UNIVERZA V LJUBLJANI
FAKULTETA ZA MATEMATIKO IN FIZIKO

Matematika – 2. stopnja

Jure Slak

TBD

Magistrsko delo

Mentor: doc. dr. George Mejak
Somentor: dr. Gregor Kosec

Ljubljana, 2017

Izjava o avtorstvu?

Zahvala?

Kazalo vsebine

Kazalo slik

Kazalo algoritmov

Program dela?

TBD

Povzetek

TBD

TBD

Abstract

TBD

Math. Subj. Class. (2010):
Ključne besede: ??
Keywords: ??

1 Uvod

Motivacija, pregled naloge

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) v mehaniki bomo označevali z malimi latinskimi ali grškimi črkami, npr. $t, \sigma \in \mathbb{R}^3 \otimes \mathbb{R}^3$. Komponente 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^T bomo označevali transpozicijo tenzorja ali matrike, definirano z $\vec{a} \cdot t\vec{b} = t^\mathsf{T}\vec{a} \cdot \vec{b}$. Z oznako $\langle \vec{a}, \vec{b}, \vec{c} \rangle$ bomo označevali mešani produkt vektorjev $\vec{a}$, $\vec{b}$ in $\vec{c}$, definiran kot $\langle \vec{a}, \vec{b}, \vec{c} \rangle = (\vec{a} \times \vec{b}) \cdot \vec{c}$.

Divergenco, gradient in Laplaceov operator bomo označevali z div, grad, $\triangle$ ali pa z $\nabla \cdot$, ∇ in ∇^2.

V razdelkih 3 in 5 bomo imeli opravka tudi z vektorji in matrikami splošnih dimenzij, ki ne predstavljajo mehaničnih količin, ampak samo končno zaporedje skalarjev. Take vektorje bomo označevali odebeljeno, npr. $\boldsymbol{u}, \boldsymbol{\varphi} \in \mathbb{R}^n$, matrike pa z velikimi tiskanimi črkami $A, B \in \mathbb{R}^{m \times n}$.

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 potrebujemo, ponavadi do drugega reda.

Prostor tenzorjev drugega reda opremimo s skalarnim produktom

$$s : t = s_{ij} t_{ij} = \operatorname{tr}(s^\mathsf{T} t).$$

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

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

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

$$s : t = \operatorname{tr}(s^\mathsf{T} t) = -\operatorname{tr}(s t^\mathsf{T}) = -\operatorname{tr}(s^\mathsf{T} t) = -s : t,$$

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

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

$$\operatorname{div}(t) \cdot \vec{a} = \operatorname{div}(t^{\mathsf{T}}\vec{a}).$$

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

Zakaj je ravno to prava definicija, nam 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*

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

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

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

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. □

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

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

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 $x \mapsto \vec{v}(x)$ in ga označujemo tudi kot $\frac{\partial \vec{v}}{\partial x}$.

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

$$\triangle\vec{v} = \operatorname{div} \operatorname{grad} \vec{v}.$$

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

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

$$\operatorname{tr} \operatorname{grad} \vec{v} = \operatorname{div} \vec{v}.$$

Dokaz.

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

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

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

Dokaz.

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

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

Trditev 1.9. *Za poljubno skalarno polje φ velja*

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

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*

$$\operatorname{div}(t^{\mathsf{T}}\vec{v}) = \vec{v}\cdot\operatorname{div}t + t : \operatorname{grad}\vec{v}.$$

Dokaz.

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

1.3 Osnovni izreki funkcionalne analize

Ponovimo še nekaj osnovnih definicij in izrekov funkcionalne analize, ki jih bomo potrebovali pri obravnavi enoličnosti in obstoje rešitve linearnih parcialnih diferencialnih enačb. Enačbe bodo veljale na neki množici Ω, za katero bomo predpostavili da je odprta, 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 funkcijske prostore, ti. prostore Soboljeva. Na temo prostorov Soboljeva je na voljo ogromno literature. Izreki in definicije iz tega razdelka so bili povzeti po [1].

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

$$W^{k,p}(\Omega) = \{u\colon \mathbb{R}^d \to \mathbb{R}; D^\alpha u \in L^p \text{ za vsak } |\alpha| \le k\}.$$

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

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

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

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

Prostore Soboljeva opremimo z normo

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

Obstajajo tudi druge pogoste definicije norme, 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 $H^k(\Omega) = W^{k,2}(\Omega)$ in 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.*

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

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

Kornova neenakost je močen rezultat, ki povezuje normo prostorov Soboljeva z energijskimi normami, kot 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 [2, str. 229] splošnejši dokaz pa v [3].

Že Soboljev sam je raziskoval vložitve prostorov Soboljeva v Lebesgueove L^p prostore, kasneje pa še mnogi drugi. V [1] je problemu vložitve posvečeno celo četrto poglavje, kjer je tudi dokaz spodnjega izreka [1, 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

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

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 [1, 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 $W^{m,p}(\Omega) \to L^q(\partial\Omega)$ zvezen.*

Opomba 1.17. Pogosto se tudi tej preslikave 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

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

Pri dokazu obstoja in enoličnosti rešitev linearnih parcialnih diferencialnih enačb, si bomo pomagali tudi s klasičnim izrekom funkcionalne analize, dokazanim v vsakem učbeniku na to temo, npr. [4, str. 188, izrek 3.8-1].

Izrek 1.18 (Riezsov 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 $\Phi\colon H \to H^*$.*

Bolj običajno se izrek pove na drugačen način.

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

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

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 kontinuum in s tem zaobidemo težave, ki bi nastale z obravnavo njihove diskretne atomske strukture, saj jih obravnavamo kot zvezno maso, ki zavzema celoten prostor, ki ga zaseda in ne kot diskreten sistem delcev. Prav tako bomo predpostavili zveznost in gladkost količin, povezanih s trdnino. To nam da možnost, da kontinuum delimo na poljubno majhne dele in s tem dobimo osnovo za uporabo teorije infinitezimalnega računa. S pomočjo tega in privzetih fizikalnih zakonov bomo izpeljali diferencialne enačbe, ki bodo opisovale deformacije in napetosti v materialu. Izkaže se, da ta teorija zelo dobro opisuje materiale, ki se uporabljajo v gradbeništvu in strojništvu kot so kovine, beton in steklo in je zaradi tega zelo pogosto uporabljena, paketi za numerično analizo strukture zgradb pa so doživeli velik razvoj in komercialni uspeh.

Začnimo s tem, da natančno definiramo pojme kot so telo in gibanje, s pomočjo katerih bomo izpeljali enačbe, ki opisujejo pojave v našem interesu. Osnovna predstavitev teorije bo povzeta po [5].

2.1 Osnove gibanja

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

$$\boldsymbol{\chi} = \{\chi\colon \mathcal{B} \to \chi(\mathcal{B}) \subseteq \mathbb{R}^3\},$$

da je za vsaki $\chi_1, \chi_2 \in \boldsymbol{\chi}$ preslikava $\chi_2 \circ \chi_1^{-1}\colon \chi_1(\mathcal{B}) \to \chi_2(\mathcal{B})$ 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

$$\{\chi_t \colon \mathcal{B} \to \chi_t(\mathcal{B}) \subseteq \mathbb{R}^3, t \in \mathbb{R}\}.$$

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. $x \in \chi_t(\mathcal{B})$.

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

$$\begin{aligned} \tilde{x} \colon \mathbb{R} \times B \subset \mathbb{R} \times \mathbb{R}^3 &\to \mathbb{R}^3 \\ (t, X) &\mapsto \chi_t(\chi_R^{-1}(X)) \end{aligned}$$

gladka kot funkcija iz $\mathbb{R}^4$ v $\mathbb{R}^3$. Preslikava $\tilde{x}$ nam podaja zvezo med prostorskimi in referenčnimi koordinatami

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

Pogosto opustimo strog zapis s preslikavami in pišemo kar $x = x(t, X)$. Ker je za vsak t preslikava $X \mapsto x(t, X)$ difeomorfizem, lahko funkcijo obrnemo in izrazimo tudi referenčne koordinate kot funkcijo prostorskih $X = X(t, x)$.

Pogosto za referenčno konfiguracijo vzamemo kar konfiguracijo na začetku gibanja, $\chi_R = \chi_{t=t_0}$, ni pa nujno temu tako.

Gibanje, kot opisano sedaj, res modelira makroskopsko gibanje, kot ga poznamo iz resničnega sveta. 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 vektorje na daljše kot $X = (X, Y, Z)$ in $x = (x, y, z)$. Naj bo dano gibanje

$$x(t, X, Y, Z) = (X + tX^2, Y + tXY, Z),$$

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 x difeomorfizem, saj velja

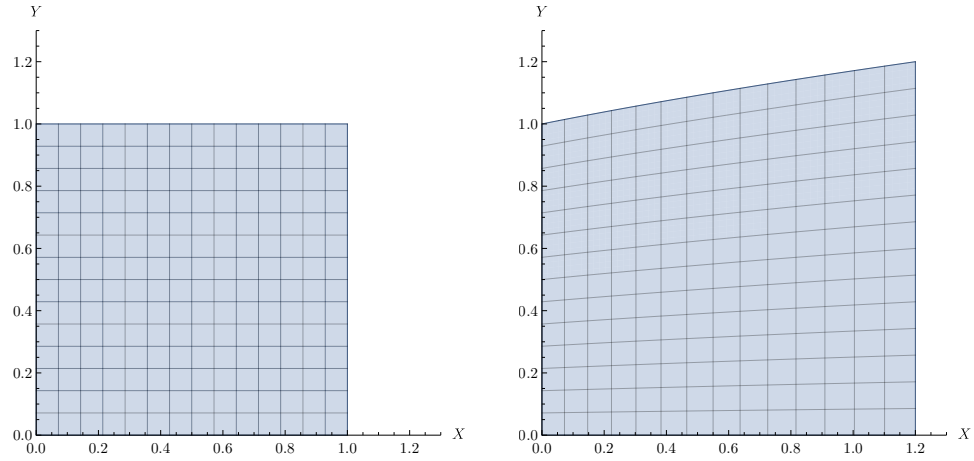
$$\det\left(\frac{\partial x}{\partial(X, Y, Z)}\right) = (1 + tX)(1 + 2tX) > 0.$$

Zato lahko izrazimo obratno preslikavo

$$X(t, x, y, z) = \left(\frac{\sqrt{4tx + 1} - 1}{2t}, \frac{y\left(\sqrt{4tx + 1} - 1\right)}{2tx}, z\right). \tag{2.1}$$

Vsako količino φ definirano na telesu $\mathcal{B}$, kot na primer temperaturo ali hitrost, lahko zapišemo na dva načina, v prostorskih ali v referenčnih koordinatah. Za funkcijo

$$\begin{aligned} \tilde{\varphi} \colon B &\to \varphi(\mathcal{B}) \\ X &\mapsto \varphi(\chi_R^{-1}(X)), \end{aligned} \tag{2.2}$$



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

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

$$\begin{aligned} \hat{\varphi}\colon B_t &\to \varphi(\mathcal{B}) \\ x &\mapsto \varphi(\chi_t^{-1}(x)), \end{aligned} \tag{2.3}$$

pravimo, da je zapisana v prostorskih koordinatah in pišemo $\hat{\varphi} = \varphi(x)$. Med obema zapisoma lahko prehajamo s pomočjo izražave $\varphi(x) = \varphi(x(t, X))$ ali $\varphi(X) = \varphi(X(x, t))$.

Podoben zapis bomo uporabljali tudi pri diferencialnih operatorjih. Pri odvodih moramo povedati, ali na količino gledamo v prostorskih ali v referenčnih koordinatah. Z velikimi črkami bomo pisali operatorje, kjer odvajamo glede na referenčne koordinate, z malimi pa tiste, kjer odvajamo glede na prostorske.

Primer 2.4. Damo imamo količino $\vartheta(t, X, Y, Z) = tX^2 + 2ZY$, ki je zapisana v referenčnih koordinatah. Če jo zapišemo v prostorskih s pomočjo izpeljane inverzne relacije (2.1) iz primera 2.3, dobimo

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

Da demonstriramo še uporabo diferencialnih operatorjev, izračunajmo

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

Te odvode bi zopet lahko zapisali kot funkcije referenčnih ali prostorskih koordinat. Podobno se obnašajo časovni odvodi.

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

Če vidimo zapis $\frac{D\vartheta}{Dt}(t,x,y,z)$, torej časovni odvod ϑ, pri konstantnih referenčnih koordinatah, zapisanega v prostorskih koordinatah, ga izračunamo tako, da izraz prevedemo v referenčne koordinate, odvajamo po času in prenesemo nazaj na prostorske koordinate. S simboli lahko to zapišemo kot

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

Definirajmo še osnovne pojme pomika, hitrosti in pospeška.

Definicija 2.5. Količino

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

imenujemo *pomik*. Količino

$$\vec{v}(X) = \frac{Dx}{Dt}(t,X)$$

imenujemo *hitrost*. Količino

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

imenujemo *pospešek*.

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

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

Pri tem je φ skalarna, vektorska ali tenzorska količina.

Dokaz. Po definiciji najprej prenesemo φ v referenčni koordinatni sistem z uporabo 2.2. Nato odvajamo po verižnem pravilu

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

□

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 predstavljata maso in volumen. Predpostavimo še, da je $m \ll V$, 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

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

za vsako merljivo podmnožico $A \subseteq \mathcal{B}$. Funkciji $\rho\colon \mathcal{B} \to [0,\infty)$ 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 $\mathcal{B}_0 \subseteq B$ 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

$$\frac{D}{Dt} m(B'_t) = 0. \tag{2.4}$$

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

Druga dva aksioma bosta poleg mase imela opraviti s silami. Kako 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 s telesa z nekim zunanjim objektov in se prenašajo preko površine. Te sile v skladu s hipotezo kontinuuma 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 bi bila sila konstantna, potem gostota sile predstavlja silo na enoto volumna ali površine. Slednjemu se (sploh v mehaniki tekočin) reče tudi *pritisk* ali *tlak*.

Definicija 2.10. *Volumenska gostota sile* $\vec{f}$ je zvezna funkcija $\vec{f}\colon B_t \to \mathbb{R}^3$. *Površinska gostota sile* $\vec{t}$ je zvezna funkcija $\vec{t}\colon \partial B_t \to \mathbb{R}^3$.

Primer 2.11. Gostota gravitacijske sile je enaka $\vec{f} = \rho\vec{g}$, kjer je $\vec{g}$ gravitacijski pospešek.

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

$$\frac{D}{Dt}\int_{B'_t} \vec{v}dm = \int_{B'_t} \vec{f}dV + \int_{\partial B'_t} \vec{t}dS. \tag{2.5}$$

Sprememba 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

$$\frac{D}{Dt}\int_{B'_t} x \times \vec{v}dm = \int_{B'_t} x \times \vec{f}dV + \int_{\partial B'_t} x \times \vec{t}dS. \tag{2.6}$$

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

Opomba 2.12. Pri aksiomu 3 smo predpostavili, da na kontinuum ne delujejo notranji, ampak da so vsi navori posledica delovanja zunanjih sil. Drugače povedano, predpostavili smo *nepolarnost* kontinuuma.

V vseh treh aksiomih nastopa odvod po času v referenčnih koordinatah nekega integrala, zapisanega v prostorskih koordinatah. Tu ne velja običajen izrek o odvajanju pod integralom, zato moramo izraz pod integralom najprej prenesti v referenčne koordinate, zamenjati odvod in integral, nato pa ga prenesti nazaj. Naredimo to najprej za aksiom o ohranitvi mase.

Trditev 2.13. *Masa se ohranja* ($\frac{D}{Dt}m(B_t) = 0$) *natanko tedaj, ko je*

$$\frac{D\rho}{Dt} + \rho \operatorname{div} \vec{v} = 0. \tag{2.7}$$

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

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

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

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

Ker je integral nič za vsako telo, mora biti nujno integrand nič, in obratno, če je integrand nič, se masa ohranja. □

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

Trditev 2.14. *Naj bo φ neka količina in privzemimo aksiom o ohranitvi mase. Potem velja*

$$\frac{D}{Dt}\int_{B_t}\varphi dm = \int_{B_t}\frac{D\varphi}{Dt}dm. \tag{2.8}$$

Dokaz. Postopamo enako kot v prejšnji trditvi. Izračunamo

$$\begin{aligned}\frac{D}{Dt}\int_{B_t}\varphi dm = \frac{D}{Dt}\int_B \varphi\rho J dV &= \int_B \frac{D(\varphi\rho J)}{Dt}dV = \\ &= \int_B \left(\frac{D\varphi}{Dt}\rho J + \varphi\left(\underbrace{\frac{D\rho}{Dt} + \rho\operatorname{div}\vec{v}}_{=0 \text{ po } (2.7)}\right)\right)dV = \\ &= \int_B \frac{D\varphi}{Dt}\rho J dV = \int_{B_t}\frac{D\varphi}{Dt}dm.\end{aligned}$$

□

2.2 Napetostni tenzor

Predstavljajmo 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. Iz telesa izrežimo neko podtelo. Na površini tega podtelesa zunanji del kontinuuma 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 drugi del.

Ta razmislek nam ponuja naslednji opis notranjih in površinskih sil telesa. V nekem trenutku t v času se postavimo v neko točko 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 sila v splošnem drugačna. Toda, če sta imeli dve podtelesi v tisti točki enako normalo $\vec{n}$ 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 se reče *Cauchyjeva hipoteza.*

Aksiom 4 (Cauchyjeva hipoteza)**.** V telesu obstaja vektorsko polje $\vec{t}$ inducirano s površinskimi silami, ki je odvisno samo od položaja, časa in normale na površino (resnično ali namišljeno), kjer ga opazujemo. S simboli lahko zapišemo

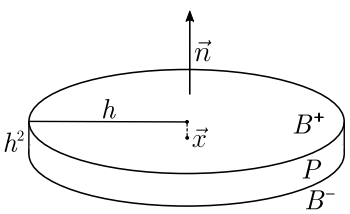
$$\vec{t} = \vec{t}(t, x, \vec{n}).$$

Iz zgornjega razmisleka se zdi, da bi morala biti sila, s katero prvi del kontinuuma deluje na drugega, nasprotno enaka sili, s katero drugi deluje na prvega. Temu se reče tudi tretji Newtonov zakon za mehaniko kontinuuma in ga ni treba privzeti kot aksiom, ampak lahko dokažemo iz do sedaj privzetih aksiomov. Dokaza tega in naslednjega izreka bosta povzeta po dokazih iz [6, str. 104–107].

Trditev 2.15 (Cauchyjeva recipročna relacija)**.**

$$\vec{t}(t, x, -\vec{n}) = -\vec{t}(t, x, \vec{n}). \tag{2.9}$$

Dokaz. Izberimo trenutek v času t in točko x. Krajše pišimo $\vec{t}_{\vec{n}} = \vec{t}(t, x, \vec{n})$. Oglejmo si podtelo v obliki krožnega valja z višino h^2 in radijem h, ki ima v središču točko x in je $\vec{n}$ normala na eno izmed osnovnic. 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 vsake osnovnice je πh^2, površina plašča pa $2\pi h^3$. Volumen valja je πh^4. Primer valja je prikazan na sliki 2.



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

Uporabimo za ta valj aksiom o gibalni količini (2.5):

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

Po trditvi 2.14 lahko zamenjamo odvod in integral, rob vclja razpišimo po ploskvch:

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

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

$$V(B)(\rho\frac{D\vec{v}}{Dt})(x_1)\cdot\vec{c} = V(B)\vec{f}(x_2)\cdot\vec{c} + S(B^+)\vec{t}_{\vec{n}}(x_3)\cdot\vec{c} + \\ + S(B^-)\vec{t}_{-\vec{n}}(x_4)\cdot\vec{c} + S(P)\vec{t}_{\vec{m}(x)}(x_5)\cdot\vec{c},$$

kjer so x_1, x_2, x_3, x_4, x_5 neke vmesne točke v ali na površini vclja. Če vstavimo notri izraze za volumen in površine ploskev dobimo

$$\pi h^4(\rho\frac{D\vec{v}}{Dt})(x_1)\cdot\vec{c} = \left[\pi h^4\vec{f}(x_2) + \pi h^2\vec{t}_{\vec{n}}(x_3) + \pi h^2\vec{t}_{-\vec{n}}(x_4) + 2\pi h^3\vec{t}_{\vec{m}(x)}(x_5)\right]\cdot\vec{c}.$$

Celoten izraz delimo z πh^2 in pogledamo limito $h\to 0$, gredo vse točke x_i proti x in dobimo

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

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

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

kar je točno to, kar smo želeli pokazati. □

Izkaže se, da velja še več: vektor $\vec{t}$ je linearno odvisen od normale. S pomočjo prejšnje trditve lahko pokažemo naslednji izrek, ki nam da na voljo matematični objekt, s pomočjo katerega lahko opišemo napetost v materialu.

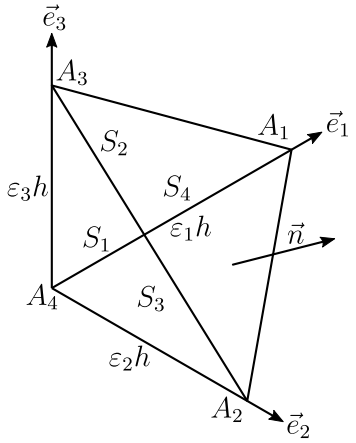
Izrek 2.16 (Cauchyjev izrek o napetosti)**.** *Obstaja tenzor σ, tako da velja*

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

Tenzor σ se imenuje Cauchyjev napetostni tenzor.

Dokaz. Podobno kot pri Cauchyjevi recipročni relaciji si tokrat izberimo majhno telo z ogliščem v točki x. Naj bo to tetraeder z 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 $A_1, A_2, A_3, A_4 = x$ in ploskev nasproti izbranega oglišča z 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 $\vec{n}$. Celotna situacija je prikazana na sliki 3. Sedaj



Slika 3: Tetraeder, uporabljen za dokaz Cauchyjevega izreka o napetosti.

lahko izračunamo 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 standardno 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.

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

Če to pomnožimo z i-tim baznim vektorjem, dobimo $a_i = (\vec{n}\cdot\vec{e}_i)a_4$. Zopet 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

$$\begin{aligned} \frac{D}{Dt}\int_{B_t}\vec{v}dm &= \int_{B_t}\vec{f}dV + \int_{\partial B_t}\vec{t}dS \\ \int_{B_t}\rho\frac{D\vec{v}}{Dt}dV &= \int_{B_t}\vec{f}dV + \int_{S_1}\vec{t}_{-\vec{e}_1}dS + \int_{S_2}\vec{t}_{-\vec{e}_2}dS + \int_{S_3}\vec{t}_{-\vec{e}_3}dS + \int_{S_4}\vec{t}_{\vec{n}}dS \\ V(\rho\frac{D\vec{v}}{Dt})(x_1)\cdot\vec{c} &= \left[V\vec{f}(x_2) + a_1\vec{t}_{-\vec{e}_1}(x_3) + a_2\vec{t}_{-\vec{e}_2}(x_4) + a_3\vec{t}_{-\vec{e}_3}(x_5) + a_4\vec{t}_{\vec{n}}(x_6)\right]\cdot\vec{c}, \end{aligned}$$

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

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

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 $\vec{c}$ in ko uporabimo še Cauchyjevo recipročno relacijo (2.9), dobimo

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

Ta zveza nam 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

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

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 2.17 (Cauchyjeva momentna enačba)**.** *Za gibanje kontinuuma velja Cauchyjeva momentna enačba*

$$\rho \frac{D\vec{v}}{Dt} = \vec{f} + \operatorname{div} \sigma. \tag{2.10}$$

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 2.14 o menjavi odvoda in integrala. Ker mora biti zadnji integral nič za vsako telo, mora biti integrand nič, kar dokaže našo enačbo. $\square$

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 2.18. *Cauchyjev napetostni tenzor je simetričen:*

$$\sigma^{\mathsf{T}} = \sigma.$$

Dokaz. Začnimo z zakonom o vrtilni količini (2.6) in ga prevedimo v lokalno obliko.

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

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

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

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

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

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

Uporabili smo definicijo in cikličnost mešanega produkta, Cauchyjevo momentno enačbo (2.10) in dejstvo, da je

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

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

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

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

$$\sigma : w = 0$$

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

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 kontinuuma kot elastične trdnine preko posplošitve Hookovega zakona, ki povezuje napetosti in 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 2.19 (Gradient deformacije)**.** Količino $F = \frac{\partial x}{\partial X}(t, X) = \operatorname{Grad} x$ imenujemo *gradient deformacije*.

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

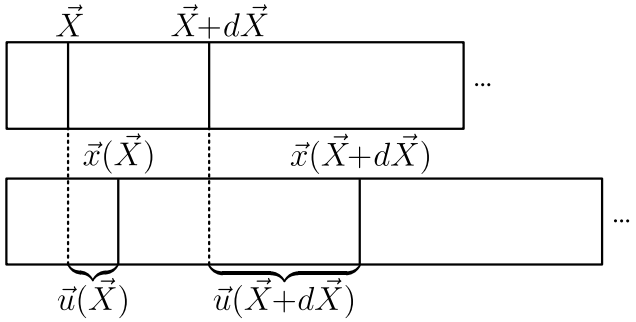
$$dx := x(t, X + dX) - x(t, X) = FdX + O(dX^2).$$

Tenzor F do prvega reda natančno opiše kako se vektor dX iz referenčne konfiguracije deformira v vektor dx v prostorski konfiguraciji.

Vendar, ni vsa deformacija, ki jo opiše tenzor F taka, da bi povzročala napetosti. Hookov zakon v eni dimenziji pravi, da je sila sorazmerna raztezku. Če imamo opravka s togim premikom $X \mapsto QX+a$, ta premik ne bo povzročil nobene napetosti, saj se bo telo samo premaknilo, ne pa raztegnilo.

Oglejmo si najprej primer v eni dimenziji.

Primer 2.20. Naj bo dana tanka palica kot na sliki 4, ki jo raztegnemo vzdolž njene nosilke. Premik v točki X je $u(X) = x - X$, v točki $X + dX$ pa $u(X + dX)$.



Slika 4: Razteg tanke palice vzdolž njene osi.

Dejanski relativni raztezek delčka palice dolžine dX pa je

$$\frac{dx - dX}{dX} = \frac{x(X + dX) - x(X) - (X + dX - X)}{dX} = \\ = \frac{\vec{u}(X + dX) - \vec{u}(X)}{dX} = \vec{u}'(X) + O(dX).$$

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

Preverimo še, da ta mera raztezka zadošča nekaterim intuitivnim zahtevam. Za togi premik $X \mapsto X + c$ je $\vec{u}(X) = c$ in $\vec{u}'(X) \equiv 0$, kot pričakovano.

Za enakomerni razteg $X \mapsto aX$ je $\vec{u}(X) = aX - X$ in $\vec{u}'(X) = a - 1$. Tudi to ustreza pričakovanjem, saj za $a = 1$ ni raztezka, za $a < 1$ je to skrčitev in je raztezek negativen, za $a > 1$ pa dobimo pozitivno število.

Posplošitev mere deformacije med točkama X in $X + dX$ v višje dimenzije bi bila

$$\varepsilon_1 = \frac{\|dx\| - \|dX\|}{\|dX\|}.$$

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

$$\varepsilon_2 = \frac{\|dx\|^2 - \|dX\|^2}{\|dX\|^2}.$$

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.

$$\begin{aligned}\varepsilon_2 &= \frac{\|dx\|^2 - \|dX\|^2}{\|dX\|^2} = \frac{\|dx\|^2}{\|dX\|^2} - 1 = \frac{\|FdX + O(dX^2)\|^2}{\|dX\|^2} - 1 = \\ &= \frac{dX^\mathsf{T} F^\mathsf{T} F dX + O(dX^3)}{\|dX\|^2} - 1 = \frac{dX}{\|dX\|}(\underbrace{F^\mathsf{T}F - I}_{2E})\frac{dX}{\|dX\|} + O(dX)\end{aligned}$$

V limiti $dX \to 0$ lahko mero ε_2 predstavimo s tenzorjem $F^\mathsf{T}F - I$, mero ε_1 pa s tenzorjem E.

Definicija 2.21 (deformacijski tenzor)**.** Količina $E = \frac{1}{2}(F^\mathsf{T}F - I)$ se imenuje (Cauchy-Greenov) deformacijski tenzor.

Trditev 2.22. *Deformacijski tenzor E lahko izrazimo z gradientom pomika kot*

$$E = \frac{1}{2}\left(\operatorname{Grad}\vec{u} + \operatorname{Grad}\vec{u}^\mathsf{T} + \operatorname{Grad}\vec{u}^\mathsf{T}\operatorname{Grad}\vec{u}\right).$$

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

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

Izračunamo:

$$\begin{aligned}&\frac{1}{2}\left(\operatorname{Grad}\vec{u} + \operatorname{Grad}\vec{u}^\mathsf{T} + \operatorname{Grad}\vec{u}^\mathsf{T}\operatorname{Grad}\vec{u}\right) = \\ &\quad= \frac{1}{2}\left(F - I + F^\mathsf{T} - I + (F - I)^\mathsf{T}(F - I)\right) = \\ &\quad= \frac{1}{2}\left(F + F^T - 2I + F^\mathsf{T}F - F - F^T + I\right) = \\ &\quad= \frac{1}{2}\left(F^\mathsf{T}F - I\right) = \\ &\quad= E.\end{aligned}$$

□

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 koordinatne 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 to, koliko se U razlikuje od identitete, toda polarni razcep matrike je težko izračunati. Poglejmo si kakšno zvezo ima zgoraj definirani deformacijski tenzor E z U:

$$E = \frac{1}{2}(f^\mathsf{T} F - I) = \frac{1}{2}(U^\mathsf{T} R^\mathsf{T} R U - I) = \frac{1}{2}(U^2 - I).$$

Vidimo, da E meri, kako se U^2 razlikuje od identitete, ki predstavlja gibanje brez deformacij.

Kot pri enodimenzionalnem primeru si oglejmo, kako se E obnaša pri enostavnih deformacijah. Za toge premike imamo želeno obnašanje, kot pokazano v naslednji trditvi.

Primer 2.23. Kot prej si oglejmo, da je za toge deformacije $E = 0$. Naj bo deformacija toga, torej oblike $X \mapsto QX + c$ z ortogonalno konstantno matriko Q in konstantnim c. Tedaj je

$$E = \frac{1}{2}(F^\mathsf{T} F - I) = \frac{1}{2}(Q^\mathsf{T} Q - I) = \frac{1}{2}(I - I) = 0.$$

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

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

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

Opomba 2.24. Velja tudi obrat trditve in posledično ekvivalenca, da je $E = 0$ natanko tedaj, kot je transformacija toga. Ta ekvivalenca je dokazana za primer infinitezimalnega deformacijskega tenzorja v trditvi 2.26.

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

Definicija 2.25 (infinitezimalni deformacijski tenzor)**.** Količino

$$\varepsilon = \frac{1}{2}(\operatorname{Grad} \vec{u} + \operatorname{Grad} \vec{u}^\mathsf{T}) \tag{2.11}$$

imenujemo *infinitezimalni deformacijski tenzor*, ki je geometrijska linearizacija deformacijskega tenzorja.

Pokažimo še naslednjo karaterizacijo, ki pove, da se tudi ε za toge pomike obnaša kot pričakovano. Dokaz je povzet po [7, str. 56].

Trditev 2.26. *Infinitezimalni deformacijski tenzor ε je ničeln natanko tedaj, ko je $\vec{u} = \vec{a} + \vec{b} \times X$, za konstantna vektorja $\vec{a}$ in $\vec{b}$.*

Dokaz. Iz $\varepsilon = 0$ direktno sledi $\operatorname{Grad} u = -\operatorname{Grad} u^{\mathsf{T}}$, torej je $\operatorname{Grad} u$ antisimetričen. Naj bo Ω odprta konveksna množica pod $\mathcal{B}$. Izbrerimo poljubni točki $X, Y \in \Omega$ in daljico med njima enakomerno parametrizirajmo, tako da je $\gamma(1) = X, \gamma(0) = Y, \dot{\gamma}(t) = X - Y$. Integrirajmo $\operatorname{Grad} \vec{u}$ po γ:

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

Če zgornjo enakost pomnožimo skalarno z $X - Y$, dobimo zaradi antisimetričnosti $\operatorname{Grad} \vec{u}$

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

Videli smo torej, da za vsaka X in Y velja $(X - Y) \cdot (\vec{u}(X) - \vec{u}(Y)) = 0$. Če to odvajamo po X, dobimo

$$\vec{u}(X) - \vec{u}(Y) + \operatorname{Grad} \vec{u}(X)^{\mathsf{T}}(X - Y) = 0$$

in če odvajamo še po Y, dobimo

$$-\operatorname{Grad} \vec{u}(Y) - \operatorname{Grad} \vec{u}(X)^{\mathsf{T}} = 0.$$

Če upoštevamo še antisimetričnost $\operatorname{Grad} \vec{u}$ dobimo, da je

$$\operatorname{Grad} \vec{u}(Y) = \operatorname{Grad} \vec{u}(X).$$

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

2.4.2 Hookov zakon

Sedaj potrebujemo relacijo, ki povezuje premike z napetostjo. V splošnem je relacija oblike $\sigma = f(\varepsilon)$, predpostavimo torej, da je napetost odvisna samo od deformacije. Funkcijo f dodatno omejimo, da mora biti taka, da je energija

$$U(\varepsilon) = \int_0^{\varepsilon} \sigma : d\varepsilon = \int_0^{\varepsilon} f(\varepsilon)d\varepsilon$$

dobro definirana, torej neodvisna od poti integracije. V tem primeru je

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

Če to drži, pravimo, da je material hiperelastičen, kot pravi naslednja definicija.

Definicija 2.27. Material je *hiperelastičen*, če je $U(\varepsilon) = \int_0^\varepsilon \sigma : d\varepsilon$ neodvisen od poti integracije.

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:

$$\sigma = C : \varepsilon.$$

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

Opomba 2.28. Aksiom 5 se po komponentah glasi $\sigma_{ij} = C_{ijkl}\varepsilon_{kl}$.

Na srečo v splošnem C nima 81 prostih komponent. Iz trditve 2.18 o simetričnosti σ sledi, da lahko prosto zamenjamo indeksa i in j in 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. Če dodatno predpostavimo še hiperelastičnost, vidimo, da je $\frac{\partial U}{\partial \varepsilon_{ij}} = \sigma_{ij} = C_{ijkl}\varepsilon_{kl}$ in

$$\frac{\partial U}{\partial \varepsilon_{ij}\varepsilon_{kl}} = C_{ijkl}.$$

Ker vrstni red drugih odvodov ni pomemben, je $C_{ijkl} = C_{klij}$. S tem smo C reducirali na 21 komponent. Od sedaj naprej bomo predpostavili, da so vsi materiali hiperelastični. Energija je v tem primeru dana s kvadratično formo

$$U = \frac{1}{2}\varepsilon : C : \varepsilon. \tag{2.12}$$

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

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

kjer so λ, μ in κ splošni skalarji. V [8] so karakterizirani vsi izotropični tenzorji do reda 8, število izotropičnih tenzorjev dimenzije n pa je navedeno kot A005043 v spletni enciklopediji celoštevilskih zaporedij [9].

Zaradi simetrij C mora za izotropičen C veljati $\kappa = \lambda$. Nekoordinatno lahko sedaj zvezo $\sigma = C : \varepsilon$ za tak tenzor zapišemo kot

$$\sigma = \lambda(\operatorname{tr}\varepsilon)I + 2\mu\varepsilon. \tag{2.13}$$

Parametra λ in μ imenujemo Laméjevi konstanti. Običajno so materiali tudi homogeni, torej snovne konstante niso odvisne od lokacije v materialu, temveč so lastnosti materiala samega. Veliko resničnih gradbenih materialov kot na primer železo, jeklo, aluminij in ostale kovine ter steklo zadoščajo vseh predpostavkam Navierove enačbe. Primer anizotropnega materiala je recimo les. V strojniških priročnikih

najdemo snovne lastnosti bolje pogosto opisane s parametroma E in ν, ki jima pravimo Youngov (prožnostni) modul in Poissonovo razmerje. Zveza med njimi je

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

Pogosto najdemo tudi druge snovne parametre kot na primer strižni modul ali stisljivost. Tabela pretvorb med različnimi parametri je na voljo v [5, tabela 5.1, str. 215]. Teoretična omejitev za parametre je dana s tem, da zahtevamo pozitivno definitnost energije kot kvadratne forme in velja, če je $\lambda, \mu > 0$ ali pa $E > 0$ in $-1 < \nu < \frac{1}{2}$. Poissonovo razmerje predstavlja razmerje med tem, koliko se telo ob 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 [10], raziskave na tem področju pa so aktivne še danes.

V praksi se izkaže, da temu brez težav zadostimo. Vrednosti parametrov za nekaj pogostih materialov so podane v tabeli 1 in jih lahko najdemo v primernem fizikalnem ali strojniškem priročniku, npr. v [vstavi referenco / vprašaj Mejaka] TODO.

material	E [GPa]	ν
jeklo	210	0.30
aluminij	69	0.33
steklo	50 – 90	0.20 – 0.27

Tabela 1: Vrednosti elastičnih parametrov za pogoste materiale.

2.5 Navierova enačba

Sedaj imamo vse pripravljeno za opis teorije linearne elastičnosti:

1) mera deformacije: $\varepsilon = \frac{1}{2}(\operatorname{Grad}\vec{u} + \operatorname{Grad}\vec{u}^{\mathsf{T}})$,

2) konstitutivna zveza: $\sigma = C : \varepsilon$ in

3) gibalna enačba: $\rho\frac{D\vec{v}}{Dt} = \vec{f} + \operatorname{div}\sigma$.

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

$$\operatorname{Grad}\varphi = \frac{\partial\varphi}{\partial X} = \frac{\partial\varphi}{\partial x}\frac{\partial x}{\partial X} = \operatorname{grad}\varphi(I + \operatorname{Grad}u) \approx \operatorname{grad}\varphi.$$

Zaradi majhnih gradientov pomika je vseeno, ali uporabljamo odvode glede na prostorske ali glede na referenčne koordinate. Enak argument lahko uporabimo tudi za divergenco in običajne odvode.

Cauchyjevo gibalno enačbo lahko zapišemo namesto v prostorskih tudi v referenčnih koordinatah. 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 referenčnih in prostorskih koordinat, saj bomo imeli od sedaj naprej enačbe vedno zapisane v referenčnih koordinatah. Odvode po času pišimo zato kar s piko, za operatorje pa uporabljajmo običajne male črke ali pa kar ∇.

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

$$\rho\ddot{\vec{u}}(t,X) = \vec{f}(t,X) + \operatorname{div}\sigma(t,X). \tag{2.14}$$

Sedaj imamo vse pripravljeno, da napišemo končno enačbo, ki ji ustreza pomik materiala. Z njeno pomočjo lahko, glede na dane sile na površini in pomike na robu, izračunamo pomike v celotnem telesu, preko deformacijskega tenzorja in Hookovega zakona pa tudi napetosti.

Izrek 2.29 (Navierova enačba)**.** *Če so gradienti pomikov v linearno elastičnem hiperelastičnem izotropičnem homogenem nepolarnem mediju majhni, potem za njih velja* Navierova enačba

$$\rho\ddot{\vec{u}} = \vec{f} + (\lambda+\mu)\operatorname{grad}\operatorname{div}\vec{u} + \mu\triangle\vec{u}. \tag{2.15}$$

Dokaz. Od linearne enačbe gibanja (2.14) do Navierove enačbe nas loči le še izračun $\operatorname{div}\sigma$. Za primerjavo naredimo izračun koordinatno in nekoordinatno. Začnimo s koordinatnim:

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

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

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

Izračun prinese enak rezultat kot prej in izrek je še enkrat dokazan. □

Opomba 2.30. Če je kontinuum v mirovanju, torej $\dot{\vec{u}} = 0$, potem se Navierova enačba glasi

$$(\lambda+\mu)\operatorname{grad}\operatorname{div}\vec{u} + \mu\triangle\vec{u} + \vec{f} = 0 \tag{2.16}$$

in jo imenujemo *stacionarna Navierova enačba.*

Opomba 2.31. 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

$$\rho\ddot{\vec{u}} = \frac{1}{2}\operatorname{div}(C : (\operatorname{grad}\vec{u} + \operatorname{grad}\vec{u}^{\mathsf{T}})) + \vec{f}. \tag{2.17}$$

2.6 Obstoj in enoličnost rešitve

Navierova enačba je linearna parcialna diferencialna enačba drugega reda. Z obstojem in enoličnostjo rešitve se bomo ukvarjali pri stacionarni Navierovi enačbi (2.16) z mešanimi robnimi pogoji pri katerih je na nekem delu roba predpisan pomik, na drugem delu pa površinska sila.

Ker je enačba linearna, je ideja dokaza obstoja in enoličnosti, da rešitev enačbe zapišemo kot linearen funkcional na primernem prostoru in nato uporabimo Riezsov reprezentacijski izrek 1.19, ki nam bo dal obstoj rešitve. Ideja poti do dokaza je vzeta iz [2, izrek 3.17.1, str. 232]. Iz izkušenj vemo, da bo rešitev obstajala in bo enolična v šibkem smislu, zato prevedimo enačbo v šibko obliko in definirajmo šibko rešitev. Začeli bomo bolj splošno in prevedli na šibko obliko stacionarno linearizirano enačbo gibanja (2.14), nato pa nadaljevali do stacionarne Navierove enačbe. Šibko obliko enačbe dobimo tako, da jo pomnožimo z splošno funkcijo $\vec{v}$, integriramo in preko integracije *per partes* ali drugih izrekov znižamo red odvoda na funkciji, ki jo iščemo.

Trditev 2.32. *Šibka oblika linearne stacionarne enačbe gibanja* $\operatorname{div}\sigma + \vec{f} = 0$ *je*

$$\int_\Omega \sigma : \varepsilon(\vec{v})dV - \int_{\partial\Omega} \vec{t}\cdot\vec{v}dS - \int_\Omega \vec{f}\cdot\vec{v}dV = 0. \tag{2.18}$$

Dokaz. Pomnožimo enačbo $\operatorname{div}\sigma + \vec{f} = 0$ skalarno z $\vec{v} \in [L^2(\Omega)]^3$ in integrirajmo po Ω ter poskušajmo prenesti 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, $\varepsilon(\vec{v}) = \frac{1}{2}(\operatorname{grad}\vec{v} + \operatorname{grad}\vec{v}^{\mathsf{T}})$. Nadaljujemo z računom in dobimo

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

□

Pri Navierovi enačbi upoštevamo še relacijo $\sigma = C : \varepsilon$ in s pomočjo tega bomo definirali šibko rešitev Navierove enačbe. Najprej pa si natančno oglejmo problem in prostor, v katerem bomo rešitve iskali. Problem zastavimo za splošno stacionarno obliko Navierove enačbe (2.17), pri čemer predpostavimo le pozitivno definitnost C. Najti želimo $\vec{u}$, ki zadošča

$$\begin{aligned}
\operatorname{div}(C : \varepsilon(\vec{u})) + \vec{f} = 0 \qquad & \text{na } \Omega \\
\vec{u} = 0 \qquad & \text{na } S \subseteq \partial\Omega \\
\sigma\vec{n} = (C : \varepsilon(\vec{u}))\vec{n} = \vec{t} \qquad & \text{na } \partial\Omega \setminus S,
\end{aligned} \tag{2.19}$$

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 $C^2(\Omega, R^3)$, za katere velja $\vec{u}|_S = 0$. Označimo ta prostor z C^2_S in na njem definirajmo bilinearno formo

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

Trditev 2.33. *Bilinearna forma* $\langle \vec{u}, \vec{v} \rangle_E = \int_\Omega \varepsilon(\vec{u}) : C : \varepsilon(\vec{v}) dV$ *na* C^2_S *določa skalarni produkt.*

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 2.26 sledi, da je $\vec{u}$ oblike $\vec{u} = \vec{a} + \vec{b} \times x$. Zaradi ničelnosti na kosu roba mora biti $\vec{u} \equiv 0$. □

Prostor C^2_S 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 2.19.

Definicija 2.34. Rešitev $\vec{u} \in E$ je šibka rešitev problema 2.19, če za vsako funkcijo $\vec{v} \in E$ velja

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

Sedaj imamo vse pripravljeno za izrek o obstoju in enoličnosti.

Izrek 2.35. *Naj bo* $\vec{f} \in [L^{\frac{6}{5}}(\Omega)]^3$ *in* $\vec{t} \in [L^{\frac{4}{5}}(\Omega)]^3$. *Potem ima problem 2.19 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 pove, da je $\|u\|^2_{[H^1(\Omega)]^3} \leq c_1 \|u\|^2_E$. Dano imamo torej zvezno vložitev E v $[H^1(\Omega)]^3$. Na vsaki komponenti $\vec{u}$ lahko sedaj uporabimo izrek o vložitvi prostorov Soboljeva 1.14 in izrek o sledi 1.16 in dobimo, da je vsaka komponenta $\vec{u}$ in $|\vec{u}|$ ležijo v L^6. Po istem izreku dobimo, da za vsako odsekoma gladko ploskev $\Sigma \subseteq \Omega$ komponente in norma $\vec{u}$ ležijo v $L^4(\Sigma)$. Dobimo verigi zveznih vložitev

$$E \hookrightarrow [H^1(\Omega)]^3 \hookrightarrow [L^6(\Omega)]^3$$
$$E \hookrightarrow [H^1(\Omega)]^3 \hookrightarrow [L^4(\Sigma)]^3,$$

ki prek norm na prvem in zadnjem prostoru implicirata neenakosti

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

Na šibko obliko Navierove enačbe iz definicije šibke rešitve 2.34 lahko gledamo kot na 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, tako da je Hilbertov. Če pokažemo še, da je F zvezen, potem lahko zanj uporabimo Riezsov reprezentacijski izrek 1.19. Pokažimo raje omejenost. Z uporabo trikotniške, Cauchy-Schwarzove, Hölderjeve in zgoraj izpeljanih neenakosti dobimo

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

in

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

Funkcional F je po trikotniški neenakost torej omejen in zato zvezen. Po izreku 1.19 torej 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 $\vec{v} \in E$, nam ne preostane drugega, kot da je $\vec{u} = \vec{u}^*$ enolična rešitev v smislu definicije 2.34. □

Samo enoličnost lahko dokažemo tudi na bolj elementaren način, ki morda nudi boljši vpogled v to, kako robni pogoji določijo enoličnost rešitve v notranjosti.

Izrek 2.36 (Kirchoff)**.** *Rešitev problema 2.19 je enolična v E, če obstaja.*

Dokaz. Pa recimo 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

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

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

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

in ker je $\vec{w}$ tudi v E, saj je na robu enaka 0, lahko vstavimo $\vec{v} = \vec{w}$. Zaradi pozitivne definitnosti C sledi $\varepsilon(\vec{w}) = 0$ od koder kot prej zaradi robnih pogojev sledi $\vec{w} = 0$ oziroma $\vec{u}_1 = \vec{u}_2$. □

Opomba 2.37. Iz enoličnost v E sledi tudi enoličnost v klasičnem smislu, saj je prostor klasičnih rešitev podprostor v E.

2.7 Priprava na numerično reševanje

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

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

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

2.7.1 Poenostavitev na dve dimenziji

Tridimenzionalne probleme se zavoljo lažje formulacije, manjše računske zahtevnosti in lažje implementacije, pogosto s pomočjo neke simetrije, prevede na nižjedimenzionalne. Spodaj sta opisani dve najbolj pogosti poenostavitvi.

Ravninska napetost: Pri tej poenostavitvi predpostavimo, da napetost nima neničelnih 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 z eno komponento manj.

Ravninska deformacija: Pri tej poenostavitvi predpostavimo, da deformacijski tenzor ε nima komponent v eni izmed koordinatnih smeri, torej, podobno kot pri ravninski napetosti lahko predpostavimo, da 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 spreminjajo, torej za (ne nujno krožne) valje. Ta poenostavitev da enačbo, ki je ekvivalentna tisti, dobljeni z zgornjo poenostavitvijo, le da je potrebno uporabiti druge materialne konstante $\hat{E} = \frac{E}{1-\nu}$, $\hat{\nu} = \frac{\nu}{1-\nu}$, pri čemer sta E in ν parametra iz tridimenzionalnega problema.

Pri obeh poenostavitvah se iz pogojev izpelje, da v Navierovi enačbi ne nastopa tretja komponenta u. V dveh dimenzijah pišemo prvo komponento pomika z u in drugo z v, koordinati pa označujemo z x in y. Če si razpišemo izraze za deformacijski tenzor in napetostni tenzor za primer ravninske napetosti dobimo:

$$\varepsilon = \begin{bmatrix} \frac{\partial u}{\partial x} & \frac{1}{2}(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}) \\ \frac{1}{2}(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}) & \frac{\partial v}{\partial y} \end{bmatrix}$$
$$\sigma = \begin{bmatrix} \lambda\frac{\partial v}{\partial y} + (\lambda + 2\mu)\frac{\partial u}{\partial x} & \mu(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}) \\ \mu(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}) & \lambda\frac{\partial u}{\partial x} + (\lambda + 2\mu)\frac{\partial v}{\partial y} \end{bmatrix}.$$

2.7.2 Robni pogoji

Pri reševanju robnih problemov bomo imeli tri vrste robnih pogojev. Dirichletove robne pogoje $u|_{\partial\Omega} = u_0$ bomo uporabili, ko bomo poznali pomike na robu. Najpogosteje bo pogoj oblike $u = 0$, ko nek konec držimo fiksen. 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 $\vec{n}$ zunanja normala na rob. Za primer desnega roba, na katerega enakomerno deluje neka gostota sile p v normalni smeri, dobimo dva pogoja:

$$\vec{t} = \sigma\vec{n} = \begin{bmatrix} \lambda\frac{\partial v}{\partial y} + (\lambda + 2\mu)\frac{\partial u}{\partial x} & \mu(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}) \\ \mu(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}) & \lambda\frac{\partial u}{\partial x} + (\lambda + 2\mu)\frac{\partial v}{\partial y} \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} p \\ 0 \end{bmatrix}.$$

3 Numerična metoda

V 20. stoletju je skupaj z razvojem računalnikov začel svojo pot 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 močni (*angl.* strong form). Najznamenitejši in zelo uspešen predstavnik prve skupine je metoda končnih elementov (MKE) (*angl.* finite element method (FEM)), 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 močni obliki pa je metoda končnih diferenc (MKD) (*angl.* finite diference method (FDM)), 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 samo mrežo na robu domene. Metoda končnih diferenc je običajno formulirana na pravokotni mreži. Obstajajo pa tudi metode, ki mreže ne potrebujejo, imenujemo jih *brezmrežne metode* (*angl.* meshfree methods). Predstavljena metoda v tem razdelku bo reševala enačbo v močni obliki in bo brezmrežna.

3.1 Izpeljava

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

3.1.1 Ideja in motivacija

Primer 3.1. Rešujemo enodimenzionalno Poissonovo enačbo. Izpeljava metode končnih diferenc 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

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

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 $i = 0, \ldots, N$. Za vsako od teh točk uvedemo neznanko u_i, ki predstavlja neznano funkcijsko vrednost v točki x_i. S pomočjo vrednosti u_{i-1}, u_i in u_{i+1} želimo sedaj aproksimirati $u''(x_i)$. To nam bo dalo zvezo med spremenljivkami in ko jo uporabimo za vse notranje točke ter upoštevamo še robne pogoje, bomo dobili sistem linearnih enačb, katerega rešitev nam bo dala dobro aproksimacijo funkcije u.

Funkcijo u v okolici x_i aproksimiramo z interpolacijskim polinomom, njene odvode pa z odvodi interpolacijskega polinoma. Da najdemo interpolacijski polinom $\hat{u}$, zapišimo

$$\hat{u}(x) = \alpha_0 + \alpha_1 x + \alpha_2 x^2 = \begin{bmatrix} 1 & x & x^2 \end{bmatrix} \begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \alpha_2 \end{bmatrix} = \boldsymbol{b}(x)^\mathsf{T} \boldsymbol{\alpha}$$

in poiščimo koeficiente $\boldsymbol{\alpha}$, da bo veljalo

$$\begin{aligned} \hat{u}(x_{i-1}) &= u_{i-1} \\ \hat{u}(x_i) &= u_i \\ \hat{u}(x_{i+1}) &= u_{i+1}. \end{aligned}$$

Če sistem enačb razpišemo, dobimo

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

oziroma v matrični obliki

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

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

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

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

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

Toda, ker u_j nastopajo linearno, lahko napišemo tudi v obliki

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

S tem smo ločili podatke, ki se nanašajo na vrednost funkcije, od podatkov, ki se nanašajo na pozicije točk. Če na primer vemo, da bomo večkrat potrebovali vrednost interpolacijskega polinoma v neki točki x^* za različne nabore funkcijskih vrednosti (vendar še vedno izmerjene v istih točkah) $\boldsymbol{u}$, potem se nam splača poračunati $\boldsymbol{\varphi}(x^*)$ vnaprej in vrednosti interpolacijskega polinoma dobimo vsakič znova le s skalarnim produktom $\hat{u}(x^*) = \boldsymbol{\varphi}(x^*)^{\mathsf{T}}\boldsymbol{u}$. Tukaj ni potrebno, da je x^* ena izmed točk x_i, ampak je lahko poljubna točka v domeni, čeprav je potrebno da je v bližini x_i, če želimo dobiti dobro aproksimacijo.

Za aproksimacijo u' in u'' bomo vzeli kar odvode $\hat{u}$. Izračunajmo jih v točki x_i in dobimo znane formule

$$\begin{aligned}\hat{u}'(x_i) &= \boldsymbol{\varphi}'(x_i)^{\mathsf{T}}\boldsymbol{u} = \begin{bmatrix}-\frac{1}{2h} & 0 & \frac{1}{2h}\end{bmatrix}\begin{bmatrix}u_{i-1}\\ u_i\\ u_{i+1}\end{bmatrix}\\ \hat{u}''(x_i) &= \boldsymbol{\varphi}''(x_i)^{\mathsf{T}}\boldsymbol{u} = \begin{bmatrix}\frac{1}{h^2} & -\frac{2}{h^2} & \frac{1}{h^2}\end{bmatrix}\begin{bmatrix}u_{i-1}\\ u_i\\ u_{i+1}\end{bmatrix}.\end{aligned}$$

To lahko uporabimo za reševanje našega problema (3.1). Namesto enakosti

$$u''(x_i) = f(x_i)$$

za vsako točko x_i v notranjosti zapišemo podobno enakost

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

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

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

ki bi jo izpeljali na enak način.

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

$$\frac{1}{h^2} \begin{bmatrix} 1 & & & & & & \\ -1 & 2 & 1 & & & & \\ & -1 & 2 & 1 & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & -1 & 2 & 1 \\ & & & h/2 & -2h & 3h/2 \end{bmatrix} \begin{bmatrix} u_0 \\ u_1 \\ u_2 \\ \vdots \\ u_{N-1} \\ u_N \end{bmatrix} = \begin{bmatrix} f(x_0) \\ f(x_1) \\ f(x_2) \\ \vdots \\ f(x_{N-1}) \\ f(x_N) \end{bmatrix}.$$

Rešitev tega sistema nam dobro aproksimira neznano funkcijo u v izbranih točkah x_i.

3.1.2 Splošna izpeljava

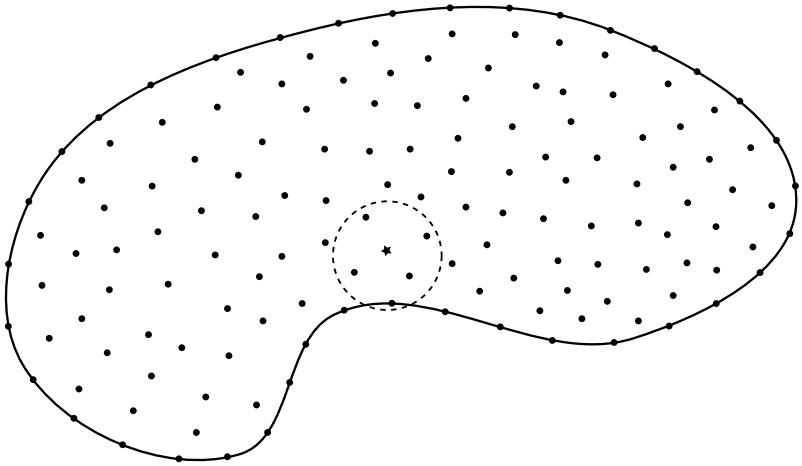
Postavimo se sedaj v splošnejši okvir. Rešujmo parcialno diferencialno enačbo

$$\begin{aligned} \mathcal{L}u &= f \text{ na } \Omega, \\ \mathcal{R}u &= g \text{ na } \partial\Omega, \end{aligned} \tag{3.2}$$

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

Poiščimo sedaj primerno diskretno obliko zgornjega zveznega problema. Izberimo N točk v zaprtju domene, $x_1, \ldots, x_N \in \overline{\Omega}$, od teh naj jih leži nekaj na robu in nekaj v notranjosti Ω. Podobno kot pri končnih diferencah bomo v teh točkah aproksimirali vrednost funkcije u. Izberimo fiksno točko $p \in \overline{\Omega}$ in n izmed točk $\{x_1, \ldots, x_N\}$, ki bodo sestavljali *soseščino* (*angl.* support) točke p. Število n imenujemo velikost soseščine. Označimo z $\mathcal{N}(p)$ soseščino točke p in z $\mathcal{I}(p) = \{i_1, \ldots, i_n\}$ množico indeksov, za katere so izbrani x_{i_j} v soseščini p. Velja torej

$$\mathcal{N}(p) = \bigcup_{i \in \mathcal{I}(p)} x_i.$$



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

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

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

$$u \approx \hat{u} = \sum_{i=1}^{m} \alpha_i b_i = \boldsymbol{b}^{\mathsf{T}} \boldsymbol{\alpha},$$

pri čemer smo z $\boldsymbol{\alpha} = (\alpha_i)_{i=1}^m$ označili vektor neznanih koeficientov in z $\boldsymbol{b} = (b_i)_{i=1}^m$ funkcijo $\boldsymbol{b} \colon \mathbb{R}^d \to \mathbb{R}^m$, 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 nam bodo predstavljale neznane prave vrednosti in nadaljujmo s simbolnim računanjem. Če zahteve za interpolacijo po vrsticah zapišemo v sistem linearnih enačb, dobimo

$$\begin{bmatrix} b_1(x_{i_1}) & \cdots & b_m(x_{i_1}) \\ \vdots & \ddots & \vdots \\ b_1(x_{i_n}) & \cdots & b_m(x_{i_n}) \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_m \end{bmatrix} = \begin{bmatrix} u_{i_1} \\ \vdots \\ u_{i_n} \end{bmatrix}. \tag{3.3}$$

Na krajše sistem zapišemo kot $B\boldsymbol{\alpha} = \tilde{\boldsymbol{u}}$. Odvisno od n, m, b_i in uteži w je ta sistem lahko poddoločen, predoločen ali običajen. Vprašanje obrnljivosti matrike B je primeru $n = m$ težko in je odvisno od izbire funkcij in lege točk. Tudi če so funkcije b_i linearno neodvisne, obstajajo primeri že v dveh dimenzijah, ko zastavljen interpolacijski problem ni korekten [11, str. 79, izrek 2.2] in zagotavljanje korektnosti v splošnem je težek problem.

V vsakem primeru lahko definiramo neke vrste rešitev sistema (3.3), za katero zahtevamo, da minimizira napako v smislu utežene diskretne 2-norme, torej da mi-

nimizira

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

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

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

kjer $+$ označuje Moore-Penroseov psevdoinverz.

To lahko vstavimo nazaj v izraz za $\hat{u}$ in dobimo

$$\hat{u} = \boldsymbol{b}^\mathsf{T}\boldsymbol{\alpha} = \boldsymbol{b}^\mathsf{T}(WB)^+ W\tilde{\boldsymbol{u}}.$$

Sedaj lahko za izbrano točko p izračunamo

$$\hat{u}(p) = \underbrace{\boldsymbol{b}(p)^\mathsf{T}(WB)^+ W}_{\boldsymbol{\varphi}_p}\tilde{\boldsymbol{u}}.$$

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 soseščini $\mathcal{N}(p)$ priredi aproksimacijo za funkcijsko vrednost v točki p.

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

$$(\mathcal{L}u)(p) \approx (\mathcal{L}\hat{u})(p) = (\mathcal{L}\boldsymbol{b})(p)^\mathsf{T}(WB)^+ W\tilde{\boldsymbol{u}}$$

od koder kot prej definiramo

$$\boldsymbol{\varphi}_{\mathcal{L},p} = (\mathcal{L}\boldsymbol{b})(p)^\mathsf{T}(WB)^+ W. \tag{3.4}$$

Funkcional $\boldsymbol{\varphi}_{\mathcal{L},p}$ je aproksimacija operatorja $\mathcal{L}$ v točki p. Pogosto se ga imenuje 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. Tudi če funkcijskih vrednosti $\tilde{\boldsymbol{u}}$ v okolici p ne poznamo, lahko $\boldsymbol{\varphi}_{\mathcal{L},p}$ izračunamo in kasneje samo s skalarnim produktom dobimo aproksimacijo za $(\mathcal{L}u)(p)$. Lahko pa to izkoristimo za zapis linearne enačbe

$$\boldsymbol{\varphi}_{\mathcal{L},p} \cdot \tilde{\boldsymbol{u}} = f(p),$$

ki je direktna aproksimacija diferencialne enačbe (3.2) v točki p,

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

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 $\mathcal{R}$ sedaj aproksimiramo po celi domeni, tako za vsako diskretizacijsko točko x_i v izračunamo $\boldsymbol{\varphi}_{\mathcal{L},x_i}$ in dobimo sistem enačb

$$\begin{aligned}\boldsymbol{\varphi}_{\mathcal{L},x_i} \cdot \tilde{\boldsymbol{u}} &= f(x_i), \text{ za vsak } i, \text{ tak, da je } x_i \in \Omega \\ \boldsymbol{\varphi}_{\mathcal{R},x_i} \cdot \tilde{\boldsymbol{u}} &= g(x_i), \text{ za vsak } i, \text{ tak, da je } x_i \in \partial\Omega.\end{aligned}$$

Te enačbe lahko zapišemo v matrični sistem

$$A\boldsymbol{u} = \boldsymbol{f}, \tag{3.5}$$

kjer ima matrika A v vrsticah zapisane funkcionale $\boldsymbol{\varphi}_{\mathcal{L},x_i}$ tako, da so neničelni elementi na tistih mestih, ki se pomnožijo z neznankami, ki ustrezajo sosedom x_i. Natančneje, elementi matrike A so

$$\begin{aligned}A(k, i_j) &= \boldsymbol{\varphi}_{\mathcal{L},p}(j), \text{ za vsak } k, \text{ tak, da je } x_k \in \Omega \text{ in za vsak } i_j \in \mathcal{I}(x_k), \\ A(k, i_j) &= \boldsymbol{\varphi}_{\mathcal{R},p}(j), \text{ za vsak } k, \text{ tak, da je } x_k \in \partial\Omega \text{ in za vsak } i_j \in \mathcal{I}(x_k).\end{aligned}$$

Razumljivejša je morda kar Matlab-ova notacija

$$A(k, \mathcal{I}(x_k)) = \begin{cases} \boldsymbol{\varphi}_{\mathcal{L},p} & x_k \in \Omega \\ \boldsymbol{\varphi}_{\mathcal{R},p} & x_k \in \partial\Omega \end{cases}, \text{ za } k = 1, \ldots, N.$$

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

$$\boldsymbol{f}(k) = \begin{cases} f(x_k) & x_k \in \Omega \\ g(x_k) & x_k \in \partial\Omega \end{cases}, \text{ za } k = 1, \ldots, N.$$

Vidimo, da je matrika A razpršena. Sama je dimenzij $N \times N$, v vsaki vrstici pa ima največ n neničelnih elementov, torej je skupno število neničelnih elementov

$$\operatorname{nnz}(A) \leq nN.$$

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.

Sistem (3.5) nato rešimo in za aproksimacijo $u(x_i)$ vzamemo $u(x_i) \approx u_i$.

3.2 Posebni primeri

Metoda iz razdelka 3.1.2 je formulirana precej splošno in v posebnih primerih lahko 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* (3.3) *obrnljiva, potem je izbira uteži nepomembna.*

Dokaz. Spomnimo se, da je

$$\boldsymbol{\varphi} = (\mathcal{L}\boldsymbol{b})(p)^{\mathsf{T}}(WB)^{+}W.$$

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

$$\boldsymbol{\varphi} = (\mathcal{L}\boldsymbol{b})(p)^{\mathsf{T}}B^{-1}W^{-1}W = (\mathcal{L}\boldsymbol{b})(p)^{\mathsf{T}}B^{-1},$$

kar je neodvisno od W. □

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 kakšno majhno singularno vrednost več ali manj, kar lahko da precej drugačne rezultate.

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 dvodimenzionalne Poissonove enačbe.

Trditev 3.4. *Metoda iz razdelka 3.1.2 se za reševanje Poissonove enačbe na enakomerni pravokotni mreži z razmakom h ujema z metodo končnih diferenc, če vzamemo $\boldsymbol{b} = \{1, x, y, x^2, y^2\}$, $w \equiv 1$ 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

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

pri čemer spremenljivka $u_{i,j}$ pripada koordinati (x_i, y_j). Če se aproksimaciji Laplaceovega 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 $p = (0, 0)$. Najbližjih n sosedov je tako $\mathcal{N}(p) = \{(0,0), (0,h), (0,-h), (h,0), (-h,0)\}$. Matrika $B = [b_j(x_i)]$ iz sistema (3.3), ki ima po stolpcih izračunane bazne funkcije v vseh sosedih je

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

in njen psevdoinverz je

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

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

$$(\triangle \boldsymbol{b})(p) = \begin{bmatrix} 0 & 0 & 0 & 2 & 2 \end{bmatrix}^\mathsf{T}.$$

Od tod dobimo po zvezi (3.4)

$$\boldsymbol{\varphi} = (\triangle \boldsymbol{b})(p)^\mathsf{T} B^+ = \begin{bmatrix} -\frac{4}{h^2} & \frac{1}{h^2} & \frac{1}{h^2} & \frac{1}{h^2} & \frac{1}{h^2} \end{bmatrix},$$

kar je ravno iskana aproksimacija. □

Opomba 3.5. Metodo za izračun funkcije oblike je zelo elegantno implementirati v Wolfram Mathematici, kar olajša simbolno raziskovanje takih aproksimacij. Zgornje matrike so bile izračunane s programom 1.

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

Program 1: Računanje funkcij oblike na pravokotni mreži.

Opomba 3.6. Tudi če izberemo $n = 9$ in $\boldsymbol{b} = \{1, x, x^2, y, xy, x^2y, y^2, y^2x, y^2x^2\}$, dobimo enako aproksimacijo kot v trditvi 3.4, le da uporabimo več sosedov. Če jih uredimo po oddaljenosti od $(0, 0)$ dobimo aproksimacijo

$$\boldsymbol{\varphi} = \begin{bmatrix} -\frac{4}{h^2} & \frac{1}{h^2} & \frac{1}{h^2} & \frac{1}{h^2} & \frac{1}{h^2} & 0 & 0 & 0 & 0 \end{bmatrix}. \tag{3.6}$$

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

Primer 3.7. 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 $\boldsymbol{b} = \{1, x, y, x^2, y^2, xy\}$. V tem primeru je matrika B velikosti 9×6 in s pomočjo programa 1 dobimo

$$\boldsymbol{\varphi} = \begin{bmatrix} -\frac{4}{3h^2} & -\frac{1}{3h^2} & -\frac{1}{3h^2} & -\frac{1}{3h^2} & -\frac{1}{3h^2} & \frac{2}{3h^2} & \frac{2}{3h^2} & \frac{2}{3h^2} & \frac{2}{3h^2} \end{bmatrix}.$$

Vidimo, da tokrat v aproksimaciji upoštevamo vseh 9 sosedov, za razliko od aproksimacije v opombi 3.6. Naravno sledi vprašanje, ali je ta aproksimacija kaj slabša,

morda drugačnega reda? Izkaže se, da ne, saj z razvojem aproksimacij v Taylorjevo vrsto dobimo

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

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

Primer 3.8. Poglejmo še, kaj se zgodi, če za bazne funkcije vzamemo radialne bazne funkcije. V tem primeru je smiselno vzeti $n = m$, 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).

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

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

in za $\boldsymbol{b}$ vzamemo

$$\boldsymbol{b} = \{x \mapsto \exp((x-c)^2/\sigma^2); c \in \mathcal{N}(p)\},$$

potem zopet s pomočjo programa 1 izračunamo aproksimacijo

$$\boldsymbol{\varphi} = \frac{4}{(e^{\frac{2h^2}{\sigma^2}}-1)^2\sigma^4}\left[(e^{\frac{2h^2}{\sigma^2}}-1)^2-4h^2e^{\frac{2h^2}{\sigma^2}} \quad h^2e^{\frac{h^2}{\sigma^2}} \quad h^2e^{\frac{h^2}{\sigma^2}} \quad h^2e^{\frac{h^2}{\sigma^2}} \quad h^2e^{\frac{h^2}{\sigma^2}} \quad 0 \quad 0 \quad 0 \quad 0\right]\cdot \tag{3.7}$$

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

$$\boldsymbol{\varphi}\cdot\tilde{\boldsymbol{u}} - \triangle u(0,0) = h^2\left(\frac{2u(0,0)}{\sigma^2}-\frac{\triangle u(0,0)}{\sigma^2}+\frac{1}{12}\left(\frac{\partial^4 u}{\partial x^4}(0,0)+\frac{\partial^4 u}{\partial y^4}(0,0)\right)\right)+O(h^4).$$

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

Trditev 3.9. *Aproksimacija Laplaceovega operatorja z devetimi Gaussovimi baznimi funkcijami konvergira proti aproksimaciji z monomi, ko gre $\sigma \to \infty$.*

Dokaz. Obe aproksimaciji smo izračunali že v primeru 3.8 in opombi 3.6. Če pogledamo limito izraza (3.7), ko gre $\sigma \to \infty$, dobimo izraz (3.6). □

3.3 Algoritem

V izpeljavi metode v razdelku 3.1.2 je veliko detajlov ostalo neizdelanih, kot na primer, kako diskretiziramo domeno ali kako poiščemo sosede. Ti detajli so opisani v naslednjih podrazdelkih, celotno metodo pa podajamo v psevdokodi kot algoritem 1.

Algoritem 1 Brezmrežna metoda za reševanje PDE iz razdelka 3.1.2.

Vhod: Parcialna diferencialna enačba, kot opisana v (3.2). 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: Skalarno polje u, ki aproksimira rešitev enačbe (3.2).

function REŠI($\Omega, \mathcal{L}, f, \mathcal{R}, g, N, Q, n, m, b, w$)
  $x \leftarrow$ DISKRETIZIRAJ(Ω, N, Q) ▷ x postane seznam N točk, brez škode za splošnost naj leži prvih Q točk v Ω in preostalih $N - Q$ na $\partial\Omega$.
  $s \leftarrow$ SOSEDI(x, n) ▷ s je seznam dolžine N, pri čemer je $s[i]$ seznam indeksov elementov v x, ki so sosedi $x[i]$, vključno z i.
  $\varphi \leftarrow$ prazen seznam dolžine N.
  for $i \leftarrow 1$ **to** Q **do** ▷ Izračunamo funkcije oblik v notranjosti.
    $\varphi[i] \leftarrow$ FUNKCIJAOBLIKE($\mathcal{L}, x[i], x, s[i], n, m, b, w$) ▷ Glej algoritem 2.
  end for
  for $i \leftarrow Q + 1$ **to** N **do** ▷ Izračunamo funkcije oblik na robu.
    $\varphi[i] \leftarrow$ FUNKCIJAOBLIKE($\mathcal{R}, x[i], x, s[i], n, m, b, w$) ▷ Glej algoritem 2.
  end for
  $A \leftarrow$ prazna razpršena $N \times N$ matrika
  for $i \leftarrow 1$ **to** N **do** ▷ Aproksimiramo enačbo.
    for $j \leftarrow 1$ **to** n **do**
      $A[i, s[j]] \leftarrow \varphi[i][j]$
    end for
  end for
  $r \leftarrow$ prazen vektor dolžine N
  for $i \leftarrow 1$ **to** Q **do** ▷ Izračunamo desno stran v notranjosti.
    $r[i] \leftarrow f(x[i])$
  end for
  for $i \leftarrow Q + 1$ **to** N **do** ▷ Izračunamo robne pogoje.
    $r[i] \leftarrow g(x[i])$
  end for
  $u \leftarrow$ REŠIRAZPRŠENSISTEM(A, r)
  return u
end function

3.3.1 Diskretizacija

Splošno d-dimenzionalno domeno Ω je težko dobro diskretizirati. Želimo si, da bi bile točke v domeni čim bolj enakomerno razporejene, saj to pomeni, da smo dobro popisali celotno področje in upamo, da s tem tudi obnašanje funkcije u, hkrati pa si zaradi numerične stabilnosti ne želimo, da bi bile točke preveč skupaj. Uvedimo dve količini, ki nam merita ti dve lastnosti. Za dano domeno Ω in množico točk X

Algoritem 2 Izračun funkcije oblike.

Vhod: Parametri metode:

- $\mathcal{L}$ ... parcialen diferencialen operator
- p ... točka, v kateri aproksimiramo operator
- x ... seznam diskretizacijskih točk
- I ... množica indeksov točk v soseščini 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.

function FUNKCIJAOBLIKE($\mathcal{L}, p, x, I, n, m, b, w$)
  $W \leftarrow$ prazen vektor dolžine n
  for $i \leftarrow 1$ **to** n **do**
    $W[i] \leftarrow \sqrt{w(x[I[i]] - p)}$
  end for
  $B \leftarrow$ prazna matrika velikosti $n \times m$
  for $i \leftarrow 1$ **to** n **do**
    for $j \leftarrow 1$ **to** m **do**
      $B[i, j] \leftarrow W[i] \cdot b[j](x[I[i]])$
    end for
  end for
  $\ell \leftarrow$ prazen vektor dolžine m
  for $j \leftarrow 1$ **to** m **do**
    $\ell[j] \leftarrow (\mathcal{L}(b[j]))(p)$
  end for
  $\varphi \leftarrow (\ell \cdot \text{PINV}(B)) \odot W$ ▷ Direktna analogija enačbe (3.4), $\odot$ označuje
  return φ ▷ Hadamardov produkt.
end function

definirajmo

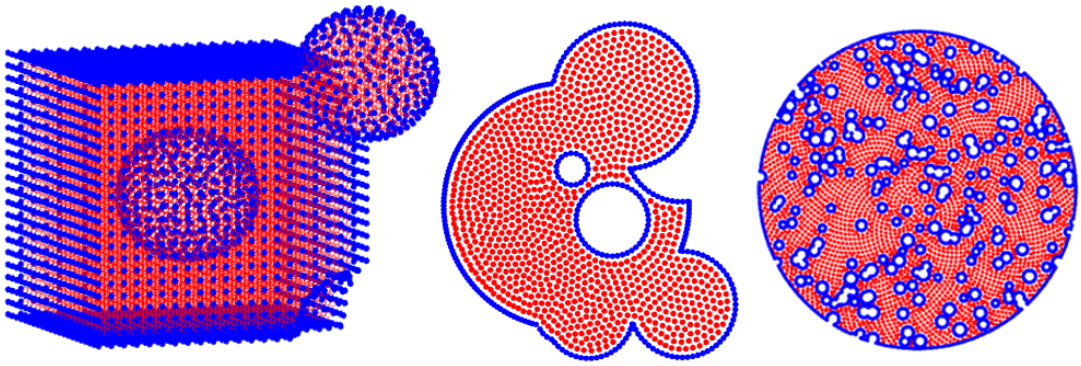
$$h_\Omega(X) = \max_{p \in \Omega} \min_{x \in X} \|p - x\| \tag{3.8}$$
$$S(X) = \min_{\substack{x,y \in X \\ x \neq y}} \|x - y\|.$$

Količina h pove, da ne glede na to, kje v domeni smo, imamo na razdalji manj ali enako h vsaj eno diskretizacijsko točko. Količina S pa pove, kako blizu so si diskretizacijske točke med seboj. Dobra diskretizacija želi maksimizirati S in minimizirati h.

Algoritmi za diskretizacijo so odvisni od načina, kako podamo domeno. V našem primeru se izkaže, da je za trenutne potrebe dovolj, da podpiramo kvadre in krogle do vključno treh dimenzij, kot tudi unije in razlike osnovnih oblik.

Tako lahko za diskretizacijo kvadra uporabimo kar enakomerno diskretizacijo. Prav tako ni težko ugotoviti, kdaj so točke na robu. Za čimbolj enakomerno diskretizacijo notranjosti kroga ali površine sfere lahko uporabimo npr. Fibonaccijevo mrežo, kot predlagano v [12] ali v [13]. Pri razlikah domen preprosto izbrišemo

točke, ki so padle izven domene, pri unijah pa naredimo tudi unijo diskretizacij. Pri tem lahko potem naredimo še en korak in pobrišemo ven kakšno izmed točk, ki so si preblizu skupaj. Primeri domen in njihovih diskretizacij, dobljenih na ta način, so prikazani na sliki 6.



Slika 6: Primeri domen in njihovih diskretizacij.

Pri enakomerni diskretizaciji smo močno uporabili naše znanje o domeni in njeni obliki. Lahko pa, da tega nimamo na voljo in želimo splošnejši algoritem, ki potrebuje npr. samo karakteristično funkcijo domene in njene meje, torej, kakšen kvader, ki našo domeno vsebuje. V tem primeru lahko vzamemo enakomerno diskretizacijo kvadra in odstranimo vse točke, ki niso v domeni, vendar je tu težko kontrolirati število točk. Druga metoda je, da naključno izbiramo točke v kvadru, in sprejmemo tiste, ki pristanejo v notranjosti. Še boljšo diskretizacijo dobimo, če namesto psevdonaključnih števil uporabimo kvazinaključna števila, ki imajo manjšo diskrepanco. Več o tem si bralec lahko prebere v [14].

Kljub njihovi splošnosti so naključne diskretizacije precej slabe, kot na primer naključna diskretizacija kroga na sliki 7 (levo). Pri tem, da bi točke v domeni porazdelili čimbolj 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 in pustimo fiziki, da opravi svoje delo. Na vsako točko tako deluje neka sila, 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 in če kakšna točka iz notranjosti zleze iz domene, jo postavimo na naključno mesto nazaj v domeno. Zapis zgornjega postopka je v psevdokodi podan kot algoritem 3. Postopku se kdaj reče tudi *sprostitev* (*angl.* relaxation) domene, kajti na začetku so med točkami napetosti, ki jih s premikanjem poskušajo minimizirati. Rezultati na primeru izboljšanja naključne diskretizacije kroga so podani na sliki 7.

Da algoritem izboljša kvaliteto aproksimacije se vidi tudi, če izračunamo parametra h in S. Pričakujemo, da se bo S povečal, saj se točke, ki so bolj skupaj, bolj odbijajo. Na sliki 8 vidimo primerjavo med kvaliteto diskretizacije s Fibonaccijevo mrežo in njeno 10-kratno izboljšavo. Vidimo, da se h ni bistveno spremenil, le varianca se mu je malce povečala, S pa se je v povprečju precej izboljšal in izkazuje celo lepše obnašanje kot prej.

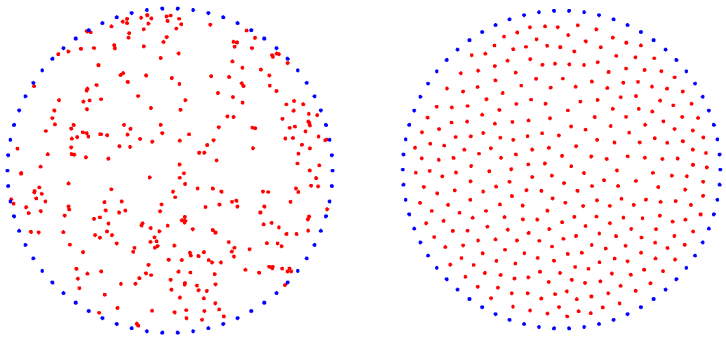
Algoritem 3 Algoritem za izboljšanje kvalitete diskretizacije domene.

Vhod: Parametri metode:

- Ω ... domena
- N ... število diskretizacijskih točk
- Q ... število diskretizacijskih točk v notranjosti Ω
- 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.

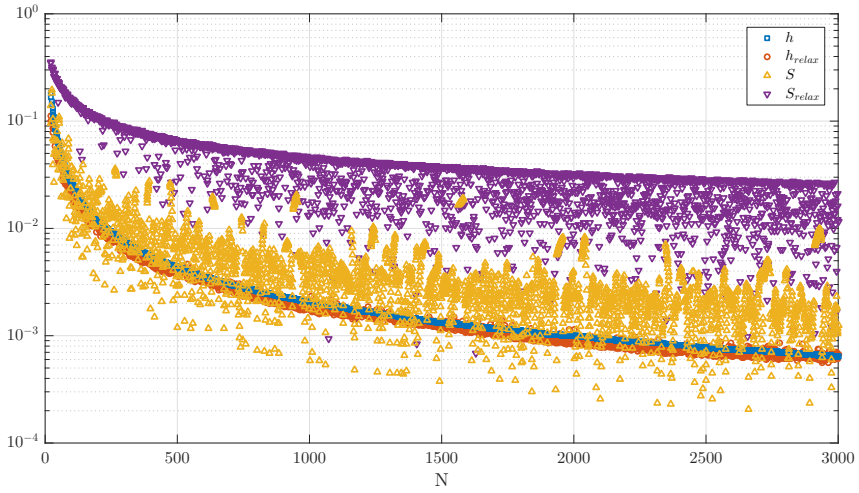
1: **function** IZBOLJŠAJ($\Omega, N, Q, X, I, s, F_0, \alpha$)
2: $r_\chi \leftarrow \left(\frac{\text{VOLUMEN}(\Omega)}{N}\right)^{\frac{1}{\text{DIMENZIJA}(\Omega)}}$ ▷ Približek povprečne razdalje med točkami.
3: **for** $i \leftarrow 1$ **to** I **do**
4: **for** $j \leftarrow 1$ **to** Q **do**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



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).

3.3.2 Iskanje najbližjih sosedov

Tako v algoritmu 1 v vrstici 3, kot v algoritmu 3 v vrstici 6 potrebujemo najti nekaj najbližjih sosedov dane točke. Navadno za okolico izberemo kar n najbližjih sosedov



Slika 8: Sprememba h in S po 10 iteracijah algoritma 3 s parametri $F_0 = 10^{-2}$, $s = 6$, $\alpha = 3$ v enotskem dvodimenzionalnem krogu z začetno Fibonaccijevo mrežo, v odvisnosti od N, z razmerjem robnih in notranjih točk $x : \frac{12}{\pi}\sqrt{x}$.

v evklidski metriki, vendar bi lahko soseščino definirali tudi drugače, na primer kot množico vozlišč, ki so oddaljeni manj kot neka fiksna razdalja R. Pri tem pristopu bi imeli manjšo kontrolo nad velikostjo soseščine, zato izberemo prvega.

Problem iskanja najbližjih sosedov (*angl.* nearest neighbour search) je znana in dobro raziskana tema z razvitimi 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 nam omogoča logaritemski dostop. Med bolj znanimi strukturami so *k-d tree* [15], *ball tree* [16] in *cover tree* [17]. V članku [18] 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 tree*. Poleg tega *k-d tree* najboljše deluje tudi, ko imajo točke, v katerih bomo iskali sosede, podobno porazdelitev kot točke, iz katerih smo naredili drevo, kar v našem primeru drži. Če vključimo še dejstvo, da je *k-d tree* 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 uporabljena implementacija je predstavljena v [19], 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.

3.3.3 Reševanje razpršenega sistema

Za reševanje razpršenih sistemov poznamo veliko različnih metod, ki se v grobem delijo na direktne (obsežno opisane v [20]) in iterativne (obsežno opisane v [21]). 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 [22] poskušajo

s preureditvijo 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, tako da moramo za višjo natančnost porabiti večje število računskih operacij.

Za metode iz numerične linearne algebre bomo uporabljali knjižnico Eigen [23], ki nudi elegantno in hitro delo z matrikami v C++. Vgrajenih ima veliko direktnih in iterativnih metod za reševanje sistemov linearnih enačb.[1] Pri sistemih tipa (3.5) se je v praksi najbolje obnesel BiCGSTAB [24] z ILUT predpogojevanjem [25]. 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 konvergence 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 nam tako parameter p kontrolira kolikokrat več spomina dovolimo za hranjenje predogojenke, parameter τ pa, kako natančna bo LU faktorizacija.

3.3.4 Časovna zahtevnost

V tem razdelku analiziramo časovno zahtevnost metode. Predpostavili bomo, 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 in ne bo nastopala v analizah.

Trditev 3.10. *Pričakovana časovna zahtevnost algoritma 1 je*

$$O(INs \log^2 N + (N+n) \log N + m^2 nN) + T, \tag{3.9}$$

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 traja $O((N+n) \log N)$ časa za konstrukcijo drevesa in iskanje sosedov.

Izvajanje funkcije FUNKCIJAOBLIKE traja po $O(n + nm + m + m^2 n + nm + n) = O(mn^2)$, kjer je prevladujoči faktor $O(m^2 n)$ rezultat računanja SVD razcepa z dvostranskim Jacobi SVD razcepom iz knjižnice Eigen.[2] Funkcija oblike se izračuna za vsako točko v domeni, torej ta del algoritma traja $O(m^2 nN)$ časa.

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

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

Pri sestavljanju razpršene matrike lahko v naprej rezerviramo prostor za n elementov v vsaki vrstici, tako da nas 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 $O(INs \log^2 N + (N+n) \log N + m^2 N) + T$, kjer je T čas, porabljen za reševanje razpršenega sistema enačb. □

3.3.5 Prostorska zahtevnost

V tem razdelku analiziramo časovno zahtevnost metode. 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 in ne bo nastopala v analizah.

Trditev 3.11. *Prostorska zahtevnost algoritma 1 je $O(nN) + P$, kjer je P prostor, ki ga potrebujemo, za reševanje sistema linearnih enačb.*

Dokaz. Pri funkciji DISKRETIZIRAJ potrebujemo $O(N)$ spomina za shrambo 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 FUNKCIJAOBLIKE 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 $O(nN) + P$. □

3.4 Pogoste vrednosti parametrov

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

Definicija 3.12. Količino r_χ, pripisano diskretizaciji X domene Ω, izračunano z

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

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 $N = |X|$ 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 $\frac{N}{N-1}$ 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. To je težje izračunati, če nimamo že zgrajenega drevesa, zato se ponavadi na enostavnejših primerih poslužujemo kar r_χ. Kadar pa to ni dobra aproksimacija razdalje med vozlišči za celo domeno, kot na primer ob goščenju mreže, se poslužimo 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 regularne domene 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 daje 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

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

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 sosedi, ki so dlje kot r_χ stran, v aproksimaciji ne upoštevajo, ne glede na to kako velik n smo izbrali. 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 $\mathbb{P}_\ell$ 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.

Definicija 3.13. Funkcija ψ_c se imenuje radialna bazna funkcija s centrom v točki c, če je odvisna samo od razdalje do centra, torej

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

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

V tabeli 2 so naštete najpogostejše uporabljene radialne bazne funkcije, povzete po [26, str. 5].

Ko si izberemo neko bazno funkcijo ψ, potem za bazne funkcije okoli točke p vzamemo

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

Pri izbiri parametra oblike σ_b moramo tudi biti do neke mere pazljivi. Izbrati ga moramo dovolj velikega, da se funkcije "prekrivajo", torej tako, da nima funkcija vrednosti blizu nič pri vseh sosedih razen pri sebi. Hkrati mora biti parameter dovolj majhen, da sistem (3.3) ni preveč nestabilen. Bolj natančna analiza izbire parametra oblike za radialne bazne funkcije in za utež je narejena na sliki 14 med numeričnimi zgledi na strani 52. V [27] je tudi pokazano, da izbire velikih parametrov oblike nudijo visoko natančnost za ceno numerične stabilnosti in obratno.

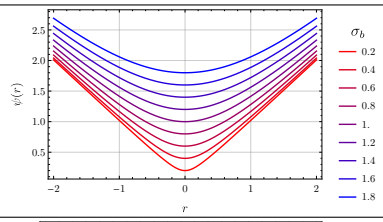
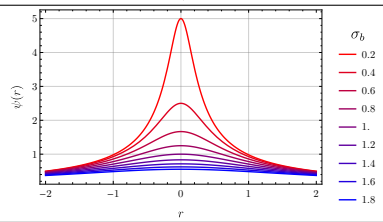
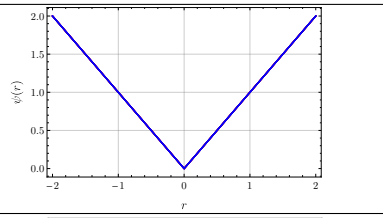
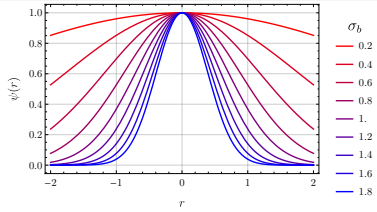
Multikvadratične (MQ) (*angl.* multiquadric) $\psi(r) = \sqrt{r^2 + \sigma_b^2}$	
Inverzne multikvadratične (IMQ) (*angl.* Inverse multiquadrics) $\psi(r) = 1/\sqrt{r^2 + \sigma_b^2}$	
Linearna razdalja (L) (*angl.* Linear distance) $\psi(r) = r$	
Gaussove funkcije (G) (*angl.* Gaussians) $\psi(r) = \exp(-(r/\sigma_b)^2)$	

Tabela 2: 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 konvergence aproksimacije z radialnimi baznimi funkcijami si lahko bralec prebere v [28].

3.5 Višjedimenzionalni problemi

Numerično zelo pogosto rešujemo probleme, kjer je neznana količina vektorsko polje in ne le skalarno polje, kot smo predpostavili v izpeljavi v razdelku 3.1.2. Navier-Stokesove enačbe in Navierova enačba sta najbolj osnovna zgleda vektorskih enačb. Pri teh problemih moramo poprijeti po drugačnih metodah, najenostavnejše preprosto obravnavamo enačbo za vektorsko polje kot tri sklopljene enačbe za tri skalarna polja. Oglejmo si postopek na primeru Navierove enačbe. Vektorsko polje razpišemo po komponentah, za primer treh dimenzij zapišemo $\vec{u} = (u, v, w)$. Vektorska oblika stacionarne Navierove enačbe

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

se razpiše v sistem treh enačb

$$
\begin{aligned}
(\lambda+\mu)\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 v}{\partial xy}+\frac{\partial^2 w}{\partial xz}\right)+\mu\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}+\frac{\partial^2 u}{\partial z^2}\right)+f_1=0\\
(\lambda+\mu)\left(\frac{\partial^2 u}{\partial xy}+\frac{\partial^2 v}{\partial y^2}+\frac{\partial^2 w}{\partial yz}\right)+\mu\left(\frac{\partial^2 v}{\partial x^2}+\frac{\partial^2 v}{\partial y^2}+\frac{\partial^2 v}{\partial z^2}\right)+f_2=0\\
(\lambda+\mu)\left(\frac{\partial^2 u}{\partial xz}+\frac{\partial^2 v}{\partial yz}+\frac{\partial^2 w}{\partial z^2}\right)+\mu\left(\frac{\partial^2 w}{\partial x^2}+\frac{\partial^2 w}{\partial y^2}+\frac{\partial^2 w}{\partial z^2}\right)+f_3=0.
\end{aligned}
$$

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 $3N\times 3N$ za tri skalarna polja hkrati. Neznanke, ki ustrezajo vektorju $\vec{u}$ predstavimo numerično kot vektor $\boldsymbol{u}$ dolžine $3N$, katerega prvih N komponent predstavlja u, naslednjih N predstavlja v, zadnjih N pa w. Enako storimo z $\vec{f}$ in robnimi pogoji. Enačbo $A\boldsymbol{u}=\boldsymbol{f}$ lahko sedaj zapišemo po blokih

$$
\begin{bmatrix} U1 & V1 & W1\\ U2 & V2 & W2\\ U3 & V3 & W3 \end{bmatrix}\begin{bmatrix} u\\ v\\ w \end{bmatrix}=\begin{bmatrix} f_1\\ f_2\\ f_3 \end{bmatrix}.
$$

Vsako izmed treh komponent Navierove enačbe bomo napisali v eno izmed (bločnih) vrstic. Funkcijo oblike za člen $\frac{\partial^2 v}{\partial xy}$ bi napisali v blok $V1$, saj nastopa v prvi enačbi in se nanaša na polje v, podobno bi aproksimacijo za $\frac{\partial^2 w}{\partial z^2}$ iz tretje enačbe zapisali v blok $W3$. Če bi katere elemente napisali na mesta, kjer so že kakšni neničelni elementi, jih med seboj seštejemo.

Pri zgornjem postopku smo operatorja $\triangle$ in grad div razbili na vsoto več elementarnih operatorjev in kar sešteli njihove aproksimacije, da smo dobili aproksimacijo celotnega operatorja. Dovoljenje za to nam da naslednja trditev.

Trditev 3.14. *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.*

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

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

□

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.

4 Implementacija

blah blah [29].

5 Osnovni numerični zgledi

V tem razdelku bomo pogledali obnašanje metode na osnovnih numeričnih zgledih, da si zgradimo intuicijo o njenem delovanju. V nadaljnjem besedilu in na vseh grafih bo metoda, opisana v razdelku 3 označena s kratico MLSM (*angl.* Meshless Least Squares Method). Če ni drugače navedeno, so vse časovne meritve narejene na prenosnem računalniku z štirijedrnim procesorjem `Intel(R) Core(TM) i7-4700MQ CPU @2.40GHz`, pri čemer vsako jedro podpira dve niti izvajanja, in 16 GB DDR3 pomnilnika. Vsi programi so bili prevedeni s prevajalnikom `g++` verzije 7.1.7 in stikali `-std=c++14 -O3 -DNDEBUG` na operacijskem sistemu Linux. Pri paralelizaciji je bila uporabljena knjižnica OpenMP [30] za paralelizacijo z deljenim pomnilnikom.

5.1 Enodimenzionalni robni problem

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

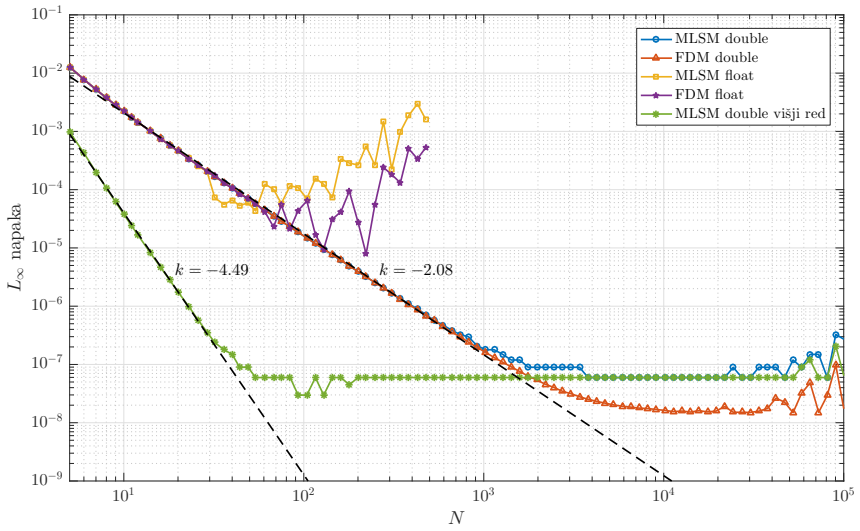
$$\begin{aligned} u''(x) &= \sin(x), \quad x \in (0,1) \\ u(0) &= 0, \\ u'(1) &= 0, \end{aligned}$$

ki ima analitično rešitev $u(x) = \cos(1)x - \sin(x)$. Na sliki 9 je prikazana konvergenca metode končnih diferenc in MLSM. Metoda MLSM je bila uporabljena s parametri, pri katerih velja trditev 3.4, torej $n = m = 3$, $w = 1$, $\boldsymbol{b} = \{1, x, x^2\}$. Pri obeh metodah smo uporabi dvojno in enojno natančnost računanja s plavajočo vejico. Za primerjavo smo rešili problem tudi z višjim redom $n = m = 5$, $w = 1$, $\boldsymbol{b} = \{1, x, x^2, x^3, x^4\}$. V obeh primerih smo uporabili direktno metodo SuperLU za reševanje sistema enačb. Za vsakega od teh primerov smo izračunali diskretno L_∞ napako v diskretizacijskih točkah:

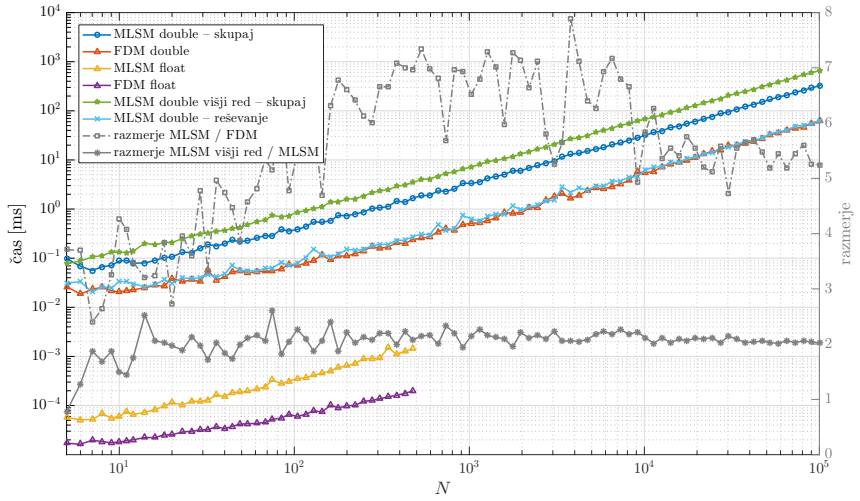
$$L_\infty = \max_{x \in X} |\hat{u}(x) - u(x)|.$$

Na grafu 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 natančnosti, med $N = 10^4$ in $N = 10^5$ diskretizacijskimi točkami pa se začnejo pojavljati numerične nestabilnosti. Manjšo končno natančnosti MLSM lahko pripišemo 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 uporabimo za računanje aproksimacije odvoda 5 sosedov namesto 3, po pričakovanjih dobimo metodo, ki konvergira z višjim redom, ima pa enako končno natančnost kot prej.

Časovna primerjava obeh metod je prikazana na sliki 10. 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 računa aproksimacije, medtem je to pri FDM izračunano na roke. Če pri MLSM posebej merimo samo čas, ki ga porabimo za sestavljanje matrike in reševanje sistema,



Slika 9: Napaka FDM in MLSM metode v primerjavi s pravilno rešitvijo.



Slika 10: Primerjava časa izvajanja MLSM in FDM metod.

vidimo, da se skoraj ujema z časom, porabljenim pri FDM. Minimalno razliko gre pripisati razliki pri načinu grajenja matrike, saj pri FDM natanko vemo pozicije in število neničelnih elementov vnaprej, pri MLSM pa v splošnem ne. MLSM višjega reda je pričakovano počasnejši saj sta n in m poleg osnovnega faktorja N glavna parametra v časovni zahtevnosti (3.9), iz razmerja vidimo, da je konsistentno dvakrat počasnejši. Toda, če pogledamo koliko časa potrebujemo, da dosežemo natančnost 10^{-6}, z grafov 9 in 10 odčitamo, da MLSM višjega reda potrebuje le $N = 22$, za kar potrebujemo 0.3 ms, pri običajnem redu pa potrebujemo $N = 400$ kar nanese 1.5 ms. Implementacija deljenih diferenc višjega reda je dovolj zoprna, da se zanjo malokrat odločimo, saj je treba uporabljati enostranske diference pri robu. Pri MLSM pa se izračunajo same, vse kar je potrebno je, da n iz 3 spremenimo na 5. Če uporabljamo enojno natančnost namesto dvojne se to pozna na času, ki ga porabimo za reševanje, vendar je enojna natančnost ponavadi neuporabna zaradi zgodnje nestabilnosti.

5.2 Poissonova enačba

Oglejmo si sedaj obnašanje metode na klasični Poissonovi enačbi na kvadratu:

$$\begin{aligned} \triangle u = 1, \quad (x,y) \in (0,1)\times(0,1) \\ u(x,0) = u(x,1) = u(0,y) = u(1,y) = 0. \end{aligned} \tag{5.1}$$

Analitično rešitev lahko dobimo s pomočjo separacije spremenljivk in se glasi

$$u(x,y) = -8 \sum_{\substack{k=1 \\ k \text{ lih}}}^{\infty} \frac{\sin(k\pi x) \sinh \frac{k\pi(1-y)}{2} \sinh \frac{k\pi y}{2}}{\cosh(\frac{k\pi}{2}) k^3 \pi^3}. \tag{5.2}$$

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

Trditev 5.1. *Za ostanek vrste* (5.2) *velja*

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

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

Dokaz. Ocenjujmo 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 nam ostane za oceniti številska vrsta

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

Uporabimo še 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). □

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

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

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 (5.2) za $\ell = 59$ manjši kot 10^{-5}, za $\ell = 181$ pa manjši kot 10^{-6}. Za izračun analitične rešitve ob primerjavi z numerično je bil uporabljen $\ell = 181$.

Opomba 5.2. Računanje izraza

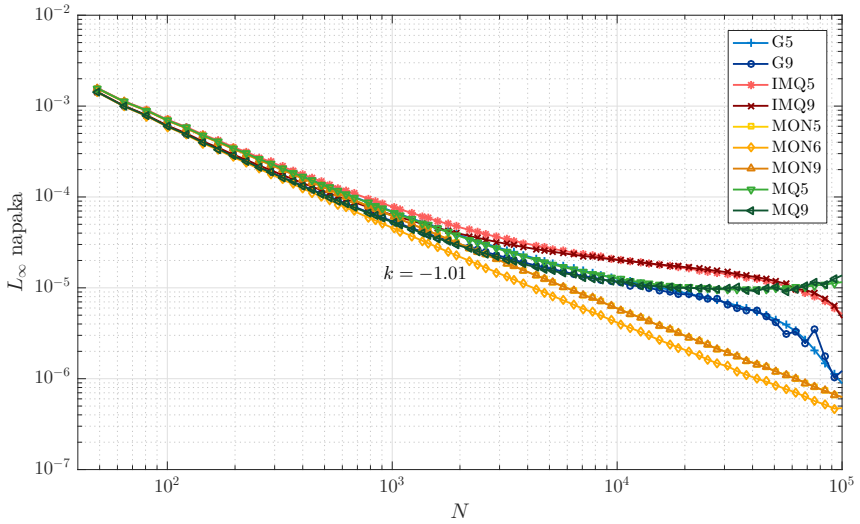
$$\frac{\sinh\frac{k\pi(1-y)}{2}\sinh\frac{k\pi y}{2}}{\cosh(\frac{k\pi}{2})}$$

ni numerično stabilno, ko k raste, kajti tako števec kot imenovalec se približujeta neskončno, kvocient pa ima končno limito. Ko za želimo izračunati numerično je bolje uporabiti stabilnejšo obliko

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

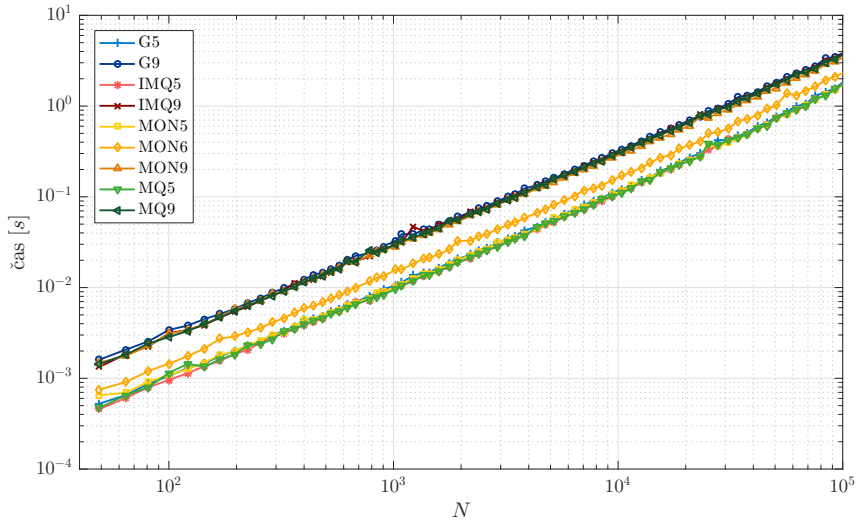
kjer vedno nastopa negativen eksponent.

Rešimo problem (5.1) numerično, s štirimi različnimi nabori baznih funkcij: monomi, Gaussovimi funkcijami, multikvadratičnimi in inverznimi multikvadratičnimi. Za parametre smo vedno vzeli $n = m = 9$ kar sovpada s primeri iz razdelka 3.2. Pri monomih smo dodali tudi primer $n = 9$ in $m = 6$. Za parameter oblike pri radialnih baznih funkcijah smo vzeli $\sigma_b = 100\, r_\chi$. Za utež vzemimo Gaussovo utež z $\sigma_w = \frac{3}{4} r_\chi$. V vseh primerih je bila za reševanje sistema enačb uporabljena direktna metoda SuperLU. Iterativni BiCGSTAB algoritem je dal enake rezultate. Napaka metod je prikazana na sliki 11.



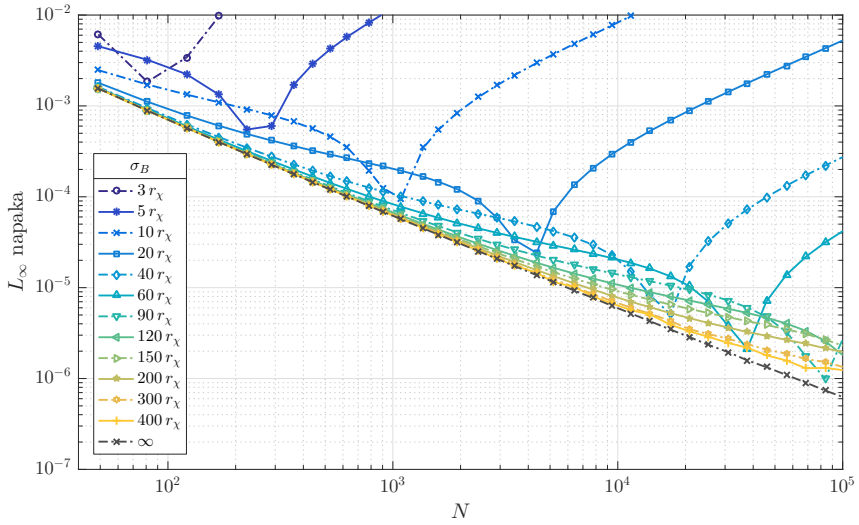
Slika 11: Konvergenca MLSM za različne parametre pri reševanju problema (5.1).

Vidimo, da monomi za $n = 5$ in $n = 9$ konvergirajo linearno z redom 1, kot vemo iz teorije končnih diferenc. Radialne bazne funkcije se na začetku ujemamo z monomi, nato pa pokažejo slabše konvergenčne lastnosti in nelinearno obnašanje, drugačno pri vsakem razredu funkcij. Funkcije, ki uporabljajo več sosedov, začnejo z malenkost nižjo napako, prav tako pa morda presenetljivo tudi monomi z $n = 6$. S slike 12, 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.



Slika 12: Čas izvajanja za različne izbore parametrov pri reševanju problema (5.1).

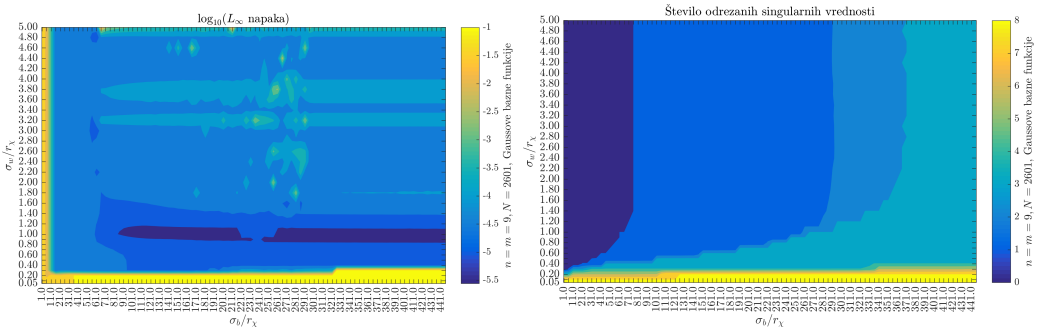
Po trditvi 3.9 aproksimacija z Gaussovimi funkcijami konvergira proti aproksimaciji z monomi. Na sliki 13 vidimo to dejstvo demonstrirano tudi numerično. Konvergenčne krivulje za Gaussove funkcije se pri povečevanju parametra oblike čedalje bolj približujejo konvergenčni krivulji monomov. Tako obnašanje zasledimo tudi pri bolj kompliciranih enačbah in drugih radialnih baznih funkcijah. Če imamo pri radialnih baznih funkcijah težavo s konvergenco, lahko poiskusimo povečati parameter oblike. Seveda to ne gre prek vseh meja, saj sistem (3.3) za izračun funkcije oblike postane čedalje bolj občutljiv.



Slika 13: Konvergenčne krivulje za Gaussove funkcije pri čedalje večjem σ in za monome ($\sigma = \infty$). Uporabili smo $n = m = 5$ in Gaussovo 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 nam bo pomagala pri izbiri parametrov pri bolj zapletenih problemih. Problem (5.1) 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 14a.



(a) Napaka v odvisnosti od σ_b in σ_w.

(b) Število odrezanim singularnih vrednosti v odvisnosti of σ_b in σ_w.

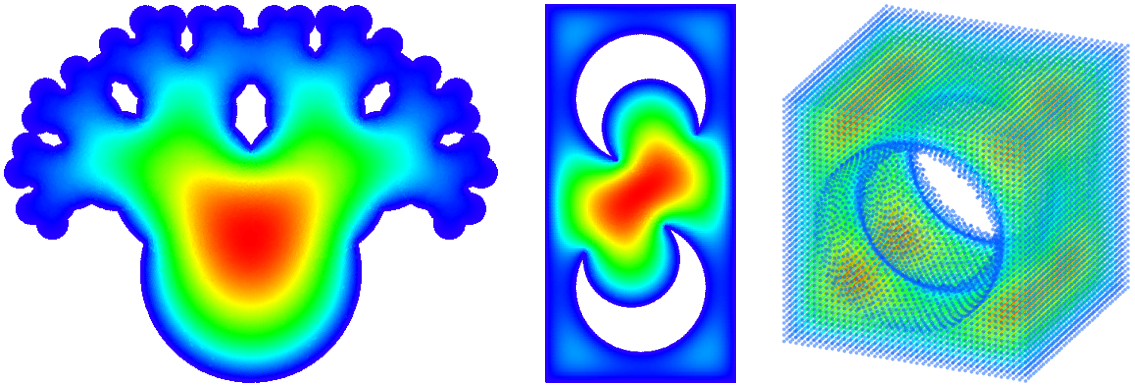
Slika 14: Reševanje problema (5.1) z Gaussovimi funkcijami in Gaussovo utežjo pri $n = m = 9$ in $N = 2601$.

Trditev 3.2 pravi, da mora v primeru $n = m$ ni obrnljive matrike B aproksimacija operatorja (in posledično tudi napaka) neodvisna od izbire uteži. Opomba po trditvi pravi, da numerično to velja, če le utež ni premajhna, da bi bila sama razlog za numerične nestabilnosti. Slika 14a 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 (3.3) 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 1.00041 in 1. Boljši vpogled v obrnljivost matrike B nam da graf na sliki 14b, ki predstavlja število odrezanim singularnih vrednosti pri SVD razcepu, s pomočjo katerega smo izračunali psevdoinverz matrike WB. Aproksimacija je res neodvisna od 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 neničelno 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 pomaga, kot vidimo v pasu nizke napake okoli $\sigma_w = 1$. Na sliki 14b vidimo, da se z večanjem σ_b reže čedalje več singularnih vrednosti in s časoma bi aproksimacija zopet 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 diferenc 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 15 so prikazane rešitve Poissonove enačbe $\triangle u = 1$ 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, vsa druga koda je ostala enaka.



Slika 15: Reševanje Poissonove enačbe $\triangle u = 1$ s homogenimi robnimi pogoji na zanimivejših domenah.

5.3 Hertzev kontaktni problem

Hertzev kontaktni problem je leta 1882 v svojem članku "Über die Berührung fester elastischer Körper." [31] obravnaval že Heinrich Hertz. Ko dve ukrivljeni telesi z različnima radijema ukrivljenosti staknemo, se na začetku dotikata le v točki ali na premici. Čim ti telesi pritisnemo skupaj z neko silo, se elastično deformirata in med njima se ustvari stična površina, prek katere poteka vsa interakcija. Glavni objekti zanimanja v elastični kontaktni mehaniki so normalne in tangencialne napetosti med telesi, ki nastanejo kot posledice medsebojnega pritiska in 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 [32, str. 122, poglavje 3.2]. Hertzeva teorija ima poleg nelepljivega stika naslednje predpostavke:

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_1,

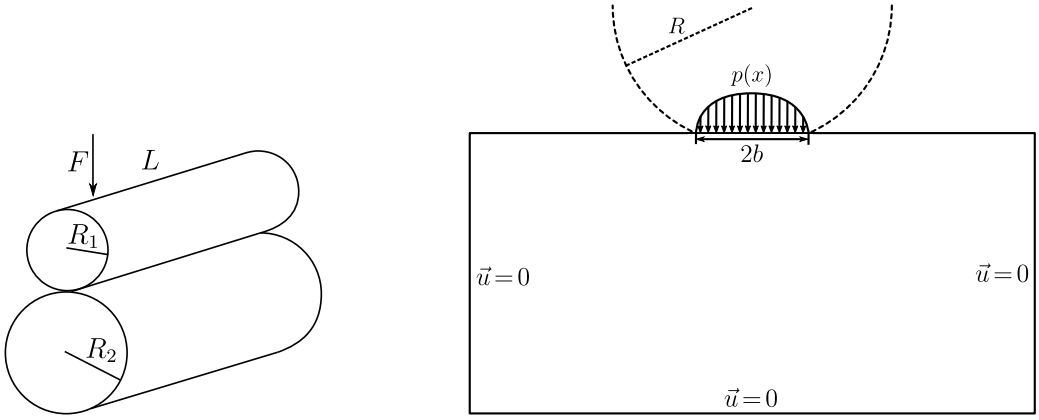
Youngovima moduloma E_1 in E_2 in Poissonovima razmerjema ν_1 in ν_2. Situacija je prikazana na sliki 16a. Napovedano območje stika med valjema je širine $2b$, kjer je

$$b = 2\sqrt{\frac{FR}{\pi E^*}},$$

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

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

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



(a) Stik dveh vzporednih valjev.

(b) Robni pogoji za numerično rešitev Hertzevega kontaktnega problema med valjem in polravnino.

Slika 16: Obravnavan Hertzev kontaktni problem.

Numerično želimo izračunati pomike in napetosti v materialu, zato rešimo stacionarno Navierovo enačbo (2.16). Rešujemo problem

$$\begin{aligned} (\lambda+\mu)\nabla(\nabla\cdot\vec{u}) + \mu\nabla^2\vec{u} &= 0 \quad \text{na } \Omega = (-\infty,\infty)\times(-\infty,0) \\ \vec{t}(x,0) &= p(x)\vec{\jmath}, \\ \lim_{x,y\to\infty} \vec{u}(x,y) &= 0. \end{aligned} \tag{5.3}$$

Analitične rešitve kontaktnih problemov se ponavadi izpelje iz Flamantove rešitve, ki reši problem točkovnega pritiska na polravnino za robni pogoj $\vec{t}(x,0) = p_0\delta(x)\vec{\jmath}$. Druge rešitve lahko dobimo z konvolucijo s Flamantovo rešitvijo ali pa z metodo kompleksnih potencialov, kot je to narejeno v [33] še za malce splošnejši problem, kot opisano zgoraj. Od tam dobimo tudi analitično rešitev v zaprti obliki za napetost,

ki se izraža v splošni točki (x, y) s funkcijama m in n,

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

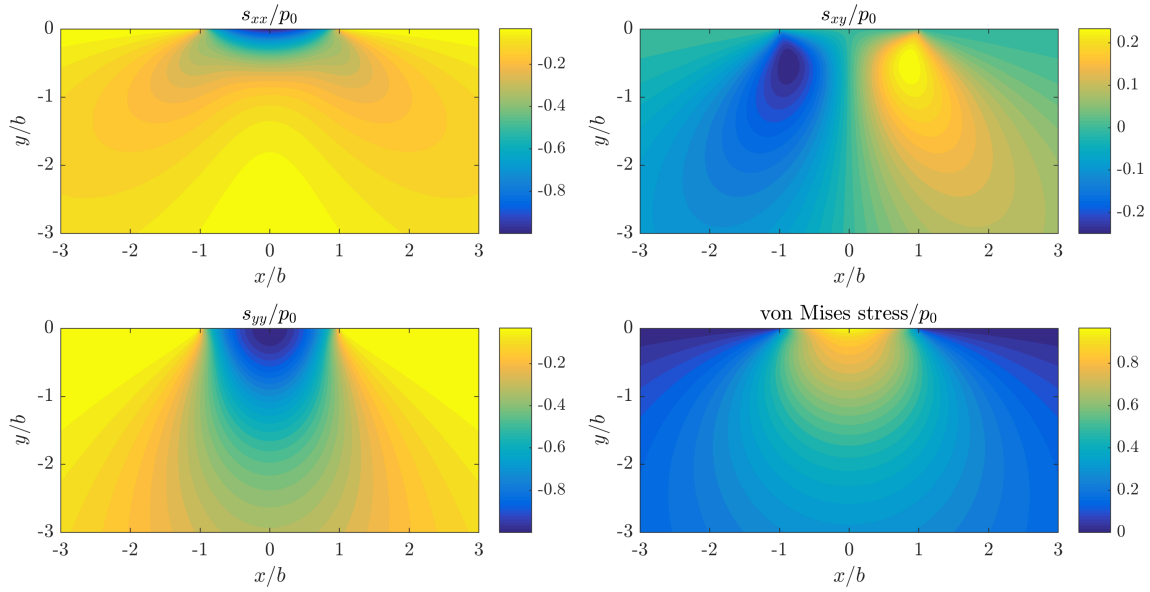
kjer je $m = \sqrt{m^2}$ in $n = \operatorname{sgn}(x)\sqrt{n^2}$.

$$\sigma_{xx} = -\frac{p_0}{b}\left[m\left(1 + \frac{y^2 + n^2}{m^2 + n^2}\right) + 2y\right]$$
$$\sigma_{yy} = -\frac{p_0}{b}m\left(1 - \frac{y^2 + n^2}{m^2 + n^2}\right)$$
$$\sigma_{xy} = \sigma_{yx} = \frac{p_0}{b}n\left(\frac{m^2 - y^2}{m^2 + n^2}\right).$$

S pomočjo zgornje rešitve bomo analizirali napako numerične rešitve. Poleg tega bomo uporabljali tudi von Misesov stress

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

ki se uporablja v von Misesovem kriteriju plastičnosti. Ta namreč pravi, da material ni več elastičen, ko σ_v preseže neko materialno konstanto σ_0. Analitična rešitev v okolici kontakta je prikazana na sliki 17.



Slika 17: Napetosti pod območjem kontakta med valjem in polravnino.

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 16b. Na zgornjem robu domene kot prej zahtevamo predpisano napetost. Za parametre problema smo vzeli $F = 543\,\mathrm{N/m}$, $E_1 = E_2 = 72.1\,\mathrm{GPa}$, $\nu_1 = \nu_2 = 0.33$, $R_1 = R = 1\,\mathrm{m}$.

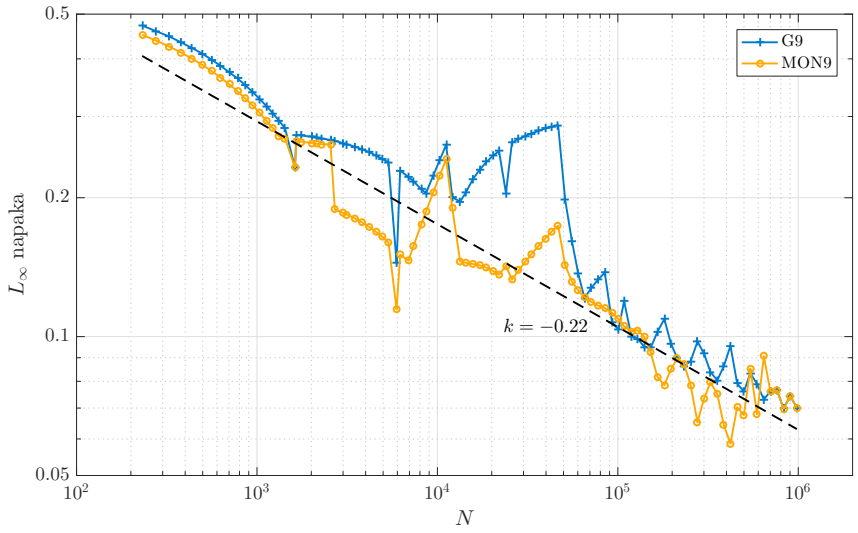
Od tod dobimo širino kontakta $b = 0.13\,\mathrm{mm}$ in maksimalni tlak $p_0 = 2.6\,\mathrm{MPa}$. Za H izberimo 10 mm, torej približno 75-krat večjo domeno, kot je pojav, ki ga opazujemo.

Numerično smo problem rešili na dva načina, z uporabo 9 monomov in 9 Gaussovih funkcij z $\sigma_b = 350\, r_\chi$. Obakrat smo uporabili Gaussovo utež z $\sigma = r_\chi$. Za reševanje linearnega sistema enačb smo uporabili iterativni BiCGSTAB algoritem z ILUT predpogojevanjem, kot opisano v razdelku 3.3.3. Uporabili smo parametra $p = 20$ in $\tau = 10^{-5}$. BiCGSTAB algoritem smo iterirali največ 300-krat ali dokler ni bila ocena napake pod 10^{-13}. Algoritem je v vseh primerih konvergiral. Omeniti je treba še, kako iz rešitve $\boldsymbol{u}$ 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 $\frac{1}{2}(\mathrm{grad}\,\vec{u} + \mathrm{grad}\,\vec{u}^{\mathsf{T}})$ ter nato preko Hookovega zakona dobimo σ.

Za normo napake si zoper 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 tako vzamemo

$$e_\infty = \max_{x \in X}\{\max\{\sigma_{xx}(x), \sigma_{yy}(x), \sigma_{xy}(x)\}\}/p_0.$$

Na sliki 18 je prikazana konvergenca numerične metode.

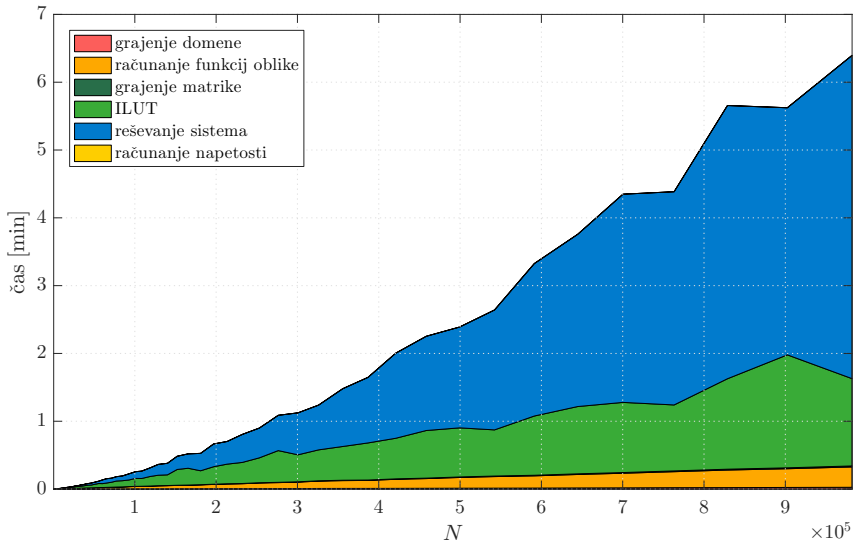


Slika 18: Konvergenca metode pri reševanju problema (5.3).

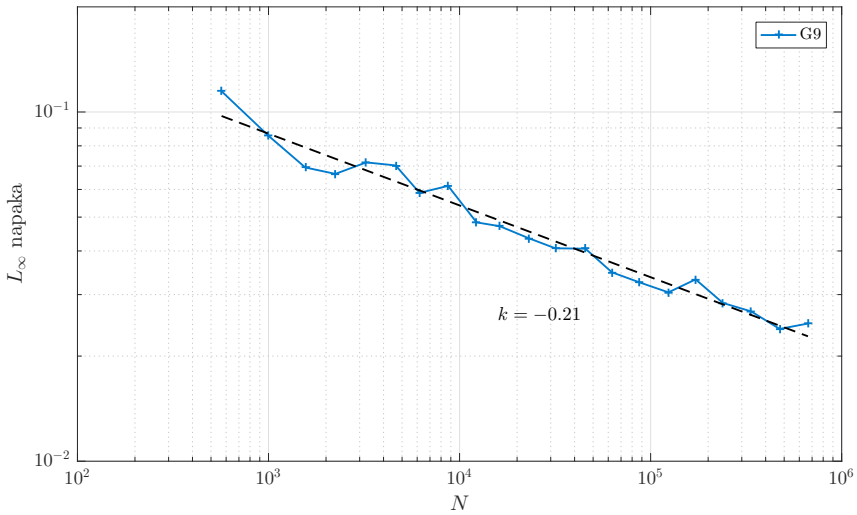
Na sliki 19 je prikazana razdelitev časa porabljenega za reševanje priblema. Časi za monome in Gaussove funkcije so po pričakovanju zelo primerljivi.

6 FWO case

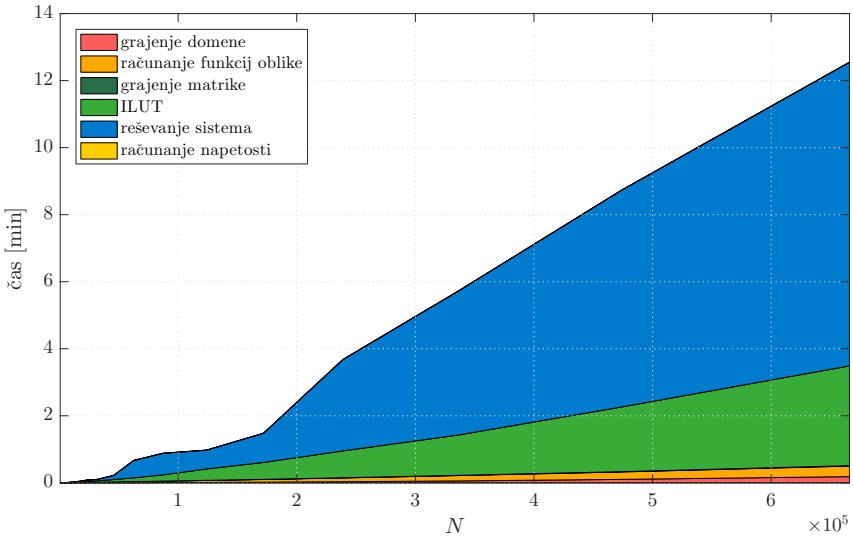
7 Zaključek



Slika 19: Časi posameznih kosov pri pri reševanju problema (5.3).



Slika 20: Konvergenca metode pri zgoščeni mreži.



Slika 21: Časi posameznih kosov pri zgoščeni mreži.

Literatura

[1] Robert A Adams and John JF Fournier. *Sobolev spaces*, volume 140. Academic press, 2003.

[2] L.P. Lebedev and M.J. Cloud. *Introduction to Mathematical Elasticity*. World Scientific, 2009.

[3] Philippe G Ciarlet. On korn's inequality. *Chinese Annals of Mathematics-Series B*, 31(5):607–618, 2010.

[4] Erwin Kreyszig. *Introductory functional analysis with applications*, volume 1. wiley New York, 1989.

[5] William S Slaughter. *The linearized theory of elasticity*. Springer Science & Business Media, 2012.

[6] Keith D Hjelmstad. *Fundamentals of structural mechanics*. Springer Science & Business Media, 2007.

[7] Morton E Gurtin. *An introduction to continuum mechanics*, volume 158. Academic press, 1982.

[8] Elliot A Kearsley and JT Fong. Linearly independent sets of isotropic Cartesian tensors of ranks up to eight. *J. Res. Natl Bureau of Standards Part B: Math. Sci. B*, 79:49–58, 1975.

[9] N. J. A. Sloane. The on-line encyclopedia of integer sequences. Sequence A005043. `http://oeis.org/A005043`, [obiskano 9. 7. 2016].

[10] Roderic Lakes. Foam structures with a negative poisson's ratio. *Science*, 235:1038–1041, 1987.

[11] Jernej Kozak. *Numerična analiza*. DMFA-založništvo, 2008.

[12] JH Hannay and JF Nye. Fibonacci numerical integration on a sphere. *Journal of Physics A: Mathematical and General*, 37(48):11591, 2004.

[13] Álvaro González. Measurement of areas on a sphere using Fibonacci and latitude–longitude lattices. *Mathematical Geosciences*, 42(1):49–64, 2010.

[14] William J Morokoff and Russel E Caflisch. Quasi-random sequences and their discrepancies. *SIAM Journal on Scientific Computing*, 15(6):1251–1279, 1994.

[15] Andrew W Moore. An intoductory tutorial on kd-trees. 1991.

[16] Stephen M Omohundro. *Five balltree construction algorithms*. International Computer Science Institute Berkeley, 1989.

[17] Alina Beygelzimer, Sham Kakade, and John Langford. Cover trees for nearest neighbor. In *Proceedings of the 23rd international conference on Machine learning*, pages 97–104. ACM, 2006.

[18] Ashraf M Kibriya and Eibe Frank. An empirical comparison of exact nearest neighbour algorithms. In *European Conference on Principles of Data Mining and Knowledge Discovery*, pages 140–151. Springer, 2007.

[19] David M Mount and Sunil Arya. ANN: library for approximate nearest neighbour searching. 1998. `https://www.cs.umd.edu/~mount/ANN/` [obiskano 16. 6. 2017].

[20] Timothy A Davis. *Direct methods for sparse linear systems*. SIAM, 2006.

[21] Yousef Saad. *Iterative methods for sparse linear systems*. SIAM, 2003.

[22] Xiaoye S Li. An overview of SuperLU: Algorithms, implementation, and user interface. *ACM Transactions on Mathematical Software (TOMS)*, 31(3):302–325, 2005.

[23] Gaël Guennebaud, Benoît Jacob, et al. Eigen v3, 2010. `http://eigen.tuxfamily.org` [obiskano 16. 6. 2017].

[24] Henk 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(2):631–644, 1992.

[25] Yousef Saad. ILUT: A dual threshold incomplete LU factorization. *Numerical linear algebra with applications*, 1(4):387–402, 1994.

[26] Robert Schaback. Error estimates and condition numbers for radial basis function interpolation. *Advances in Computational Mathematics*, 3(3):251–264, 1995.

[27] Robert Schaback. On the efficiency of interpolation by radial basis functions. 1997.

[28] Martin D Buhmann. Radial basis functions. *Acta Numerica 2000*, 9:1–38, 2000.

[29] Bjarne Stroustrup. *The C++ programming language*. Pearson Education India, 1995.

[30] Leonardo Dagum and Ramesh Menon. OpenMP: an industry standard API for shared-memory programming. *IEEE computational science and engineering*, 5(1):46–55, 1998. `http://www.openmp.org/` [obiskano 21. 06. 2017].

[31] Heinrich Hertz. Über die berührung fester elastischer körper. *Journal für die reine und angewandte Mathematik*, 92:156–171, 1882.

[32] John A Williams and Rob S Dwyer-Joyce. Contact between solid surfaces. *Modern tribology handbook*, 1:121–162, 2001.

[33] Ewen 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(303):454–459, 1949.