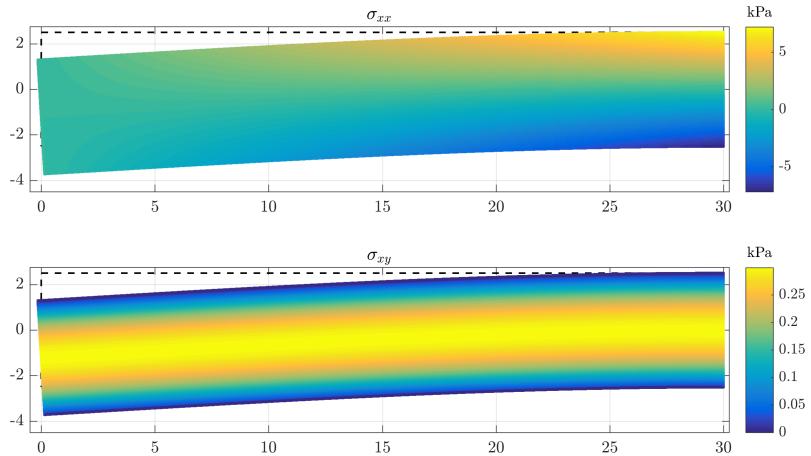


Slika 18: Matrika končnega sistema enačb (4.15) pri reševanju problema vpetega nosilca z 9 monomi in pravokotno diskretizacijo z $N = 39$ točkami

Problem rešimo numerično z uporabo 9 monomov in 9 Gaussovih baznih funkcij z $\sigma_b = 500 r_\chi$, obakrat z Gaussovo utežjo $\sigma_w = r_\chi$. Linearni sistem enačb rešimo z iterativnim BiCGSTAB postopkom z ILUT predpogojevanjem kot opisano v

razdelku 3.3.4. Pri ILUT uporabimo parametra $p = 20$ in $\tau = 10^{-5}$. BiCGSTAB algoritem iteriramo največ 300-krat ali dokler ni ocena napake pod 10^{-13} . Algoritem je v vseh primerih konvergiral.

Za parametre problema izberemo $L = 30\text{ m}$, $D = 5\text{ m}$, $E = 72.1\text{ GPa}$, $\nu = 0.33$, $P = 1000\text{ N/m}$. Numerična rešitev je prikazana na sliki 19. Omeniti je treba še, kako iz numerične rešitve \mathbf{u} dobimo aproksimacijo za σ . Komponente σ so linearne kombinacije odvodov \vec{u} , tako da v vsaki diskretizacijski točki izračunamo funkciji oblike, ki aproksimirata $\frac{\partial}{\partial x}$ in $\frac{\partial}{\partial y}$ in z njuno pomočjo izračunamo $\frac{1}{2}(\text{grad } \vec{u} + \text{grad } \vec{u}^T)$ ter nato preko Hookovega zakona (2.49) dobimo σ .



Slika 19: Numerična rešitev problema vpetega nosilca. Pomiki so zaradi boljše vidnosti povečani za faktor 10^5 . Komponenta σ_{yy} je kot v analitični rešitvi enaka 0.

Izračunamo še, da je največji pomik

$$\max_{x \in \Omega} |\vec{u}(x)| \approx 1.22 \cdot 10^{-5} \text{ m} = 12.2 \mu\text{m} \quad (4.16)$$

in največji gradient pomika

$$\max_{x \in \Omega} \|\text{grad } \vec{u}(x)\| \approx 8.53 \cdot 10^{-7}. \quad (4.17)$$

Predpostavka o majhnosti pomikov in njihovih gradientov torej res velja.

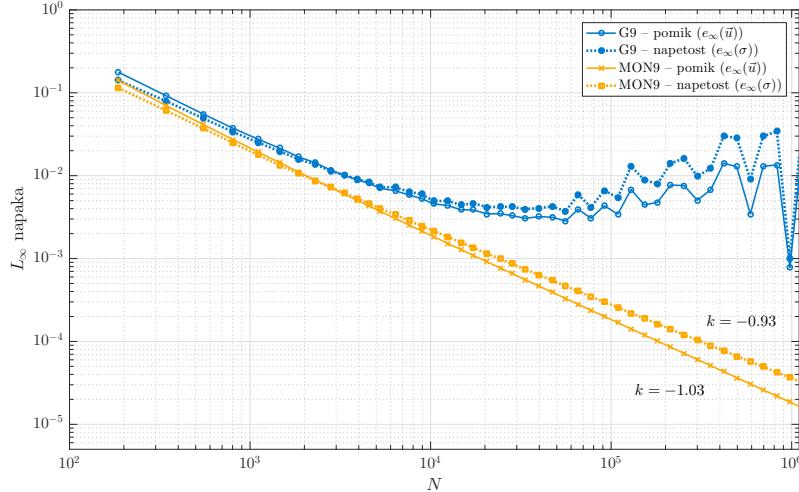
Na sliki 20 je prikazana konvergenca numerične rešitve proti analitični. Napako pomikov in napetosti merimo relativno v L_∞ normi

$$e_\infty(\vec{u}) = \frac{\max_{x \in \mathcal{X}} \{\max\{|u(x) - \hat{u}(x)|, |v(x) - \hat{v}(x)|\}\}}{\max_{x \in \mathcal{X}} \{\max\{|u(x)|, |v(x)|\}\}}, \quad (4.18)$$

$$e_\infty(\sigma) = \frac{\max_{x \in \mathcal{X}} \{\max\{|\sigma_{xx}(x) - \hat{\sigma}_{xx}(x)|, |\sigma_{yy}(x) - \hat{\sigma}_{yy}(x)|, |\sigma_{xy}(x) - \hat{\sigma}_{xy}(x)|\}\}}{\max_{x \in \mathcal{X}} \{\max\{|\sigma_{xx}(x)|, |\sigma_{yy}(x)|, |\sigma_{xy}(x)|\}\}}. \quad (4.19)$$

kjer s strešico označimo izračunane količine, brez strešice pa analitične.

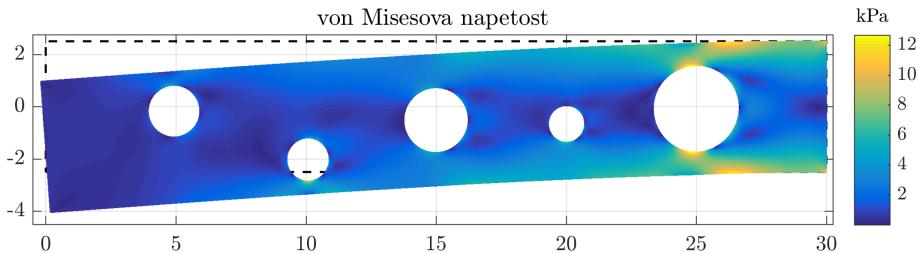
Rezultati so podobni kot pri Poissonovi enačbi na sliki 13. Monomi konvergirajo zelo regularno z redom 2, medtem ko z uporabo Gaussovih baznih funkcij dosegamo nekoliko slabše konvergenčno obnašanje. Vidimo, da konvergirajo tako vrednosti pomikov, kot tudi vrednosti napetosti, pri čemer imamo pri napetostih nekoliko



Slika 20: Konvergenca numerične metode pri reševanju problema vpetega nosilca.

večjo napako in malenkost slabši red, saj jih ocenujemo iz pomikov, kaj zahteva še eno diferenčno aproksimacijo več.

Za demonstracijo splošnosti metode dodajmo v nosilec še nekaj luknenj. Nosilec pritrdimo na desnem koncu ($\vec{u} = 0$) in vlecimo navzdol na levem z enakomerno porazdeljeno silo P enake velikosti kot prej ($\vec{t} = -P/D\vec{j}$). V domeno razporedimo $N = 177617$ diskretizacijskih točk z večjo gostoto okoli robov luknenj, po ustvarjanju domene pa jo še dodatno izboljšamo z algoritmom 3 s 50 iteracijami na šestih sosedih z $F_0 = 10^{-3}$ in $\alpha = 3$. Upogib nosilca, obarvan z vrednostjo von Misesove napetosti, je prikazan na sliki 21.



Slika 21: Numerična rešitev upogiba nosilca z nekaj luknjami. Pomiki so zaradi boljše vidnosti povečani za faktor 10^5 .

Von Misesova napetost se za primer ravninske napetosti izračuna kot

$$\sigma_v = \sqrt{\sigma_{xx}^2 - \sigma_{xx}\sigma_{yy} + \sigma_{yy}^2 + 3\sigma_{xy}^2}, \quad (4.20)$$

in se uporablja v von Misesovem kriteriju plastičnosti. Ta pravi, da material ni več elastičen, ko σ_v preseže neko materialno konstanto σ_0 . Višje vrednosti σ_v , torej nakujujejo, da je material bližje meji svoje elastičnosti in visoke vrednosti σ_v označujejo najverjetnejša mesta za poškodbe. Največji pomik v tem primeru je

$$\max_{x \in \Omega} |\vec{u}(x)| \approx 1.55 \cdot 10^{-5} \text{ m} = 15.5 \mu\text{m}. \quad (4.21)$$

4.4 Hertzev kontaktni problem

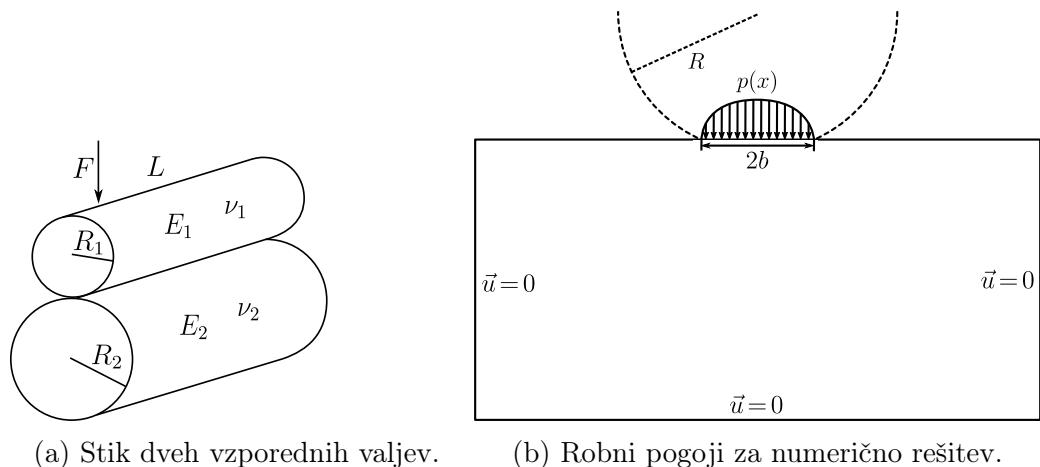
Hertzev kontaktni problem je leta 1882 v svojem članku “Über die Berührung fester elastischer Körper.” [Her82] obravnaval že Heinrich Hertz. Ko dve ukrivljeni telesi z različnima radijema ukrivljenosti staknemo, se na začetku dotikata le v točki ali na premici. Ko ju pritisnemo skupaj z neko silo, se elastično deformirata in med njima se ustvari stična površina. Izračunati želimo napetost v telesih, ki nastanejo kot posledice medsebojnega pritiska ali trenja. Klasična Hertzeva teorija predpostavlja, da je kontakt med telesoma nelepljiv (*angl. non-adhesive*). To pomeni, da je pritisk na kontaktni ploskvi lahko samo pozitiven in da se telesi ne moreta sprijeti med seboj ter da za njuno ločitev ne potrebujemo nobene sile. Kasnejše teorije so veljavne tudi za lepljive kontakte in upoštevajo več parametrov površine, npr. tudi njeno hrapavost. Kljub temu je klasična teorija še vedno aktualna in se uporablja v drugih vejah mehanike, npr. v tribologiji, vedi o trenju, obrabi in mazanju materialov.

Za prvi primer uporabe izpeljane numerične metode na problemu iz elastomehanike obravnavajmo elastičen Hertzev stik valja in polravnine, kot opisan v [Bhu01, str. 122, poglavje 3.2]. Hertzeva teorija je, poleg predpostavke o nelepljivem stiku, osnovana na naslednjih predpostavkah:

1. kontaktni ploskvi sta gladki, ne sovpadata in sta brez trenja,
2. kontaktna površina je majhna v primerjavi z velikostjo teles v kontaktu,
3. vsako od teles lahko v bližini kontakta obravnavamo kot elastičen polprostor,
4. vrzel med telesoma v okolici kontakta je možno aproksimirati z izrazom oblike $Ax^2 + By^2$, kjer sta x in y koordinati v ravnini, tangentni na območje stika.

Točka 4 omeji klasično Hertzevo teorijo na krogle, valje in elipsoide ter njihove limite, ko pošljemo ukrivljenosti proti 0.

Hertzeva teorija za dvodimenzionalni pritisk valja na elastično polravnino se izpelje iz stika med dvema vzporednima valjema dolžine L z radijema R_1 in R_2 , Youngovima moduloma E_1 in E_2 ter Poissonovima razmerjema ν_1 in ν_2 (slika 22a).



Slika 22: Obravnavan Hertzev kontaktni problem.

Območje stika med valjema je širine $2b$, kjer je

$$b = 2\sqrt{\frac{FR}{\pi E^*}}, \quad (4.22)$$

pri čemer je F sila na enoto dolžine, R kombiniran krivinski radij, dan z $\frac{1}{R} = \frac{1}{R_1} + \frac{1}{R_2}$, in E^* kombiniran elastični modul, dan z $\frac{1}{E^*} = \frac{1-\nu_1^2}{E_1} + \frac{1-\nu_2^2}{E_2}$. Pritisak na kontaktno površino je dan z

$$p(x) = \begin{cases} p_0 \sqrt{1 - \frac{x^2}{b^2}}; & |x| < b \\ 0; & \text{sicer} \end{cases}, \quad p_0 = \sqrt{\frac{FE^*}{\pi R}}. \quad (4.23)$$

Pri enem valju pošljemo krivinski radij proti neskončno in problem prevedemo na ravninskega preko predpostavke o ravninski napetosti.

Numerično želimo izračunati pomike in napetosti v materialu, zato rešimo stacionarno Navierovo enačbo (2.54). Rešujemo problem

$$\begin{aligned} (\lambda + \mu) \nabla(\nabla \cdot \vec{u}) + \mu \nabla^2 \vec{u} &= 0 \quad \text{na } \Omega = (-\infty, \infty) \times (-\infty, 0), \\ \vec{t}(x, 0) &= -p(x) \vec{j}, \\ \lim_{x, y \rightarrow \infty} \vec{u}(x, y) &= 0. \end{aligned} \quad (4.24)$$

Analitične rešitve kontaktnih problemov se ponavadi izpelje iz Flamantove rešitve [Sla12, str. 294], ki v posebnem primeru reši problem točkovnega pritiska na polravnino za robni pogoj $\vec{t}(x, 0) = p_0 \delta(x) \vec{j}$. Druge rešitve lahko dobimo s konvolucijo s Flamantovo rešitvijo ali pa z metodo kompleksnih potencialov, kot je to narejeno v [M'E49] še za nekoliko splošnejši problem. Od tam dobimo tudi analitično rešitev v zaprti obliki za napetost, ki se izraža v splošni točki (x, y) s funkcijama m in n ,

$$m^2 = \frac{1}{2} \left(\sqrt{(b^2 - x^2 + y^2)^2 + 4x^2y^2} + b^2 - x^2 + y^2 \right), \quad (4.25)$$

$$n^2 = \frac{1}{2} \left(\sqrt{(b^2 - x^2 + y^2)^2 + 4x^2y^2} - (b^2 - x^2 + y^2) \right), \quad (4.26)$$

kjer je $m = \sqrt{m^2}$ in $n = \operatorname{sgn}(x)\sqrt{n^2}$. Z njuno pomočjo zapišemo napetosti

$$\sigma_{xx} = -\frac{p_0}{b} \left[m \left(1 + \frac{y^2 + n^2}{m^2 + n^2} \right) + 2y \right], \quad (4.27)$$

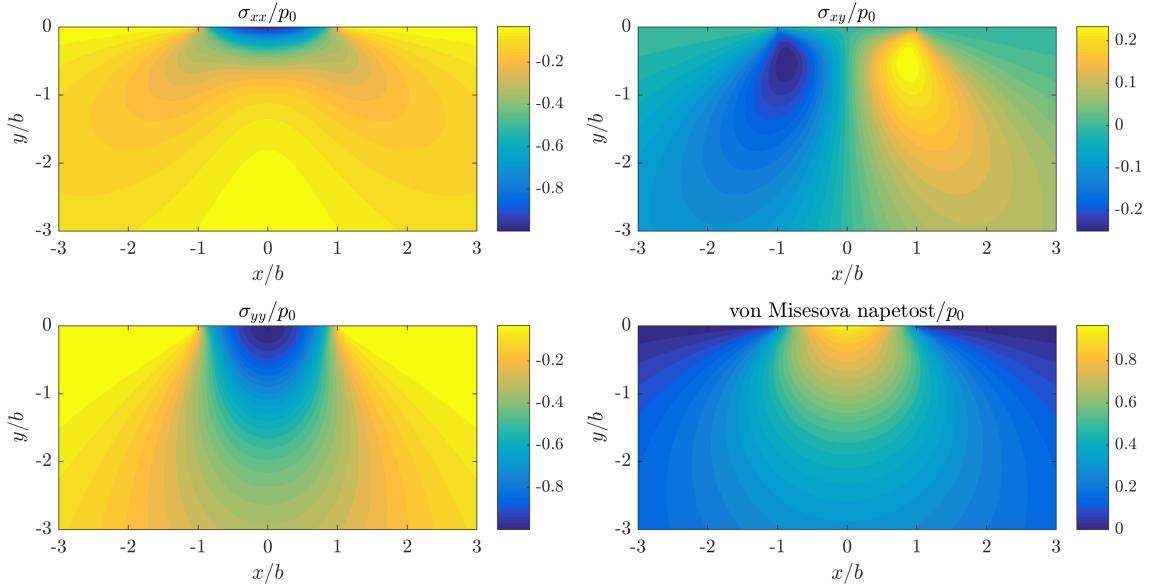
$$\sigma_{yy} = -\frac{p_0}{b} m \left(1 - \frac{y^2 + n^2}{m^2 + n^2} \right), \quad (4.28)$$

$$\sigma_{xy} = \sigma_{yx} = \frac{p_0}{b} n \left(\frac{m^2 - y^2}{m^2 + n^2} \right). \quad (4.29)$$

S pomočjo zgornje rešitve bomo analizirali napako numerične rešitve. Analitične vrednosti napetosti v okolici kontakta so prikazane na sliki 23.

Za normo napake si ponovno izberemo diskretno L_∞ normo, le da tokrat primerjamo napetosti σ_{xx} , σ_{yy} in σ_{xy} . Pošteno je primerjati brezdimenzijske količine σ_{xx}/p_0 , saj so neodvisne od izbire b in p_0 . Za napako med izračunanimi vrednostmi, označenimi s strešico, in pravimi vrednostmi, brez strešice, tako vzamemo

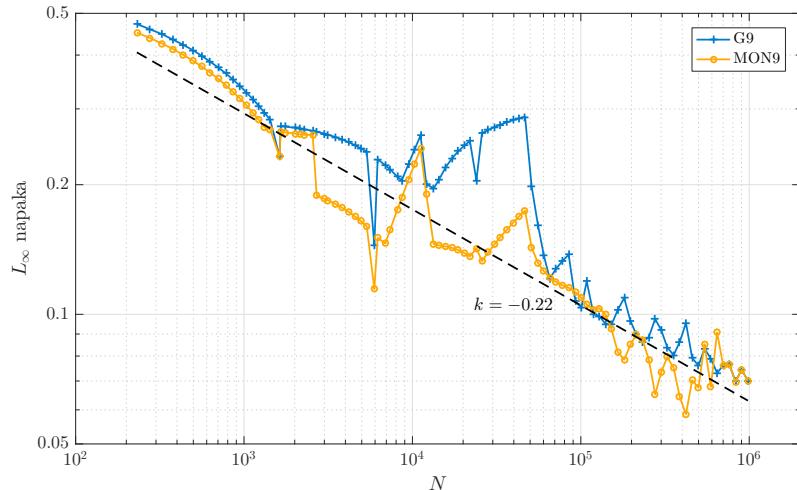
$$e_\infty = \max_{x \in \mathcal{X}} \{ \max \{ |\sigma_{xx}(x) - \hat{\sigma}_{xx}(x)|, |\sigma_{yy}(x) - \hat{\sigma}_{yy}(x)|, |\sigma_{xy}(x) - \hat{\sigma}_{xy}(x)| \} \} / p_0. \quad (4.30)$$



Slika 23: Napetosti pod območjem kontakta med valjem in polravnino.

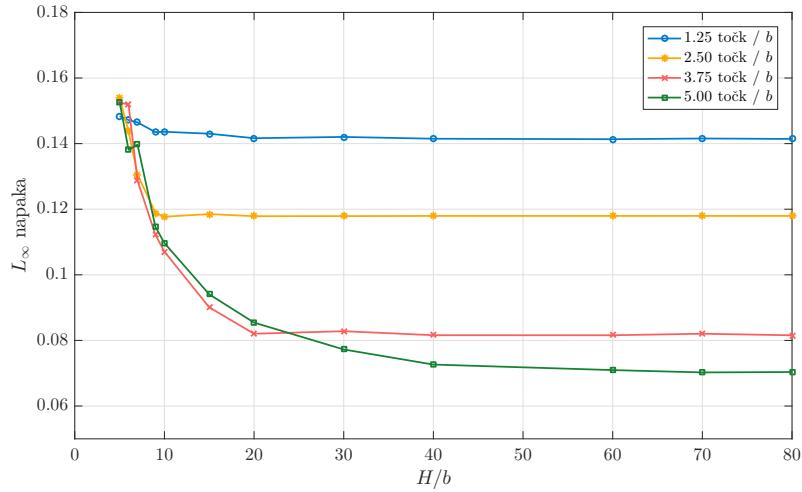
Poskusimo problem rešiti na enak način kot pri problemu vpetega nosilca. Za numerično reševanje neskončno domeno omejimo na $[-H, H] \times [-H, 0]$ za dovolj velik H in na robu postavimo premik na 0, kot prikazano na sliki 22b. Na zgornjem robu domene kot prej zahtevamo predpisano napetost. Za parametre problema vzamemo $F = 543 \text{ N/m}$, $E_1 = E_2 = 72.1 \text{ GPa}$, $\nu_1 = \nu_2 = 0.33$, $R_1 = R = 1 \text{ m}$. Od tod dobimo širino kontakta $b = 0.13 \text{ mm}$ in maksimalni tlak $p_0 = 2.6 \text{ MPa}$. Za H izberimo 10 mm, torej približno 75-krat večjo domeno, kot je pojavi, ki ga opazujemo.

Numerično problem rešimo na dva načina, z uporabo 9 monomov in 9 Gaussovih funkcij z $\sigma_b = 350 r_\chi$. Obakrat uporabimo Gaussovo utež s $\sigma = r_\chi$. Za reševanje linearnega sistema enačb uporabimo iterativni BiCGSTAB algoritem z ILUT predpogojevanjem, s parametrom $p = 20$ in $\tau = 10^{-5}$. BiCGSTAB algoritem iteriramo največ 300-krat ali dokler ni ocena napake pod 10^{-13} . Na sliki 24 je prikazana konvergenca numerične metode.



Slika 24: Konvergenca metode pri reševanju Hertzevega kontaktnega problema (4.24).

Vidimo, da je konvergenca zelo neregularna, nizkega reda in da imamo relativno visoko napako. Visoka napaka je posledica relativno redke diskretizacije, saj tudi pri $N = 10^6$ točkah v domeni to pomeni, da imamo le 10 točk na dolžino b . Neregularnost izhaja iz neregularnosti robnih pogojev, funkcija p namreč ni niti Lipschitzeva in njen vogal pri $\pm b$ je težko dobro aproksimirati. Celotno območje $[-2b, 2b] \times [-2b, 0]$ v bližnji okolini kontakta torej popisuje le slabih 800 točk. Iz grafa konvergencije pri problemu vpetega nosilca na sliki 20 odčitamo, da bi pri takem številu točk imeli napako okoli 0.04, kar je v enakem velikostnem razredu. Večjo napako v našem primeru gre pripisati aproksimaciji robnih pogojev in njihovi neregularnosti. Za zmanjšanje napake imamo dve strategiji, glede na to od kje izvira glavni del napake: ali povečamo domeno ali pa izboljšamo diskretizacijo. Oglejmo si najprej, kaj lahko pridobimo s povečanjem domene, in kako vemo, kdaj smo izbrali dovolj velik H . Na sliki 25 so prikazane napake v odvisnosti od velikosti domene pri različnih gostotah diskretizacij.



Slika 25: Napaka pri različnih gostotah diskretizacij v odvisnosti od velikosti domene.

Vidimo, da na začetku napaka, povzročena z netočnimi robnimi pogoji, pada, in ko pade pod vrednost napake, ki je nastala zaradi diskretizacije, večanje H ne pomaga več. Pri bolj natančnih diskretizacijah moramo vzeti tudi večji H , saj je napaka diskretizacije toliko nižja. Če torej ugotovimo, da se konvergenca ustavi ali ne nadaljuje več z enakim redom kot prej, smo najverjetneje zadeli ob napako zaradi končnih robnih pogojev.

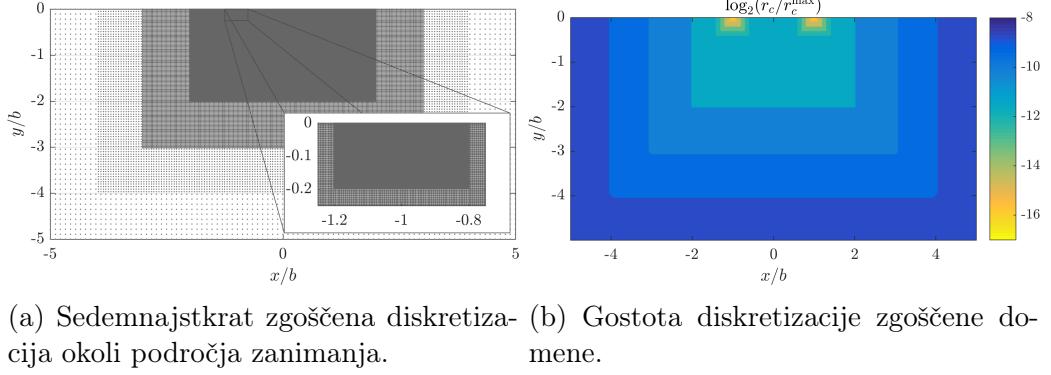
Pri veliki domeni je računsko nemogoče enakomerno gosto večati diskretizacijo. Neposredna rešitev je lokalno zgoščevanje mreže v območju, kjer pričakujemo največjo napako, in sicer v okolini kontakta. Izberimo $H = 1 \text{ m} \approx 75\,000 b$ in zgostimo mrežo kot opisano v razdelku 3.3.2. Za parametre zgoščevanja izberemo $\ell = 8$ in $f = 0.4$. Najprej jo primarno zgostimo v pravokotnikih obliko

$$[-hb, hb] \times [0, hb], \text{ za } h \in \{1000, 500, 200, 100, 50, 20, 10, 5, 4, 3, 2\}, \quad (4.31)$$

nato še sekundarno zgostimo domeno okoli točk $\pm b$ na površini v pravokotnikih obliko

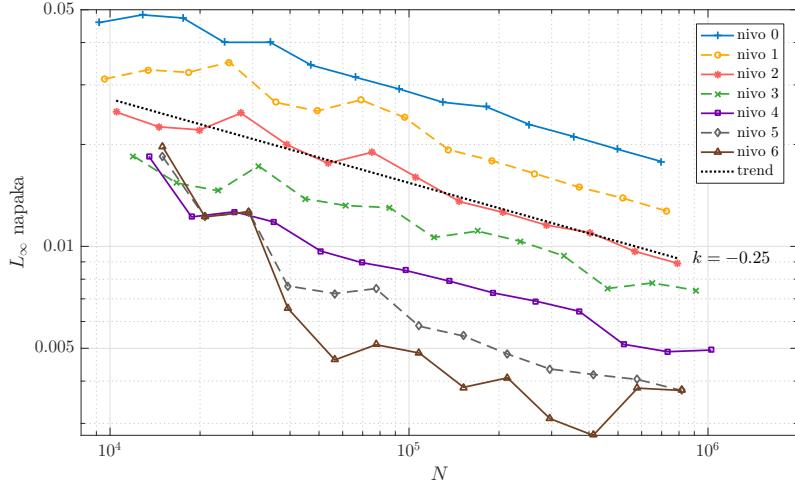
$$[c - hb, c + hb] \times [-hb, 0], \text{ za } c = \pm b \text{ in } h \in \{0.4, 0.3, 0.2, 0.1, 0.05, 0.0025\}. \quad (4.32)$$

Na sliki 26a je prikazan kos tako zgoščene domene, na sliki 26b pa gostota diskretizacije v domeni, podana kot razmerje med razdaljo do najbližjega sosedja in največjo možno razdaljo do najbližjega sosedja.



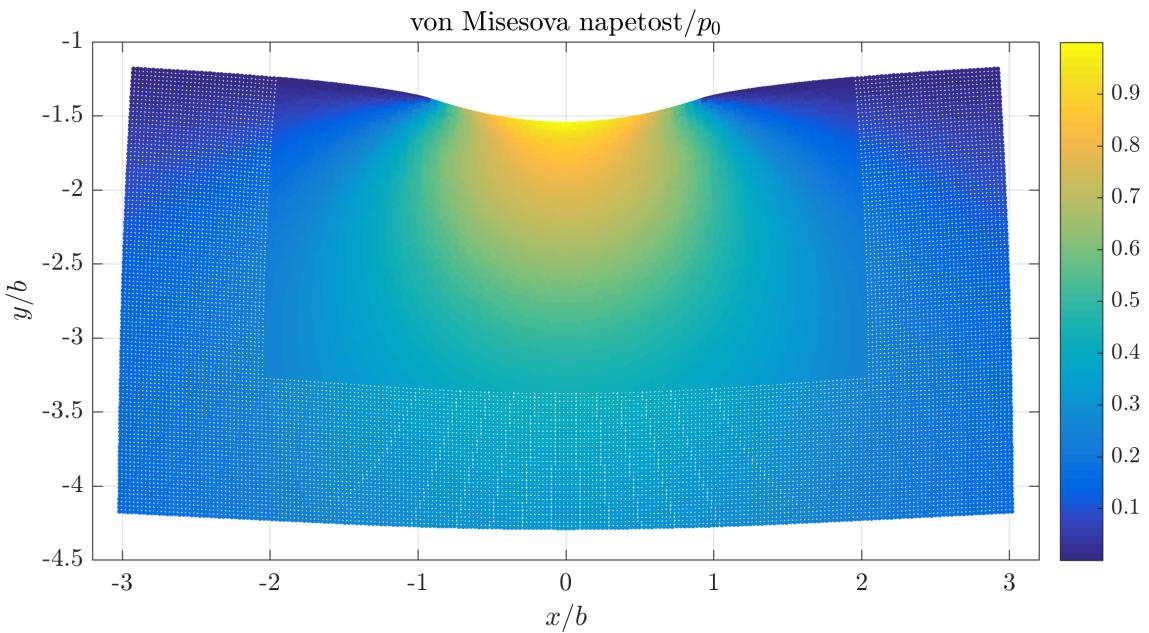
Slika 26: Primer sedemnajstkrat zgoščene domene, s pomočjo katere lahko bolj natančno rešimo Hertzev kontaktni problem (4.24).

Rešimo problem na zgoščeni domeni z uporabo Gaussovih funkcij s parametrom oblike $\sigma_b = 350 r_c$, Gaussovo utežjo s $\sigma_w = r_c$ in $n = m = 9$. Sistem enačb rešimo z iterativnim BiCGSTAB algoritmom z ILUT predpogojevanjem. Uporabimo paramestra $p = 100$ in $\tau = 10^{-6}$. BiCGSTAB algoritem iteriramo največ 1000-krat ali dokler ni ocena napake pod 10^{-15} . Algoritem je v vseh primerih konvergiral. Na sliki 27 je prikazana konvergenca numerične metode pri različnih nivojih sekundarnega zgoščevanja.



Slika 27: Napaka metode pri različnih nivojih sekundarnega zgoščevanja.

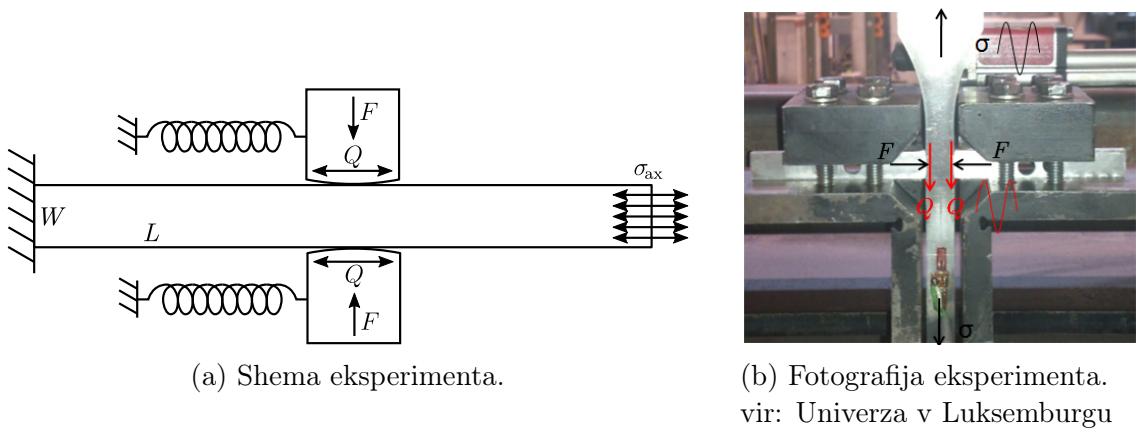
Vidimo, da zgoščevanje zelo pomaga. Že samo pri primarnem zgoščevanju in $N = 10^4$ je napaka manjša, kot smo jo bili sposobni doseči brez zgoščevanja. Vsak dodaten nivo zgoščevanja še izboljša aproksimacijo. Numerična rešitev, pridobljena z najfinješo diskretizacijo, je prikazana na sliki 28.



Slika 28: Numerična rešitev Hertzevega kontaktnega problema (4.24). Zaradi boljše vidnosti so pomiki povečani za faktor 10^5 .

5 Zgled iz mehanike utrujenosti materialov

Oglejmo si še nekoliko bolj konkreten primer elastostatičnega problema iz področja mehanike utrujanja materialov (*angl. fretting fatigue*). Ena izmed oblik utrujanja je obraba materiala zaradi drgnjenja. Drgnjenje je ponavadi oscilatorno gibanje z majhno amplitudo in lahko skrajša življenjsko dobo materiala tudi do 50 % [JKL15] glede na življenjsko dobo, pri kateri oscilacij ne bi bilo. To je vsekakor dovolj dober motiv, da je razumevanje tega pojava pomembna znanstvena disciplina. Eden izmed pogostih eksperimentov za razumevanje tega pojava se izvede tako, da se ploščat kos materiala, ki ga imenujmo *vzorec*, vpne na enem koncu, drug konec pa periodično raztegujemo z napetostjo σ_{ax} [HTW⁺14]. Z drugih dveh strani na vzorec pritiskata dve cilindrični glavi z normalno silo F , ki se tudi gibljeta oscilatorno tangentno na površino materiala in nanj delujeta s tangentno silo Q . Shema eksperimenta je prikazana na sliki 29a, fotografija njegove izvedbe pa na sliki 29b.



Slika 29: Eksperiment za raziskovanje utrujanja materialov.

Z eksperimentom želimo napovedati nastanek in rast razpok. Problem obravnavamo v sklopu mednarodnega projekta (G018916N “Multi-analysis of fretting fatigue using physical and virtual experiments”) z Univerzo v Luksemburgu, Univerzo v Gentu in Inštitutom “Jožef Stefan” tudi numerično [PBT⁺16]. Vloga Inštituta “Jožef Stefan” in Univerze v Luksemburgu je razviti originalno numerično rešitev problema, saj do sedaj uporabljeni komercialni rešitevi ABAQUS® [HKS], ki jo uporabljajo na Univerzi v Gentu, ni zagotovila zadovoljivih rezultatov pri simulaciji širjenja razpoke. Prvi korak k simulaciji eksperimenta je izračun napetosti na površini kontakta. Za referenčno rešitev uporabimo [PBT⁺16], kjer so predstavljeni rezultati simulacije eksperimenta v statičnem stanju maksimalne oscilacije s tangentno silo enako Q in osno napetostjo enako σ_{ax} . Eksperiment ima vodoravno simetrijsko os (slika 29a) zato lahko pri simulaciji obravnavamo samo eno glavo in polovično širino ter rezultate nato prezrcalimo. Poleg tega je debelina materiala majhna in vse sile in napetosti so v ravni materiala, torej problem ustrezza predpostavki o ravninski napetosti in ga lahko prevedemo na dve dimenziji. Problem nas spominja na teorijo Hertzevega kontakta (4.4), le da ima prisotno tudi tangentno silo, osno napetost in končne dimenzije. V [PBT⁺16, razdelek 2] so opisani analitični izrazi za modeliranje napetosti, s katerimi cilindrična glava pritiska na vzorec. Problem poenostavimo in simulirajmo pritisk glave z analitičnimi robnimi pogoji. Označimo krivinski radij

glave z R , širino vzorca z W , dolžino z L in debelino z t . Izraz za širino kontakta je enak kot pri Hertzevem problemu, pri čemer upoštevamo, da je dolžinska gostota sile enaka F/t . Širina polovice kontaktnega območja se tako izraža z

$$a = 2\sqrt{\frac{FR}{t\pi E^*}}, \quad (5.1)$$

normalni pritisk na kontaktno površino pa je dan z

$$p(x) = \begin{cases} p_0 \sqrt{1 - \frac{x^2}{a^2}}; & |x| < a \\ 0; & \text{sicer} \end{cases}, \quad p_0 = \sqrt{\frac{FE^*}{t\pi R}}. \quad (5.2)$$

Pri tem je E^* definiran enako kot pri Hertzevem kontaktu v razdelku 4.4. Na tangentno silo bo imel vpliv tudi koeficient trenja med površinama, ki ga, nekoliko nesrečno, označimo z μ , tako kot Laméjev drugi parameter. Iz konteksta, vrednosti in dimenzij μ bo razvidno, katero izmed količin predstavlja. Za tangentno silo predpostavimo, da je manjša kot mejna sila trenja, ki je dana s Coulombovim zakonom kot μF . V tem primeru se kontakt razdeli na dve območji: območje drsenja in območje lepljenja. Tangentna napetost je dana s

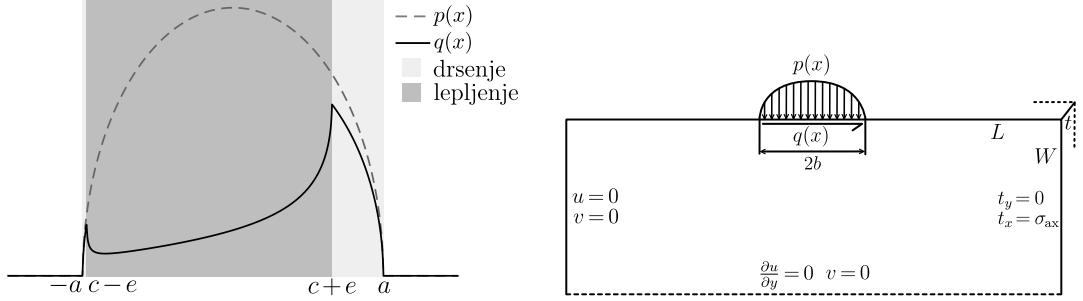
$$\tilde{q}(x) = \begin{cases} -\mu p_0 \left(\sqrt{1 - \frac{x^2}{a^2}} - \frac{c}{a} \sqrt{1 - \frac{x^2}{c^2}} \right); & |x| < c \\ -\mu p_0 \sqrt{1 - \frac{x^2}{a^2}}; & c \leq |x| \leq a \\ 0; & \text{sicer} \end{cases}, \quad (5.3)$$

pri čemer je $c = a\sqrt{1 - \frac{Q}{\mu F}}$ in $\{x; |x| < c\}$ območje lepljenja. Zgornji izraz je možno prilagoditi tudi tako, da je pravilen ob prisotnosti osne napetosti σ_{ax} z uvedbo ekscentričnosti $e = \text{sgn}(Q)\frac{a\sigma_{ax}}{4\mu p_0}$. Z njeno pomočjo izrazimo q s

$$q(x) = \begin{cases} -\mu p_0 \left(\sqrt{1 - \frac{x^2}{a^2}} - \frac{c}{a} \sqrt{1 - \frac{(x-e)^2}{c^2}} \right); & |x - e| < c \\ -\mu p_0 \sqrt{1 - \frac{x^2}{a^2}}; & c \leq |x - e| \text{ in } |x| \leq a \\ 0; & \text{sicer} \end{cases}, \quad (5.4)$$

pri čemer mora veljati $\sigma_{ax} \leq 4 \left(1 - \sqrt{1 - \frac{Q}{\mu F}} \right)$, da ekscentričnost ne povzroči, da bi območje lepljivosti segalo izven območja kontakta. Sedaj imamo vse pripravljeno za numerično reševanje. Rešujemo torej problem

$$\begin{aligned} (\lambda + \mu) \nabla(\nabla \cdot \vec{u}) + \mu \nabla^2 \vec{u} &= 0, & \text{na } \Omega = [-\frac{L}{2}, \frac{L}{2}] \times [-\frac{W}{2}, 0] \\ \vec{t}(x, 0) &= q(x) \vec{i} - p(x) \vec{j}, & \text{(napetost na vrhu)} \\ \vec{u}(-\frac{L}{2}, y) &= 0, & \text{(vpet levo)} \\ \vec{t}(\frac{L}{2}, y) &= 0, & \text{(prost desno)} \\ \vec{u}(x, -\frac{W}{2}) \cdot \vec{j} &= 0, \quad \frac{\partial \vec{u}(x, -t/2) \cdot \vec{i}}{\partial y} = 0, & \text{(simetrija spodaj)} \end{aligned} \quad (5.5)$$



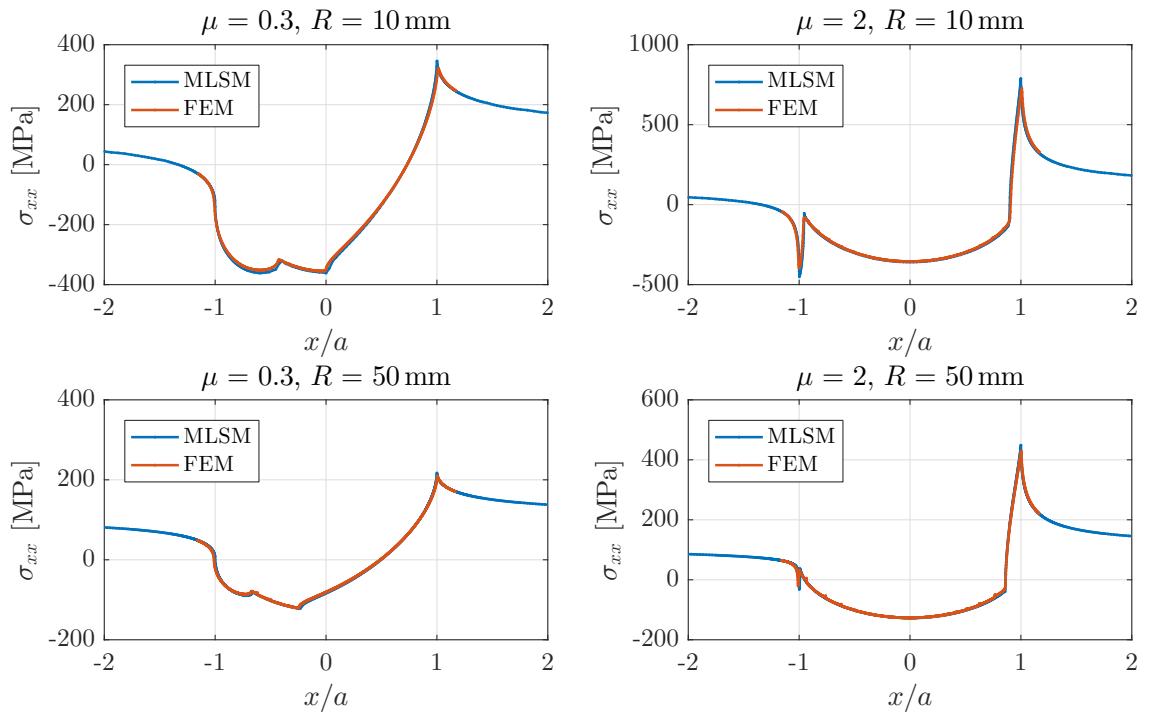
(a) Primer napetosti $p(x)$ in $q(x)$ z označenima območjema drsenja in lepljenja za $R = 50$ mm in $\mu = 0.85$.

(b) Domena in robni pogoji. Označili smo $\vec{u} = (u, v)$, $\vec{t} = (t_x, t_y)$.

Slika 30: Domena in robni pogoji za numerično reševanje problema (5.5).

grafično prikazan na sliki 30b.

V eksperimentu, opisanem v [HTW⁺14], je bil uporabljen aluminij 2420-T3, katerega materialni lastnosti sta $E = 72.1$ GPa in $\nu = 0.33$. Vzorec je bil velikosti $t = 4$ mm, $W = 10$ mm in $L = 40$ mm. Največji izmerjeni sili sta bili $F = 543$ N v smeri $-\vec{j}$ in $Q = 155$ N v smeri $-\vec{i}$, osna napetost pa je bila $\sigma_{ax} = 100$ MPa. Numerična simulacija v [PBT⁺16] je bila narejena za $R = 10$ mm in $R = 50$ mm in tri različne koeficiente trenja $\mu = 0.3$, $\mu = 0.85$ in $\mu = 2$, zato tudi mi izberimo te parametre. Na Hertzevem kontaktnem problemu (4.24) smo se naučili, kako dobro diskretizirati domeno. Tudi tukaj uporabimo enake parametre, le goščenje domene prilagodimo manjšim dimenzijam. Metoda za vse izbire parametrov R in μ s povečevanjem N konvergira proti limitni rešitvi. Na sliki 31 je prikazano ujemanje naše numerične rešitve z rešitvijo, izračunano v [PBT⁺16], za $N = 52659$ pri manjšem in $N = 178897$ pri večjem radiju.



Slika 31: Ujemanje MLSM in FEM rešitve pri analizi kontakta.

Vidimo, da se rešitvi dobro ujemata, a ne povsem, kar je tudi pričakovati, saj je FEM rešitev nastala s simulacijo kontakta, MLSM pa z analitičnimi robnimi pogoji. Ujemanje rešitev kljub temu nakazuje, da je metoda MLSM primerna tudi za reševanje problemov iz kontaktne mehanike.

Posebno zanimanje pa ima vrednost σ_{xx} v špici na desni strani kontaktne površine, saj so raziskave pokazale, da se glavna razpoka, ki pripomore k zlomu materiala, zgodi v neposredni bližini $x = a$. Analitična aproksimacija za največjo vrednost σ_{xx} se izraža kot [PBT⁺16, enačba (7), str. 6]

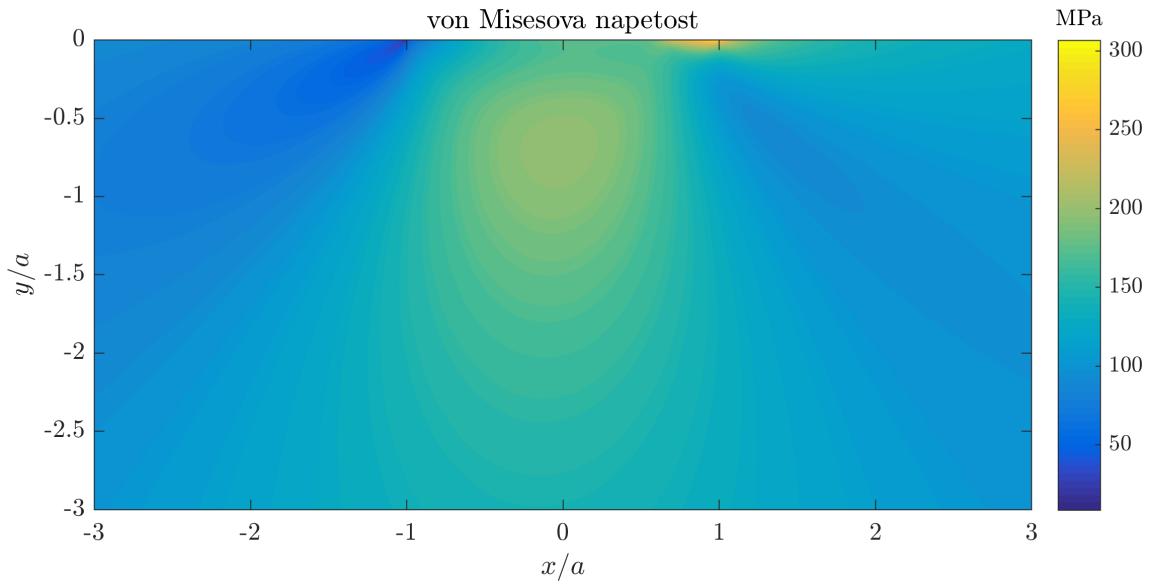
$$\sigma_{xx,\max} = 2p_0 \sqrt{\frac{\mu Q}{F}} + \sigma_{ax}. \quad (5.6)$$

V tabeli 2 je prikazana izračunana vrednost $\sigma_{xx,\max}$ za FEM in MLSM pri različnih N . Tabela vrednosti za FEM je vzeta iz [PBT⁺16, tabela 3]. Vidimo, da vrednosti

μ	minimalna razdalja [μm]	$R = 10 \text{ mm}$		$R = 50 \text{ mm}$	
		MSLM	FEM	MSLM	FEM
0.3	10	325.36	242.57	210.78	182.87
	5	340.99	268.70	216.54	192.96
	2.5	354.46	290.50	214.69	200.09
	1.25	350.33	307.81	217.82	205.33
	0.625	357.84	320.33	216.76	208.99
	0.3125	355.65	329.94	218.35	212.00
	analitična	342.28	342.28	208.35	208.35
0.85	10	479.12	349.91	293.75	254.12
	5	496.08	398.69	305.06	274.64
	2.5	522.88	440.60	307.62	287.93
	1.25	527.76	474.40	311.77	297.38
	0.625	539.26	496.58	314.20	303.26
	0.3125	543.71	513.08	315.77	308.06
	analitična	507.82	507.82	283.90	283.90
2.0	10	686.07	441.20	417.73	336.52
	5	729.38	537.49	435.49	373.08
	2.5	769.58	612.33	438.91	399.72
	1.25	785.98	683.24	448.12	417.17
	0.625	808.62	719.75	447.74	428.46
	0.3125	809.06	750.50	451.95	436.21
	analitična	725.56	725.56	382.10	382.10

Tabela 2: Tabela vrednosti $\sigma_{xx,\max}$ [MPa] za FEM, MLSM in analitično aproksimacijo pri različnih gostotah diskretizacije.

konvergirajo k neki limiti, ki pa se razlikuje pri obeh metodah, vendar je v istem velikostnem razredu. Naša rešitev dosega višje vrednosti $\sigma_{xx,\max}$, kot je tudi razvidno iz grafa na sliki 31. To je zato, ker rešujemo problem z danimi analitičnimi robnimi pogoji, medtem ko FEM rešitev kontakt simulira in slabše aproksimira ostre vogale. Primer izračunane napetosti za problem (5.5) za $\mu = 0.85$ in $R = 50 \text{ mm}$ z $N = 178897$ točkami je prikazan na sliki 32.



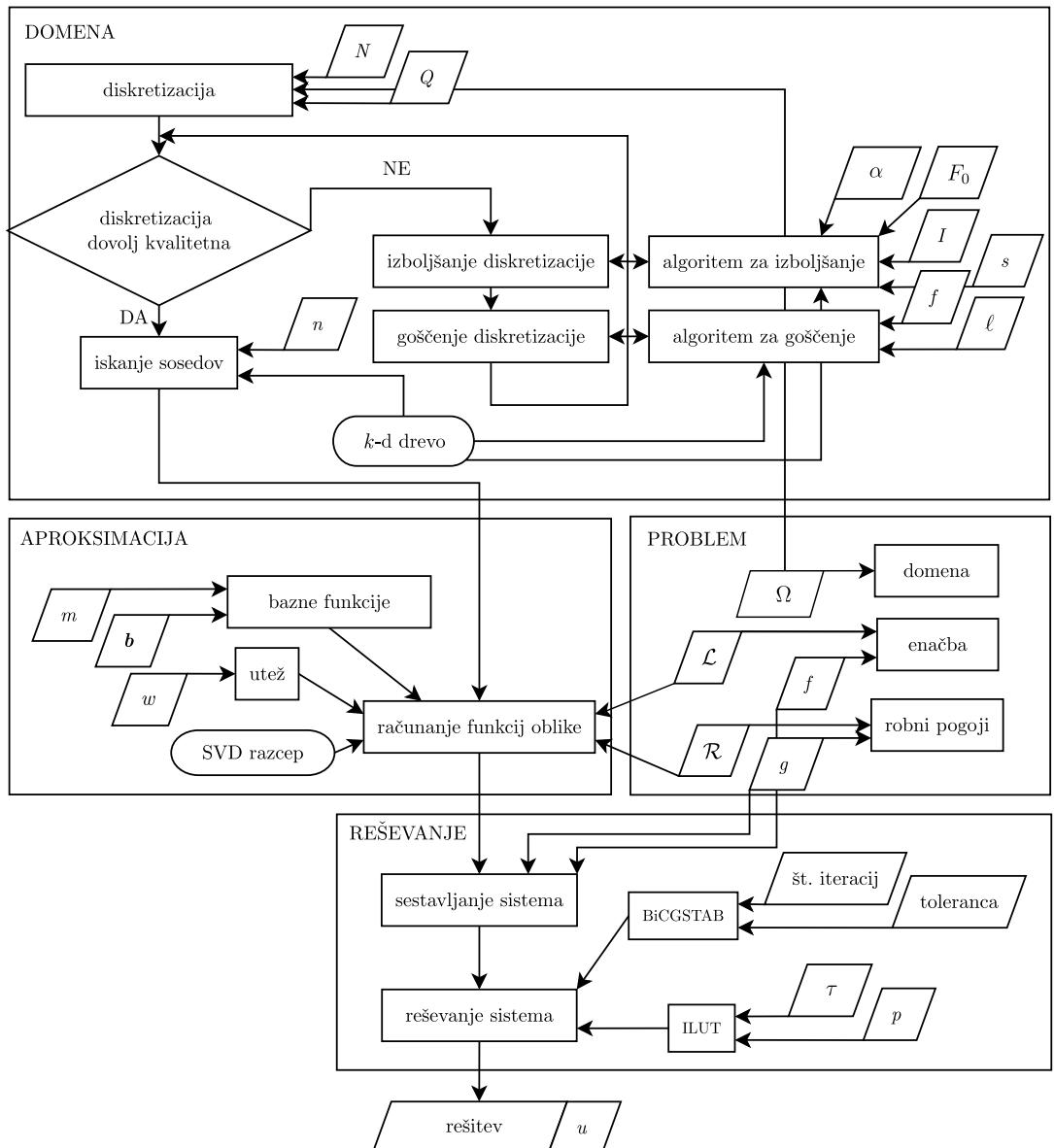
Slika 32: Primer rešitve problema (5.5) za $\mu = 0.65$ in $R = 50$ mm.

Če to sliko primerjamo s sliko 28, na kateri je rešitev Hertzevega kontaktnega problema, opazimo podobno splošno obliko nivojnici napetosti v notranjosti, na površini pa se na sliki 32 močno pozna vpliv tangente sile Q . Ta povzroči zamik nivojnici in koncentracijo napetosti okoli točke $x = a$ na površini, ki je mnogo večja kot drugje v vzorcu.

6 Implementacija

Numerična metoda skupaj z vsemi dodatnimi orodji, za katere ni navedeno drugače, je bila implementirana na oddelku E6 Instituta “Jožef Stefan” v Laboratoriju za vzporedno in porazdeljeno računanje.⁴ Implementirana je v programskem jeziku C++ [Str95] zaradi hitrosti in dovolj močnih jezikovnih konstruktorjev, kot so recimo predloge (*angl. templates*), ki omogočajo elegantno implementacijo. Podoben koncept uporablja knjižnica Eigen [GJ⁺10], uporabljena za numerično linearno algebro. Za iskanje najbližjih sosedov je uporabljena knjižnica ANN [MA98]. Celotna implementacija je prosto dostopna na repozitoriju programske skupine [KKS16] z dokumentacijo, prav tako na voljo na spletu⁵.

Dejanska implementacija v grobem sledi postopku, opisanem v algoritmu 1. Na sliki 33 je prikazan diagram izvedbe izračuna.



Slika 33: Diagram implementacije in izvedbe algoritma 1.

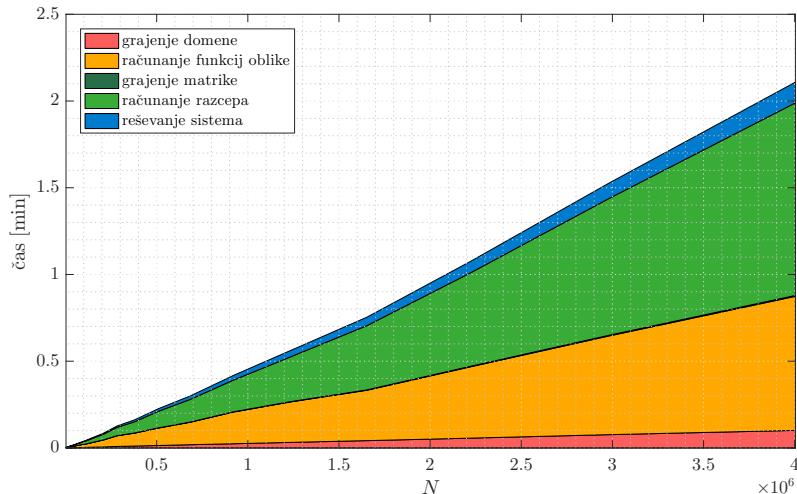
⁴<http://www-e6.ijs.si/ParallelAndDistributedSystems/>

⁵<http://www-e6.ijs.si/ParallelAndDistributedSystems/MeshlessMachine/>

6.1 Vzporedno računanje

Oglejmo si še možnost pospešitve računanja s pomočjo paralelizacije. Vse meritve časa v tem razdelku so narejene na računalniku z dvema centralno procesnima enotama Intel(R) Xeon(R) CPU E5-2620 v3 @ 2.40GHz s po 6 jedri, od katerih vsako jedro podpira dve niti izvajanja, in 64 GiB pomnilnika. Programi so prevedeni z Intelovim C++ prevajalnikom icpc s stikali -std=c++11 -O3 -DNDEBUG na operacijskem sistemu Ubuntu Linux 14.04. Za matematične operacije je uporabljena Intelova knjižnica MKL [Int16]. Pri paralelizaciji je uporabljena Intelova implementacija knjižnice OpenMP [DM98] za paralelizacijo z deljenim pomnilnikom. Za reševanje sistemov linearnih enačb je uporabljena Intelova implementacija sistema Pardiso [SGK⁺10]. Vse knjižnice in programi so verzije 2016.1.150 za Linux.

Oglejmo si najprej, kateri deli postopka so časovno najbolj potratni. Iz časovne analize v razdelku 3.3.5 lahko sklepamo, da je to računanje funkcij oblike, ki traja $O(m^2nN)$ časa, ali pa reševanje sistema. Da bi bolje razumeli, koliko časa stane kateri del, to izmerimo. Na sliki 34 vidimo porazdelitev časa, ki ga posamezni kosi metode potrebujejo pri reševanju Poissonovega robnega problema (4.3).

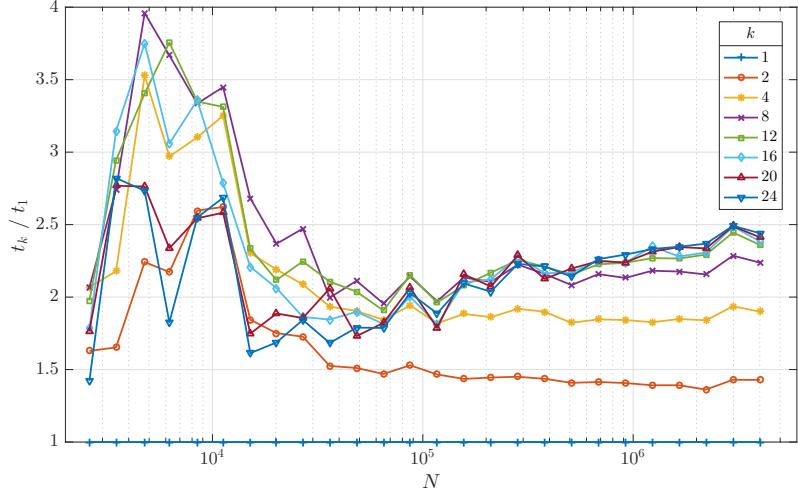


Slika 34: Čas izvajanja posameznih delov programa pri reševanju problema (4.3).

Vidimo, da sta najdražja dela res računanje funkcij oblike in reševanje sistema, zato se ju najbolj splača paralelizirati. Zanko, ki računa funkcije oblik na vrstici 8 algoritma 1 lahko trivialno paraleliziramo, saj so operacije neodvisne med seboj, deljeni podatki pa se ne spreminjajo. Paralelizacija reševanja razpršenega sistema linearnih enačb ni tako enostavna in je področje mnogo raziskav z veliko razvitiimi sistemi za reševanje, kot na primer Pardiso [SGK⁺10], PaStiX [HRR02] in MUMPS [AD⁺00]. Zaradi najlažje integracije z Eigen in dostopnosti v MKL knjižnici za osnovni test izberemo Pardiso. Isti robni problem sedaj rešimo vzporedno s k nitmi in s $t_k(N)$ označimo čas izvajanja programa s k nitmi na problemu z N točkami v domeni. Opazovali bomo faktor pohitritve (angl. speed-up)

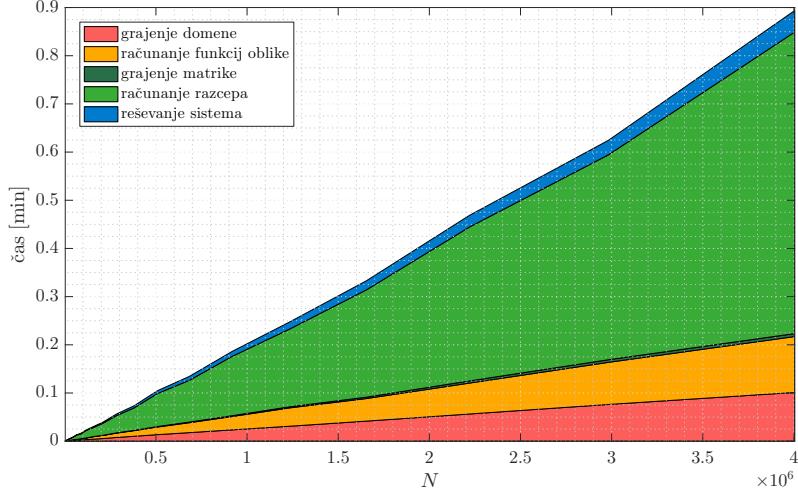
$$f_k(N) = t_k(N)/t_1(N). \quad (6.1)$$

Vrednost $f_3(10^6) = 2.5$ pove, da problem velikosti 10^6 rešimo 2.5-krat hitreje s tremi nitmi izvajanja kot z eno samo. Graf faktorjev pohitritve je prikazan na sliki 35.



Slika 35: Faktorji pohitritve $f_k(N)$ pri reševanju problema (4.3).

Pri problemih z $N < 5 \cdot 10^4$ se program že brez paralelizacije izvede v manj kot sekundi, tako da so meritve pohitritve bolj nezanesljive in sama pohitritev ni tako pomembna, zato glejmo graf za večje N . Vidimo, da se izvajanje pohitri, toda ne zelo, z asimptotičnim faktorjem pospešitve malo manj kot 1.5 za dve jedri, okoli 1.8 za štiri jedra in končnim faktorjem malo pod 2.5 za 12 jedr in več. Zanimivo si je tudi ogledati novo razporeditev časov izvajanja posameznih kosov, prikazano za primer $k = 12$ na sliki 36.



Slika 36: Čas izvajanja posameznih delov programa pri reševanju problema (4.3) z 12 nitmi.

Pričakovano se poveča delež, ki ga porabimo za grajenje domene, relativno pa se precej poveča delež časa, ki ga porabimo za razcep matrike, saj se ta paralelizira slabše kot grajenje funkcij oblike. Iz grafov na slikah 34 in 36 preberemo, da je faktor pohitritve samo za funkcije oblike približno 6.4, za razcep matrike pa 1.7.

Kljub slabim faktorjem pohitritve pri paralelizaciji, ta še vedno vsaj dvakrat skrajša čas izvajanja, brez veliko truda ali sprememb v izvorni kodi, tako da ni razloga, da je ne bi uporabili.

7 Zaključek

V magistrskem delu smo si ogledali klasično teorijo elastomehanike in izpeljali Navierovo enačbo, ki opisuje deformacijo teles pod zunanjimi obremenitvami. Pokazali smo obstoj in enoličnost šibkih rešitev Navierove enačbe. Samega reševanja smo se lotili numerično in sicer s splošno MLSM metodo, za katero smo pokazali, da v posebnih primerih preide v metodo končnih diferenc. Vpeljali smo tudi algoritem za zgoščevanje in prilaganje diskretizacije poljubnim oblikam ter demonstrirali njuno uporabnost na več neregularnih domenah.

Metodo smo najprej testirali na preprosti Poissonovi enačbi, kjer smo analizirali obnašanje glede na uporabo različnih baznih funkcij in uteži ter različnega števila sosedov. V nadaljevanju smo rešili dva klasična primera iz elastostatike: problem vpetega nosilca ter Hertzev kontaktni problem, na katerem smo dodatno demonstrirali možnost goščenja diskretizacije, saj smo domeno tako zgostili, da je razdalja med sosednjimi vozlišči na najbolj zgoščenem delu 2^{17} -krat manjša kot na najmanj zgoščenem. Splošnost MLSM glede postavitve diskretizacijskih točk smo predstavili preko rešitve navrtanega vpetega nosilca, kjer smo morali opisati neregularno domeno ter zgostiti točke v okolini lukanj. Za konec smo rešili še primer, motiviran s strani mehanike utrujanja materialov in pokazali ujemanje metode z referenčno rešitvijo. Nekateri rezultati magistrskega dela so bili tudi poslani v objavo [SK17].

Nadaljnje delo bo usmerjeno v razvoj rešitvenega postopka za simulacijo rasti razpoke. V ta namen bomo MLSM obogatili z adaptacijo baznih funkcij, kar bomo potrebovali za opis konice razpoke. Za opis razpoke bomo prav tako potrebovali cenilko napake rešitve, saj ne bo več v naprej znano, kje lahko pričakujemo največje napake. Pri računanju funkcij oblike bomo problem v posebnih, a pogostih primerih, zastavili preko reševanja sistema in tako pridobili na času reševanja, ki bo pri simulacijah rasti razpoke postal pomemben faktor. V ta namen se bomo posvetili tudi implementaciji paralelnih rešitvenih postopkov, tako na arhitekturah s skupnim pomnilnikom, kot tudi na distribuiranih arhitekturah.

Literatura

- [AAN12] J. Amani, M. Afshar in M. Naisipour, *Mixed discrete least squares meshless method for planar elasticity problems using regular and irregular nodal distributions*, Eng. Anal. Boundary Elem. **36** (2012) 894–902, doi:10.1016/j.enganabound.2011.09.012.
- [AD⁺00] P. R. Amestoy, I. S. Duff in dr., *MUMPS: a general purpose distributed memory sparse solver*, v: International Workshop on Applied Parallel Computing, Springer, str. 121–130, doi:10.1007/3-540-70734-4_16.
- [AF03] R. A. Adams in J. J. Fournier, *Sobolev spaces*, Pure and Applied Mathematics **140**, Academic press, 2003.
- [Aut] Autodesk Inc., *Autodesk FEA*, [ogled 11. 8. 2017], dostopno na <https://www.autodesk.com/solutions/finite-element-analysis>.
- [Bhu01] B. Bhushan (urednik), *Modern Tribology Handbook*, CRC Press, Mechanics & Materials Science **1**, pogl. 3, 2001, str. 121–162, doi:10.1201/9780849377877.ch3.
- [BKL06] A. Beygelzimer, S. Kakade in J. Langford, *Cover trees for nearest neighbor*, v: Proceedings of the 23rd international conference on Machine learning, ACM, str. 97–104, doi:10.1145/1143844.1143857.
- [Buh00] M. D. Buhmann, *Radial basis functions*, Acta Numerica **9** (2000) 1–38.
- [BW80] H. Brézis in S. Wainger, *A note on limiting cases of Sobolev embeddings and convolution inequalities*, Communications in Partial Differential Equations **5** (1980) 773–789, doi:10.1080/03605308008820154.
- [Cam03] Cambridge University Engineering Department, *Materials Data Book*, Cambridge University, Cambridge, 2003, dostopno na <http://www-mdp.eng.cam.ac.uk/web/library/enginfo/cueddatabooks/materials.pdf>.
- [Cia10] P. G. Ciarlet, *On Korn's inequality*, Chin. Ann. Math. Ser. B **31** (2010) 607–618, doi:10.1007/s11401-010-0606-3.
- [CLE06] Y. Chen, J. D. Lee in A. Eskandarian, *Meshless methods in solid mechanics*, Springer, New York, NY, 2006.
- [DAM14] M. Dehghan, M. Abbaszadeh in A. Mohebbi, *The numerical solution of nonlinear high dimensional generalized Benjamin–Bona–Mahony–Burgers equation via the meshless method of radial basis functions*, Comput. Math. Appl. **68** (2014) 212–237, doi:10.1016/j.camwa.2014.05.019.
- [Dav06] T. A. Davis, *Direct methods for sparse linear systems*, SIAM, 2006.
- [DM98] L. Dagum in R. Menon, *OpenMP: an industry standard API for shared-memory programming*, IEEE computational science and engineering **5** (1998) 46–55, doi:10.1109/99.660313, dostopno na <http://www.openmp.org/>.

- [DNPV12] E. Di Nezza, G. Palatucci in E. Valdinoci, *Hitchhiker's guide to the fractional Sobolev spaces*, Bulletin des Sciences Mathématiques **136** (2012) 521–573, doi:10.1016/j.bulsci.2011.12.004.
- [EH87] W. Evans in D. Harris, *Sobolev embeddings for generalized ridged domains*, Proceedings of the London Mathematical Society **3** (1987) 141–175, doi:10.1112/plms/s3-54.1.141.
- [ER00] D. Edmunds in J. Rákosník, *Sobolev embeddings with variable exponent*, Studia Mathematica **3** (2000) 267–293.
- [GJ⁺10] G. Guennebaud, B. Jacob in dr., *Eigen v3*, 2010, [ogled 16. 6. 2017], dostopno na <http://eigen.tuxfamily.org>.
- [Gon10] Á. González, *Measurement of areas on a sphere using Fibonacci and latitude-longitude lattices*, Math. Geosci. **42** (2010) 49–64, doi:10.1007/s11004-009-9257-x.
- [Gur82] M. E. Gurtin, *An introduction to continuum mechanics*, Mathematics in Science and Engineering **158**, Academic Press, 1982.
- [Her82] H. Hertz, *Über die Berührung fester elastischer Körper.*, Journal für die reine und angewandte Mathematik **92** (1882) 156–171.
- [HH95] J. Hattel in P. Hansen, *A control volume-based finite difference method for solving the equilibrium equations in terms of displacements*, Appl. Math. Modell. **19** (1995) 210–243, doi:10.1016/0307-904x(94)00015-x.
- [Hje07] K. D. Hjelmstad, *Fundamentals of structural mechanics*, Springer Science & Business Media, 2007.
- [HKS] Hibbit, Karlsson in Sorensen, *ABAQUS*, [ogled 21. 7. 2017], dostopno na <https://www.3ds.com/products-services/simulia/products/abaqus/>.
- [HN04] J. Hannay in J. Nye, *Fibonacci numerical integration on a sphere*, J. Phys. A: Math. Gen. **37** (2004) 11591–11601, doi:10.1088/0305-4470/37/48/005.
- [HRR02] P. Hénon, P. Ramet in J. Roman, *PASTIX: a high-performance parallel direct solver for sparse symmetric positive definite systems*, Parallel Comput. **28** (2002) 301–321, doi:10.1016/s0167-8191(01)00141-7.
- [HTW⁺14] R. Hojjati-Talemi, M. A. Wahab in dr., *Prediction of fretting fatigue crack initiation and propagation lifetime for cylindrical contact configuration*, Tribol. Int. **76** (2014) 73–91, doi:10.1016/j.triboint.2014.02.017.
- [Int16] Intel, *Intel math kernel library, version 11.3*, 2016, [ogled 20. 7. 2017], dostopno na <https://software.intel.com/en-us/mkl>.
- [JKL15] H.-K. Jeung, J.-D. Kwon in C. Y. Lee, *Crack initiation and propagation under fretting fatigue of inconel 600 alloy*, J. Mech. Sci. Technol. **29** (2015) 5241–5244, doi:10.1007/s12206-015-1124-8.

- [KF75] E. A. Kearsley in J. Fong, *Linearly independent sets of isotropic Cartesian tensors of ranks up to eight*, J. Res. Natl Bureau of Standards Part B: Math. Sci. B **79** (1975) 49–58, doi:10.6028/jres.079b.005.
- [KF07] A. M. Kibriya in E. Frank, *An empirical comparison of exact nearest neighbour algorithms*, v: European Conference on Principles of Data Mining and Knowledge Discovery, Springer, str. 140–151, doi:10.1007/978-3-540-74976-9_16.
- [KKS16] G. Kosec, M. Kolman in J. Slak, *Utilities for solving PDEs with meshless methods*, 2016, [ogled 11. 8. 2017], dostopno na <https://gitlab.com/e62Lab/e62numcodes.git>.
- [Kos16] G. Kosec, *A local numerical solution of a fluid-flow problem on an irregular domain*, Adv. Eng. Software (2016), doi:10.1016/j.advengsoft.2016.05.010.
- [Koz08] J. Kozak, *Numerična analiza*, DMFA – založništvo, 2008.
- [Kre89] E. Kreyszig, *Introductory Functional Analysis with Applications*, Wiley, New York, 1989.
- [KŠ11] G. Kosec in B. Šarler, *H-adaptive local radial basis function collocation meshless method*, Computers Materials and Continua **26** (2011) 227, doi:doi:10.3970/cmc.2011.026.227.
- [Lak87] R. Lakes, *Foam structures with a negative Poisson's ratio*, Science **235** (1987) 1038–1041, doi:10.1126/science.235.4792.1038.
- [LC09] L. Lebedev in M. Cloud, *Introduction to Mathematical Elasticity*, World Scientific, 2009.
- [LCC13] M. Li, W. Chen in C. Chen, *The localized RBFs collocation methods for solving high dimensional PDEs*, Eng. Anal. Boundary Elem. **37** (2013) 1300–1304, doi:10.1016/j.enganabound.2013.06.001.
- [LG01] G. R. Liu in Y. T. Gu, *A point interpolation method for two-dimensional solids*, Int. J. Numer. Methods Eng. **50** (2001) 937–951, doi:10.1002/1097-0207(20010210)50:4<937::AID-NME62>3.0.CO;2-X.
- [Li05] X. S. Li, *An overview of SuperLU: Algorithms, implementation, and user interface*, ACM Trans. Math. Software **31** (2005) 302–325, doi:10.1145/1089014.1089017.
- [LL02] S. Li in K. W. Liu, *Meshfree and particle methods and their applications*, Appl. Mech. Rev. **55** (2002) 1–34, doi:10.1115/1.1431547.
- [MA98] D. M. Mount in S. Arya, *ANN: library for approximate nearest neighbour searching*, 1998, doi:10.1.1.91.8209, [ogled 16. 6. 2017], dostopno na <https://www.cs.umd.edu/~mount/ANN/>.
- [Maz85] V. G. Maz'ya, *Sobolev Spaces*, Springer Series in Soviet Mathematics, Springer Berlin Heidelberg, 1985.

- [MC94] W. J. Morokoff in R. E. Caflisch, *Quasi-random sequences and their discrepancies*, SIAM J. Sci. Comput. **15** (1994) 1251–1279, doi:10.1137/0915077.
- [M'E49] E. M'Ewen, *Stresses in elastic cylinders in contact along a generatrix (including the effect of tangential friction)*, The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science **40** (1949) 454–459, doi:10.1080/14786444908521733.
- [Moo91] A. W. Moore, *An introductory tutorial on kd-trees*, 1991, doi:10.1.1.28.6468.
- [Mv15] B. Mavrič in B. Šarler, *Local radial basis function collocation method for linear thermoelasticity in two dimensions*, Int. J. Numer. Methods Heat Fluid Flow **25** (2015) 1488–1510, doi:10.1108/hff-11-2014-0359.
- [Omo89] S. M. Omohundro, *Five balltree construction algorithms*, teh. por., International Computer Science Institute, 1989.
- [PBT⁺16] K. Pereira in dr., *On the convergence of stresses in fretting fatigue*, Materials **9** (2016), doi:10.3390/ma9080639.
- [RR00] B. Rao in S. Rahman, *An efficient meshless method for fracture analysis of cracks*, Comput. Mech. **26** (2000) 398–408, doi:10.1007/s004660000189.
- [Saa94] Y. Saad, *ILUT: A dual threshold incomplete LU factorization*, Numerical linear algebra with applications **1** (1994) 387–402, doi:10.1002/nla.1680010405.
- [Saa03] Y. Saad, *Iterative methods for sparse linear systems*, SIAM, 2003.
- [Sch95] R. Schaback, *Error estimates and condition numbers for radial basis function interpolation*, Adv. Comput. Math. **3** (1995) 251–264, doi:10.1007/bf02432002.
- [SGK⁺10] O. Schenk in dr., *PARDISO solver project*, 2010, [ogled 20. 7. 2017], dostopno na <http://www.pardiso-project.org>.
- [SK17] J. Slak in G. Kosec, *Refined meshless local strong form solution of cauchy-navier equations on an irregular domain*, 2017, poslano v objavo v Engineering Analysis with Boundary Elements.
- [Sla12] W. S. Slaughter, *The linearized theory of elasticity*, Springer Science & Business Media, 2012.
- [Slo91] N. J. A. Sloane, *The On-Line Encyclopedia of Integer Sequences*, Sequence A005043, 1991, [ogled 9. 7. 2017], dostopno na <http://oeis.org/A005043>.
- [Sob38] S. Soboleff, *Sur un théorème d'analyse fonctionnelle*, Matematicheskii Sbornik **46** (1938) 471–497, dostopno na <http://www.mathnet.ru/links/b3fb5ec17551fb44064f794cb7305b0d/sm5759.pdf>.

- [Sol] SolidWorks Corporation, *SOLIDWORKS Simulation*, [ogled 11. 8. 2017], dostopno na <http://www.solidworks.com/sw/products/simulation/finite-element-analysis.htm>.
- [Str95] B. Stroustrup, *The C++ programming language*, Pearson Education India, 1995.
- [TK15] R. Trobec in G. Kosec, *Parallel scientific computing: theory, algorithms, and applications of mesh based and meshless methods*, SpringerBriefs in Computer Science, Springer, 2015.
- [Vor92] H. A. Van der Vorst, *Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems*, SIAM Journal on Scientific and Statistical Computing **13** (1992) 631–644, doi:10.1137/0913035.
- [Wol17] Wolfram Research, *Wolfram Mathematica, version 11*, 2017, [ogled 11. 8. 2017], dostopno na <http://www.wolfram.com/mathematica/>.
- [ZT00] O. C. Zienkiewicz in R. L. Taylor, *The Finite Element Method: Solid Mechanics*, The Finite Element Method **2**, Butterworth-Heinemann, 2000.