

LATVIJAS UNIVERSITĀTE  
FIZIKAS UN MATEMĀTIKAS FAKULTĀTE  
FIZIKAS NODAĻA

**Līdzsvara figūras un dinamika magnētiskā šķidruma pilienam  
ārējā laukā**

MAGISTRA DARBS

Autors:

Jānis Erdmanis

Studenta apliecības Nr.: je11011

Darba vadītājs:

profesors Andrejs Cēbers

RĪGA – 2016

# Anotācija

Līdzvara figūras rotējošām pašgravitējošiem šķidruma pilieniem ir pētītas vairāk nekā 100 gadus, bet tikai nesen tās iegūtas eksperimentāli pilienam veidotam no magnētiskā šķidruma. Iepriekš kvalitatīvi modeļi ir izmantoti to aprakstam, kurus šeit pārbaudu ar skaitlisku simulāciju. Šajā darbā izmantoju 3D robežintegrālvienādojumu formulējumu, ar kuru magnētisko lauku uz virsmas aprēķinu ar jaunu robeželementu metodes algoritmu un kurā izvairoš no tieša liekuma reķināšanas, izmantojot integrālās transformācijas. Skaitliskais algoritms ir testēts magnētiskā lauka aprēķinam elipsoīdā, izstiepta piliena relaksācijai, pielienu izstiepšanai konstantā laukā un tā saspiešanai rotējošā augstfrekvences laukā, kur analītiski rezultāti ir pieejami vai ir izstrādāti. Skaitlisko simulāciju izmantoju kvazi-līdzvara figūru iegūšanai, kas dod skaitlisku novērtējumu, kad rotējoša augstfrekvences lauka tuvinājums ir izmantojams. Ar rotējošu augstfrekvences lauku spontānā simetrijas laušana ir novērota pārejai no pankūkas uz plakantārpā tipa figūru, kas ir salīdzināts ar pusanalītisku modeli elipsoīdam. Skaitliskā algoritma efektivitāte ir ilustrēta simulējot spontāno simetrijas laušanu no pankūkas uz jūraszvaigznes nestabilitāti.

Atslēgas vārdi: līdzvara figūras, REM, magnētiskā šķidruma piliens, simulācija, spontānā simetrijas laušana

# Abstract

Equilibrium figures of rotating self-gravitating liquid drops have been under investigation for more than 100 years, but only recently they have been observed experimentally for a drop made of magnetic liquid. Some qualitative models have been used previously which are here under numerical investigation. A 3D boundary integral formulation here is used with which magnetic field on the surface is calculated with a novel boundary element algorithm and explicit curvature calculation is avoided with integral transformations. The numerical algorithm is tested here for a magnetic field in an ellipsoid, elongated drop relaxation dynamics, elongation in constant external field and compression in infinitely fast rotating field, where analytical solutions are available or are developed. The numerical simulation is used to obtain quasi-equilibrium figures in a slow rotating field giving an estimate for frequency when infinitely fast rotating field approximation is valid. With infinitely fast rotating field the spontaneous symmetry breaking is observed from pancake to a worm like figure, which is compared to a semi-analytical model for an ellipsoid. The efficiency of a numerical algorithm is illustrated by simulation of spontaneous symmetry breaking from pancake to a starfish instability.

Keywords: equilibrium figures, BEM, magnetic liquid drop, simulation, spontaneous symmetry breaking

# Saturs

<b>1 Ievads</b>	<b>1</b>
<b>2 Matemātiskais modelis un integrālvienādojumu formulējums</b>	<b>4</b>
2.1 Diferenciālais formulējums . . . . .	4
2.2 Spriegumu robežnosacījums . . . . .	6
2.3 Robežintegrālvienādojumu formulējums virsmas ātrumam . . . . .	9
2.4 Magnētiskā lauka aprēķināšana pie virsmas . . . . .	11
2.5 Īss kopsavilkums . . . . .	14
<b>3 Virsmas ģeometrijas triangulešana</b>	<b>16</b>
3.1 Virsmas triangulēšana . . . . .	16
3.2 Virsmas normāļu un liekuma aprēķināšana . . . . .	17
3.3 Virsmas ātruma sadalījuma integrēšana . . . . .	18
3.3.1 Virsmas ātrumu integrēšana . . . . .	19
3.3.2 Pasīvā stabilizācija . . . . .	19
3.3.3 Aktīvā stabilizācija . . . . .	20
<b>4 Magnētiskā lauka aprēķināšana</b>	<b>24</b>
4.1 Robežintegrālvienādojumu diskretizācija . . . . .	24
4.2 Potenciāla aprēķināšana . . . . .	26
4.2.1 Robežintegrālvienādojuma regularizācija . . . . .	26
4.2.2 Regularizētā robežintegrālvienādojuma risināšana ar liektiem elementiem . . . . .	27

4.2.3	Singulāra robežintegrālvienādojuma risināšana ar adaptīvo integrēšanu . . . . .	28
4.2.4	Potenciāla rēķināšanas algoritmu testi . . . . .	29
4.3	Lauka normālās komponentes aprēķināšana . . . . .	32
4.3.1	Singulārais formulējums . . . . .	32
4.3.2	Vāji singulārs formulējums . . . . .	33
4.3.3	Pārrēķinot no lauka tangenciālās komponentes . . . . .	34
4.3.4	Lauka normālās komponentes aprēķināšanas algoritmu salīdzinājums . . . . .	35
4.4	Lauka normālās un tangenciālās komponentes aprēķināšanas algoritms	36
<b>5</b>	<b>Virsmas ātruma aprēķināšana ar robežintegrālvienādojumu metodi</b>	<b>39</b>
5.1	Stoksa robežintegrālvienādojuma regularizācija un pārveidošana bezliekumu formā . . . . .	39
5.2	Relaksācija par sfēru . . . . .	41
5.3	Līdzvara figūras konstantā laukā . . . . .	43
<b>6</b>	<b>Simulācijas ar rotējošu lauku</b>	<b>46</b>
6.1	Līdzvara figūru atkarība no frekvences . . . . .	46
6.2	Rotējošs augstfrekvences lauks . . . . .	47
6.2.1	Skaitliska metode augstfrekvences lauka simulācijām . . . . .	47
6.2.2	Skaitliskas simulācijas augstfrekvences laukā . . . . .	48
6.2.3	Enerģijas minimums aksiālsimetriskam elipsoīdam . . . . .	49
6.2.4	Enerģijas minimums trīsasu elipsoīdam . . . . .	54
<b>7</b>	<b>Secinājumi</b>	<b>56</b>
<b>A</b>	<b>Integrālvienādojumu teorēmas</b>	<b>58</b>
A.1	Diverģence no normāles vektora . . . . .	58
A.2	Stoksa vienādojuma integrālā forma . . . . .	59
A.3	Robežintegrālvienādojums potenciālam . . . . .	59
A.4	Robežintegrālvienādojums magnētiskajam laukam . . . . .	61

A.5	Atbrīvošanās no virsmas liekuma . . . . .	62
A.6	Stoksa robežintegrālvienādojuma regularizācijas formula . . . . .	64
A.7	Potenciāla robežintegrālvienādojuma regularizācijas formula . . . . .	64
A.8	Magnētiskā lauka robežintegrālvienādojuma regularizācija . . . . .	65
<b>B</b>	<b>Vektorpotenciāla integrālvienādojuma formulējums</b>	<b>66</b>
<b>C</b>	<b>Pasīvās stabilizācijas enerģijas funkcijas minimizācija</b>	<b>68</b>
<b>D</b>	<b>Koda izvilkumi</b>	<b>71</b>
D.1	Potenciāla aprēķināšana . . . . .	71
D.2	Tangenciālā lauka komponenšu atrašana ar galīgajām diferencēm . .	72
D.3	Lauka normālās komponentes aprēķināšana . . . . .	73
D.4	Virsmas ātruma aprēķināšana . . . . .	75
D.5	Piemērs paralēlai implementācijai ar julia . . . . .	76
<b>Pateicības</b>		<b>78</b>
<b>Literatūra</b>		<b>79</b>

# Attēlu saraksts

1-1	Magnētiskā šķidruma piliena nestabilitātes plakantārpa formā augstfrekvences laukā. . . . .	1
1-2	Magnētiskā šķidruma piliena zvaigznes tipa nestabilitātes augstfrekvences laukā . . . . .	2
2-1	Magnētiskā šķidruma piliena matemātiskais modelis. . . . .	4
2-2	Spēki tilpuma elementam, kas šķērso piliena virsmu. . . . .	7
2-3	Lauks un magnetizācija tilpuma elementam, kas šķērso piliena virsmu.	8
3-1	CGAL un Distmesh ģenerēto triangulāciju salīdzinājums. . . . .	17
3-2	Pasīvās stabilizācijas algoritma tests ar Enright ātruma sadalījumu. .	21
3-3	Enright tests apvienotam pasīvās un aktīvās stabilizācijas algoritmam.	22
3-4	Enright testa apvienotam pasīvās un aktīvās stabilizācijas algoritmam skats no sāna. . . . .	23
4-1	Liekta kvadrātiska elementa raksturojošie parametri. . . . .	27
4-2	Adaptīvās kvadratūras shēma, kad veiktas 2 dalīšanas iterācijas. . .	29
4-3	Lauka normālās un tangenciālās komponentes relatīvā klūda (%) katrā triangulācijas virsotnē apvienotajā algoritmā. . . . .	38
5-1	Ļoti izstiepta piliena relaksācijas simulācija. . . . .	42
5-2	Maz izstiepta elipsoīda deformētības atkarība no laika relaksējot. .	42
5-3	Piemērs elipsoīda relaksācijai ar ārēju uzliktu lauku pie $\mu = 10$ . . .	44
5-4	Pusasu attiecības atkarība no Bonda skaitļa. . . . .	45

6-1	Magnētiskā šķidruma piliena formas atkarība no frekvences. . . . .	47
6-2	Magnētiskā šķidruma piliena dinamika augstfrekvences laukā. . . . .	50
6-3	Magnētiskā šķidruma piliena līdzvara figūra augstfrekvences laukā pie Bonda skaitļa 22.3. . . . .	50
6-4	Magnētiskā šķidruma piliena līdzvara figūra augstfrekvences laukā pie Bonda skaitļa 72. . . . .	51
6-5	Magnētiskā šķidruma piliena zvaigžņu simulācijas augstfrekvences laukā.	51
6-6	Piliena enerģijas atkarība saspieduma . . . . .	52
6-7	Minimālās enerģijas pusasu attiecības atkarība no magnētiskā Bonda skaitļa. . . . .	53
6-8	Magnētiskā šķidruma piliena skaitliskās simulācijas rotācijas figūras pusasu atkarība no Bonda skaitļa. . . . .	53
6-9	Magnētiskā šķidruma piliena skaitlisko simulāciju pusasu attiecības lauka rotācijas plaknē atkarība no Bonda skaitļa. . . . .	54
6-10	Magnētiskā šķidruma piliena skaitlisko simulāciju pusasu attiecības perpendikulāri lauka rotācijas plaknei atkarība no magnētiskā Bonda skaitļa. . . . .	55

# Tabulu saraksts

4.1	Potenciāla rēķināšanas algoritmu salīdzinājums. . . . .	31
4.2	Lauka normālās komponentes rēķināšanas algoritmu salīdzinājums . .	36
4.3	Lauka normālās un tangenciālās komponentes aprēķināšana apvienotajā algoritmā. . . . .	37

# Apzīmējumu saraksts

$\int \dots dS_x$	integrālis pa patvalīgu noslēgtu virsmu
$\int_{\partial D}$	integrālis pa piliena virsmu
$\mathcal{P} \int \dots dS_x$	Košī tipa pa patvalīgu noslēgtu virsmu
$\dots$ vai $\dots^{(i)}$	parametra vērtība tuvojoties virsmai no piliena iekšpuses
$\dots^{(e)}$	parametra vērtība tuvojoties virsmai no piliena ārpuses
$\eta$	šķidruma viskozitāte
$\mu$	relatīvā magnētiskā caurlaidība
$n_x$	normāles vektors punktā $x$
$M$	piliena magnetizācija
$H$	magnētiskā lauka intensitāte
$B$	magnētiskā lauka indukcija
$\gamma$	virsmas spraigums
$R_0$	piliena rādiuss relaksētā stāvoklī
$\sigma$	virsmas sprieguma tensors
$u$	šķidruma ātruma sadalījums
$\psi$	magnētiskais potenciāls ( $H = \nabla \psi$ )
$H_0$	magnētiskā lauka intensitāte, ja piliena nebūtu. Arī lauka intensitāte tālu no piliena, ja ārējais lauks ir homogēns
$r$	rādiusvektors $x - x_0$
$\{x\}$	visu virsmas punktu kopa

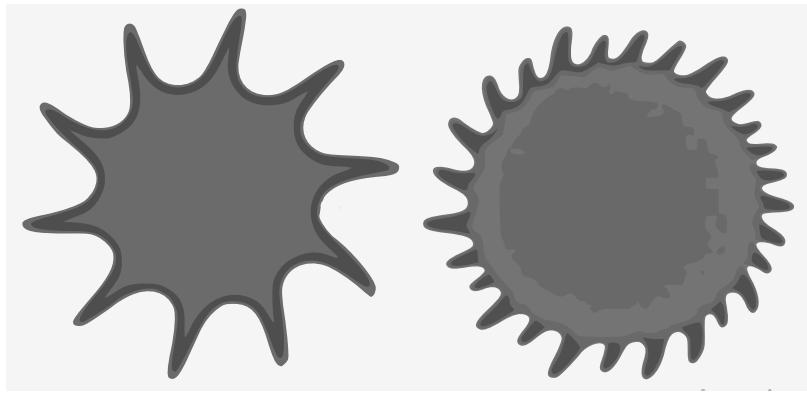
# Nodaļa 1

## Ievads

Līdzsvara figūras rotējošiem pilieniem, kur savstarpēji sacenšas centrālās un pašgravitacijas vai centrālās un virsmas spraiguma spēki, ir pētītas vairāk nekā 100 gadus [1]. Tikai salīdzinoši nesen šai saimei pievienojās magnētiskā šķidruma piliens rotējošā un precesējošā magnētiskā laukā, kur sacenšas pašmagnetizācijas un virsmas spraiguma spēks, demonstrējot nestabilitātes plakantārpa un jūraszvaigznes formā [2], [3] (skatīt attēlus 1-1 un 1-2). To pētišana pagaidām aprobežojusies ar kvalitatīviem modeļiem, kuru sekas šajā darbā pārbaudu kvantitatīvi ar skaitliskas simulācijas palīdzību.



Att. 1-1: Magnētiskā šķidruma piliena nestabilitātes plakantārpa formā augstfrekvenčes laukā [3]. Pie neesoša lauka piliens ieņem sfērisku formu (augšējais kreisais stūris), palielinot lauku iegūst rotācijas elipsoīdu (augšā pa vidu), kas kļūst nestabils pie vēl lielāka lauka un ieņem trīsas elipsoīda jeb plakantārpa formu. Vēl lielāka lauka intensitāte izstiepj plakantārpu (attēls apakšā) un panāk, ka tā gali noliecās.



Att. 1-2: Magnētiskā šķidruma piliena zvaigznes tipa nestabilitātes augstfrekvences laukā [3]. Zvaigznes formas staru skaits palielinās līdz ar uzlikto lauka intensitāti.

Magnētiskais šķidrums ir feromagnētisku nanodaļiņu koloīds, ko izmanto blīvju radišanai, skaļruņu spoļu dzesēšanai un arī biomedicīnā, ko parasti iegūst ar izsēdināšanas metodi  $Fe^{2+}$  un  $Fe^{3+}$  daļiņām [4]. Šim homogēnajam šķidrumam var inducēt sadalīšanos divās fāzēs - vāji un stipri koncentrētā [3]. Koncentrētā fāze ir ar spēcīgu magnētisko uzņēmību  $\mu = 40 \dots 80$  un tai ir pārsteidzoši mazs virsmas spraigums, kas ļauj veikt ar to pilieniem eksperimentus vājā magnētiskā laukā. Šādam magnētiskā šķidruma pilienam, atrodoties ārējā konstantā laukā, notiek tā stiepšanās lauka virzienā [5], un spēj izsekot arī lēni rotējoša lauka virzienam (sinhronais režīms). Pie lielākām lauka rotācijas frekvencēm lauka virziens apsteidz piliena rotāciju un tas šādā laukā svārstās (asinhronais režīms), bet vidēji rotē lauka virzienā. Savukārt pie augstām ārējā lauka frekvencēm var novērot piliena formas tārpa un zvaigznes formā (skatīt attēlus 1-1 un 1-2), kuras palielinot frekvenci kļūst nekustīgas [2], [3].

Jūraszvaigznes tipa nestabilitātēm (attēls 1-2) iepriekš ir pētīta staru skaita atkarība no magnētiskā lauka intensitātes [2], kur arī piedāvā kvalitatīvu modeli to aprakstam. Savukārt plakantārpa tipa nestabilitātes (attēls 1-2) ar analītiskām metodēm ir pētītas Bakalaura darbā [6], kur parādīta histerēze un iegūtas kritikās vērtības, kad rotācijas elipsoīds pārtop par trīsasu elipsoīdu. Tomēr šajā darbā tiek izdarīts pieņems, ka piliens vienmēr ieņem elipsoīda formu. Tādēļ pilnvērtīgai līdzsvara figūru pētišanai ir jāveido 3D datorsimulācija patvalīgam magnētiskā šķidruma pilienam patvalīgā ārējā laukā, kas arī ļaus pētīt piliena dinamiku.

Iepriekš veiktie aprēķini ar dielektrisku un magnētisku pielienu parādījuši robežintegrālvienādojumu efektivitāti simulācijām 2D un aksiālsimetriskā 3D gadījumā [7, 8, 9, 10, 11, 12, 13, 14], kur turpretim 3D simulācijas risinot attiecīgos vienādojumus visā tilpumā [15], rada milzīgas kļūdas, tādēļ šajā darbā izmantoju robežintegrālvienādojumu metodi. Patvalīgas formas 3D pilienam ar robežintegrālvienādojumu metodi jau ir veiktas simulācijas viskozā šķidrumā bez ārēja lauka, kur iegūst labu sakritību ar eksperimentu pilienu savstarpējai mijiedarbībai [16], [17] un to sadalīšanai [18], [19]. Šos algoritmus tad magnētiskā šķidruma piliena simulācijām būtu jāpapildina ar magnētiskā lauka aprēķinu uz virsmas<sup>1</sup>, kas ļauj aprēķināt magnetizācijas radīto spēku.

Lai magnētisko lauku uz virsmas atrastu, varētu risināt analogus [7], [8], [9], [10] integrālvienādojumus 3D gadījumam, tomēr literatūrā nav atrodama to pielietošana un šajā darbā arī redzēsim, ka šī pieeja 3D dod samērā lielu kļūdu. Citāda pieeja ir atrast potenciālu risinot attiecīgo robežintegrālvienādojumu, no kura pēc galīgo diferenču metodes atrod lauku tangenciālas komponentes. Lai atrastu lauka normālo komponenti, varētu risināt pirmā veida integrālvienādojumu, kur kā ievadi dotu potenciālu tomēr testi parādījuši, ka arī šī metode nedod pietiekami precīzu rezultātu. Tādēļ šajā darbā ir izstrādāts principiāli atšķirīgs algoritms lauka normālās komponentes atrašanai, ar ko var no lauka tangenciālās komponentes pārrēķināt lauka normālo komponenti.

---

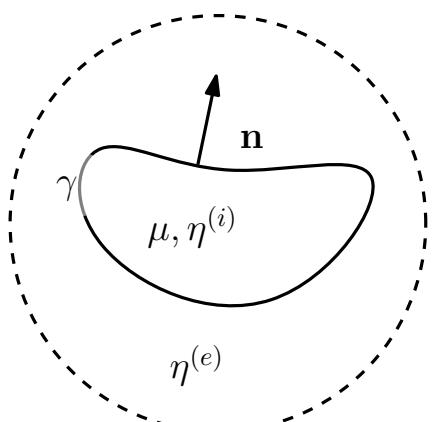
<sup>1</sup>Tuvojoties virsmai no piliena iekšpuses.

## Nodaļa 2

# Matemātiskais modelis un integrālvienādojumu formulējums

Šajā nodaļā formulēsim matemātikas fizikas problēmu magnētiskā šķidruma pilienam ārējā laukā pieņemot, ka magnētiskais materiāls ir lineārs un šķidrums ir viskozs. Formulēsim vienādojumus ar robežnosacījumiem diferenciālā formā, no kuriem iegūsim integrālo formu, gan magnētiskā lauka, gan virsmas ātruma atrašanai.

### 2.1 Diferenciālais formulējums



$$\mathbf{H}(\mathbf{x} \rightarrow \infty) = \mathbf{H}_0$$

Att. 2-1: Magnētiskā šķidruma piliena matemātiskais modelis.

Magnētiskā šķidruma piliens ir pēc izmēriem mazs  $R_0 = 15\mu m$ , viskozitāte pilienam un apkārtējai videi ir attiecīgi  $\eta^{(i)} = 100mPa \cdot s$  un  $\eta^{(e)} = 1mPa \cdot s$ , piliena virsmas spraugums ir  $\gamma = 2.8 \cdot 10^{-6}J/m^2$  [2]. Šķidruma raksturīgo ātrumu var novērtēt no zvaigznes nestabilitātes staru rotācijas ātruma, kas kādā augstfrekvences mērījumā ir  $\omega_c = 4Hz$ , kas ļauj novērtēt Reinoldsa skaitli

$$Re = \frac{\rho R_0^2 \omega_c}{\eta^{(i)}} = 0.09 \quad (2.1)$$

tādēļ inerces efektus atmetam un apskatām viskoza šķidruma modeli gan pilienam gan apkārtējai videi.

Tā kā piliens ir magnetizējams, tad uz to papildus iedarbojās tilpuma spēks  $(\mathbf{M} \cdot \nabla) \mathbf{H}$ , kur  $\mathbf{M}$  ir magnetizācija un  $\mathbf{H}$  magnētiskā lauka intensitāte. Stoksa vienādojums pilienā tādēļ ir rakstāms šādi:

$$-\nabla p + \eta^{(i)} \Delta \mathbf{u} + (\mathbf{M} \cdot \nabla) \mathbf{H} = 0 \quad \nabla \cdot \mathbf{u} = 0 \quad (2.2)$$

kur ar  $\eta^{(i)}$  apzīmējam iekšējās vides viskozitāti. Ārējā vides magnetizācija ir 0 tādēļ tai atbilst ierastais Stoksa vienādojums

$$-\nabla p + \eta^{(e)} \Delta \mathbf{u} = 0 \quad \nabla \cdot \mathbf{u} = 0 \quad (2.3)$$

Magnētiskā lauka aprēķināšanai izmantojam magnetostatikas vienādojumus:

$$\nabla \cdot \mathbf{B} = 0 \quad \nabla \times \mathbf{H} = 0 \quad (2.4)$$

kurus varam pielietot, jo  $R_0^2 \omega^2 \ll c^2$ , kur  $R_0$  piliena raksturīgais izmērs. Dēļ tā ka starp apkārtējo vidi un pilienu ir straujš parametru lēciens, tad robežu izslēdzam no vienādojuma atrisinājuma ārā, ko panākam ar robežnosacījumiem uz piliena virsmas

un ar robežnosacījumu bezgalībā

$$\begin{aligned} B_n^{(i)} &= B_n^{(e)} & \mathbf{H}_t^{(i)} &= \mathbf{H}_t^{(e)} \\ \mathbf{H}(\mathbf{x} \rightarrow \infty) &= \mathbf{H}_0 & (2.5) \end{aligned}$$

Lai pabeigtu formulējumu matemātiskās fizikas problēmai, izmantojam, ka magnetizācija ir lineāri atkarīga no lauka intensitātes jeb  $\mathbf{B} = \mu \mathbf{H}$  [4].

Matemātiskās fizikas problēmu var arī noformulēt potenciālam  $\mathbf{H} = \nabla \psi$  no kā iegūstam, ka jārisina Laplaša vienādojums piliena iekšpusē un ārpusē<sup>1</sup>.

$$\Delta \psi = 0 \quad (2.6)$$

Tā robežnosacījumi attiecīgi izriet no (2.5), kā<sup>2</sup>

$$\begin{aligned} \psi^{(i)} &= \psi^{(e)} & \mu \frac{\partial \psi^{(i)}}{\partial n} &= \frac{\partial \psi^{(e)}}{\partial n} \\ \psi(\mathbf{x} \rightarrow \infty) &= \mathbf{H}_0 \cdot \mathbf{x} & (2.7) \end{aligned}$$

## 2.2 Spriegumu robežnosacījums

Apskatīsim tilpuma elementu, kas šķērso magnētiskā piliena robežu. No augšas un apakšas iedarbojās šķidruma sprieguma  $\boldsymbol{\sigma}$  radītais spēks jeb spēks uz laukuma vienību  $\boldsymbol{\sigma} \cdot \mathbf{n}$ . No elementa sāniem virsmas spraigums, ko raksturo spēks uz garuma vienību  $\gamma$  un vērstības binormāles virzienā  $\mathbf{b} = \mathbf{n} \times \mathbf{t}$ . Ārējā magnētiskā laukā papildus nāk arī tilpuma spēks  $(\mathbf{M} \cdot \nabla) \mathbf{H}$  un tādēļ spēka līdzsvara vienādojums tilpuma elementam dod

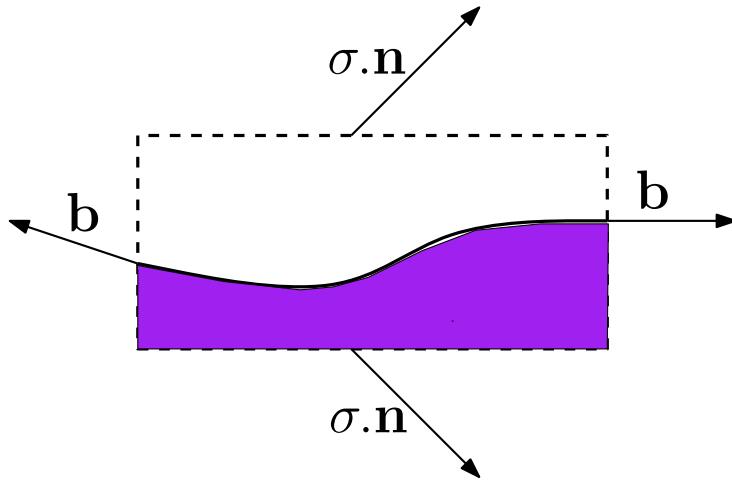
$$\int \boldsymbol{\sigma} \cdot \mathbf{n} dS + \int \gamma \mathbf{b} dl = \int (\mathbf{M} \cdot \nabla) \mathbf{H} dV \quad (2.8)$$

Virsmas spraiguma integrāli varam pārveidot izmantot sekojošu variāciju Stoksa

---

<sup>1</sup>Bet ne uz robežas.

<sup>2</sup>Pienemot, ka bezgalībā lauks ir homogēns, un izvēloties atskaites sistēmu potenciālam, kur  $\psi = 0$ , kad  $\mathbf{x} = 0$ .



Att. 2-2: Spēki tilpuma elementam, kas šķērso pilienu virsmu.

teorēmai

$$\int [\mathbf{F} \times \mathbf{t}]_k dl = \int (n_k \nabla \cdot \mathbf{F} - n_i \partial_k F_i) dS \quad (2.9)$$

Ievietojot  $\mathbf{F} = \gamma \mathbf{n}$  un izmantojot  $n_i \nabla n_i = \frac{1}{2} \nabla(\mathbf{n} \cdot \mathbf{n}) = 0^3$  iegūstam virsmas spraiguma integrālim sekojošu izteiksmi

$$\int \gamma \mathbf{b} dl = \int \gamma \mathbf{t} \times \mathbf{n} dl = - \int \gamma \mathbf{n} \nabla \cdot \mathbf{n} dS + \int (\mathbf{I} - \mathbf{n} \otimes \mathbf{n}) \cdot \nabla \gamma dS \quad (2.10)$$

Otrais integrālis ir atbildīgs par Marangoni efektu<sup>4</sup>, ko iespējams novērot, ja šķidrumā atrodas virsmas aktīvās vielas, kuru koncentrācija nav homogēna, un ja nehomogēns ir temperatūras sadalījums pilienā. Šajā fizikālajā modelī šos efektus neapskatām un pieņemam, ka  $\nabla \gamma = 0$ .

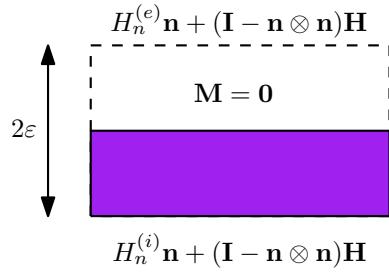
Lai iegūtu izteiksmi spēkam, ko rada magnetizācijas lēciens, apskatīsim mazu elementu, kas šķērso pilienu virsmu.

Elementa izmērs ir tāds, ka uz tā augšējās robežas (skatīt attēlu) magnētiskā lauka normālā komponente ir  $H_n^{(e)}$ , uz apakšējās robežas  $H_n^{(i)}$ , bet tangenciālā lauka komponente ir konstanta visā elementā un vienāda ar  $(\mathbf{I} - \mathbf{n} \otimes \mathbf{n}) \mathbf{H}$ . Tādēļ magnetizācijas

---

<sup>3</sup>Seit pieņemam, ka virsmu var uzdot ar līmenīvirsmu  $F(x, y, z) = 0$ , kurās gradiencts ir proporcionāls normālēs virzienam  $\mathbf{n} \propto \nabla F$ .

<sup>4</sup>Skaitliskas simulācijas pilieniem ar virsmas aktīvām vielām veiktas ar robežintegrālvienādoju-miem [20], kur koncentrācijas atrašanai risina difūzijas vienādojumu visā tilpumā.



Att. 2-3: Lauks un magnetizācija tilpuma elementam, kas šķērso piliena virsmu.

radīto spēku tilpuma vienībā varam rakstīt šādi

$$(\mathbf{M} \cdot \nabla)\mathbf{H} = (\mathbf{M}_t \cdot \nabla)\mathbf{H} + M_n \frac{\partial \mathbf{H}}{\partial n} = (\mathbf{M}_t \cdot \nabla)\mathbf{H} + M_n \frac{\partial H_n}{\partial n} \mathbf{n} \quad (2.11)$$

Tā kā \$(\mathbf{M}\_t \cdot \nabla)\mathbf{H}\$ ir ar galīgu lēcienu visā elementā, tad tās ieguldījums integrējot pa visu elementu ir 0. No kā seko, ka magnetizācijas radītais spēks uz tilpuma elementu, kas šķērso virsmu ir

$$\begin{aligned} \int (\mathbf{M} \cdot \nabla)\mathbf{H} dV &= \Delta S \mathbf{n} \int_{-\varepsilon}^{+\varepsilon} M_n \frac{\partial H_n}{\partial n} dn = \Delta S \mathbf{n} \int_{-\varepsilon}^{+\varepsilon} M_n \frac{H_n^{(e)} - H_n^{(i)}}{2\varepsilon} dn \\ &= \Delta S \mathbf{n} \frac{1}{2} M_n (H_n^{(i)} - H_n^{(e)}) = -2\pi M_n^2 \Delta S \mathbf{n} \end{aligned} \quad (2.12)$$

kur izmantojām \$M\_n(z) = M\_n \Theta(-z)\$, \$\mathbf{B}^{(i)} = \mathbf{H}^{(i)} + 4\pi \mathbf{M}\$ un \$\mathbf{B}^{(e)} = \mathbf{H}^{(e)}.

Lai iegūtu izteiksmes robežnosacījumiem, ievietojam virsmas spraiguma un magnētiskā spēka izteiksmes (2.10), (2.12) vienādojumā (2.8) un tiecinām \$\Delta S\$ uz \$0^5\$

$$\sigma_{nn}^{(e)} - \sigma_{nn}^{(i)} = \gamma \nabla \cdot \mathbf{n} - 2\pi M_n^2 \quad (2.13)$$

Savukārt tangenciālais spriegums ir nepārtrauks

$$\sigma_{tn}^{(e)} - \sigma_{tn}^{(i)} = 0 \quad (2.14)$$

---

<sup>5</sup>\$\frac{1}{2}\nabla \cdot \mathbf{n}\$ ir vienāds arī ar virsmas vidējo liekumu, kam izvedumu var atrast § A.1

## 2.3 Robežintegrālvienādojumu formulējums virsmas ātrumam

Lai iegūtu robežintegrālvienādojumus virsmas ātruma atrašanai, izmantosim Stoksa robežintegrālvienādojuma formulējumu gan piliena iekšpusē, gan ārpuse. Tā izmantošanai vienādojumā (2.2) ievietojam  $\mathbf{M} = (\mu - 1)\mathbf{H}/4\pi$  un to pārrakstām formā

$$-\nabla \left( p - \frac{\mu - 1}{8\pi} \mathbf{H}^2 \right) + \eta^{(i)} \Delta \mathbf{u} = 0 \quad \nabla \cdot \mathbf{u} = 0 \quad (2.15)$$

Tā kā  $\sigma_{ij} = -p\delta_{ij} + \eta(\partial_i u_k + \partial_k u_i)$ , jo Stoksa vienādojums rakstāms arī kā  $\nabla \cdot \sigma = 0$ , tad varam augstāko vienādojumu reducēt par Stoksa vienādojumu sprieguma tenzoram, kas uzrakstīts šādi

$$\sigma_{ij}^{(i)} \rightarrow - \left( p - \frac{\mu - 1}{8\pi} \mathbf{H}^2 \right) \delta_{ij} + \eta(\partial_i u_k + \partial_k u_i) = \sigma_{ij}^{(i)} + \frac{\mu - 1}{8\pi} \mathbf{H}^2 \delta_{ij} \quad (2.16)$$

Tādēļ visā telpas apgabalā (izņemot virsmu) spēkā Stoksa vienādojums

$$-\nabla p + \eta \Delta \mathbf{u} = 0 \quad \nabla \cdot \mathbf{u} = 0 \quad (2.17)$$

Kam atbilstošais robežnosacījums uz piliena virsmas pēc vienādojumiem (2.13), (2.14) un (2.16) ir<sup>6</sup>

$$\begin{aligned} \sigma_{nn}^{(e)} - \sigma_{nn}^{(i)} &= \gamma \nabla \cdot \mathbf{n} - 2\pi M_n^2 - \frac{\mu - 1}{8\pi} \mathbf{H}^2 = \gamma \nabla \cdot \mathbf{n} - \frac{\mu(\mu - 1)}{8\pi} H_n^2 - \frac{\mu - 1}{8\pi} H_t^2 \\ \sigma_{tn}^{(e)} - \sigma_{tn}^{(i)} &= 0 \end{aligned} \quad (2.18)$$

kur  $H_n$  un  $H_t$  ir magnētiskā lauka intensitātes normālā un tangenciālā komponente iekšpus pilienam pie tā virsmas.<sup>7</sup>

---

<sup>6</sup>Spriegumam piliena iekšienē ir jauna definīcija (2.16)!

<sup>7</sup>Turpmāk atmetam augšējo indeksu lauka intensitātei.

Stoksa vienādojuma (2.17) integrālā forma ir (skatīt § A.2):

$$C(\mathbf{x}_0)u_j(\mathbf{x}_0) = -\frac{1}{\eta} \int G_{ji}(\mathbf{x}, \mathbf{x}_0)\sigma_{ik}n_{\mathbf{x}k}dS_{\mathbf{x}} + \int u_i(\mathbf{x})T_{ijk}(\mathbf{x}, \mathbf{x}_0)n_{\mathbf{x}k}dS_{\mathbf{x}} \quad (2.19)$$

$$G_{ij}(\mathbf{x}, \mathbf{x}_0) = \frac{\delta_{ij}}{r} + \frac{r_i r_j}{r^3} \quad T_{ijk}(\mathbf{x}, \mathbf{x}_0) = -6 \frac{r_i r_j r_k}{r^5} \quad r = \mathbf{x} - \mathbf{x}_0 \quad (2.20)$$

kur  $C(\mathbf{x}_0)$  ir  $8\pi$ , ja  $\mathbf{x}_0$  atrodas iekšpusē,  $4\pi$  uz virsmas un 0, ja ārpus noslēgtā apgabala. Izmantojot to izvedīsim robežintegrālvienādojumu piliena virsmas ātrumam.

Apzīmēsim ar  $\partial D$  piliena virsmu, bet ar  $\partial\infty$  pilienu iekļaujošu virsmu, kas atrodas tālu no tā. Normāles vektoru izvēlamies vērstu no piliena uz ārpusi (skatīt attēlu 2-1), tad augstāk esošais vienādojums pilienam

$$4\pi u_j(\mathbf{x}_0) = -\frac{1}{\eta^{(i)}} \int_{\partial D} G_{ji}(\mathbf{x}, \mathbf{x}_0)\sigma_{ik}^{(i)}n_{\mathbf{x}k}dS_{\mathbf{x}} + \mathcal{P} \int_{\partial D} u_i(\mathbf{x})T_{ijk}(\mathbf{x}, \mathbf{x}_0)n_{\mathbf{x}k}dS_{\mathbf{x}} \quad (2.21)$$

un apkārtējai videi

$$\begin{aligned} 4\pi u_j(\mathbf{x}_0) &= +\frac{1}{\eta^{(e)}} \int_{\partial D} G_{ji}(\mathbf{x}, \mathbf{x}_0)\sigma_{ik}^{(e)}n_{\mathbf{x}k}dS_{\mathbf{x}} - \mathcal{P} \int_{\partial D} u_i(\mathbf{x})T_{ijk}(\mathbf{x}, \mathbf{x}_0)n_{\mathbf{x}k}dS_{\mathbf{x}} + 8\pi u_{0j}(\mathbf{x}_0) \\ u_{0j}(\mathbf{x}_0) &= -\frac{1}{8\pi\eta^{(e)}} \int_{\partial\infty} G_{ji}(\mathbf{x}, \mathbf{x}_0)\sigma_{ik}^{(e)}n_{\mathbf{x}k}dS_{\mathbf{x}} + \frac{1}{8\pi} \int_{\partial\infty} u_i(\mathbf{x})T_{ijk}(\mathbf{x}, \mathbf{x}_0)n_{\mathbf{x}k}dS_{\mathbf{x}} \end{aligned} \quad (2.22)$$

Pareizinot ar  $\eta^{(i)}$  vienādojumu (2.21) un pieskaitot pie tā  $\eta^{(e)}$  reizes vienādojumu (2.22) iegūstam integrālvienādojumu piliena virsmas ātrumam

$$\begin{aligned} u_j(\mathbf{x}_0) &= \frac{2\eta^{(e)}}{\eta^{(i)} + \eta^{(e)}} u_{0j}(\mathbf{x}_0) - \frac{1}{4\pi(\eta^{(e)} + \eta^{(i)})} \int_{\partial D} G_{ji}(\mathbf{x}, \mathbf{x}_0)(\sigma_{ik}^{(i)} - \sigma_{ik}^{(e)})n_{\mathbf{x}k}dS_{\mathbf{x}} \\ &\quad + \frac{1}{4\pi} \frac{\eta^{(i)} - \eta^{(e)}}{\eta^{(i)} + \eta^{(e)}} \mathcal{P} \int_{\partial D} u_i(\mathbf{x})T_{ijk}(\mathbf{x}, \mathbf{x}_0)n_{\mathbf{x}k}dS_{\mathbf{x}} \end{aligned} \quad (2.23)$$

Ievietojot robežnosacījumus (2.18) un ņemot vērā, ka tālu no piliena šķidrums ir

nekustīgs jeb  $\mathbf{u}_0(\mathbf{x}) = 0$ , tad iegūstam

$$\begin{aligned} u_j(\mathbf{x}_0) &= \frac{1}{4\pi(\eta^{(e)} + \eta^{(i)})} \int_{\partial D} G_{ji}(\mathbf{x}, \mathbf{x}_0) n_{\mathbf{x}i} \left( \gamma \nabla_{\mathbf{x}} \cdot \mathbf{n}_{\mathbf{x}} - \frac{\mu(\mu-1)}{8\pi} H_n^2 - \frac{\mu-1}{8\pi} H_t^2 \right) dS_{\mathbf{x}} \\ &\quad + \frac{1}{4\pi} \frac{\eta^{(i)} - \eta^{(e)}}{\eta^{(i)} + \eta^{(e)}} \mathcal{P} \int_{\partial D} u_i(\mathbf{x}) T_{ijk}(\mathbf{x}, \mathbf{x}_0) n_{\mathbf{x}k} dS_{\mathbf{x}} \end{aligned} \quad (2.24)$$

Kaut gan eksperimentāli viskozitātes starp piliena iekšpusi un ārpusi atšķiras pat 100 reizes [2], integrālvienādojumā (2.24) redzam, ka līdzvara stāvoklis, kad  $\mathbf{u} = 0$ , nav no tās atkarīgs. Tādēļ līdzvaru figūru iegūšanai varam izmantot vienādu viskozitāšu attiecību, kas novērtē pie vienkāršas formulas virsmas ātruma aprēķināšanai

$$u_j(\mathbf{x}_0) = \frac{1}{8\pi\eta} \int_{\partial D} G_{ji}(\mathbf{x}, \mathbf{x}_0) n_{\mathbf{x}i} \left( \gamma \nabla_{\mathbf{x}} \cdot \mathbf{n}_{\mathbf{x}} - \frac{\mu(\mu-1)}{8\pi} H_n^2 - \frac{\mu-1}{8\pi} H_t^2 \right) dS_{\mathbf{x}} \quad (2.25)$$

## 2.4 Magnētiskā lauka aprēķināšana pie virsmas

Magnētiskam laukam ir iespējami dažādi robežintegrālvienādojumu formulējumi, kuri sīkāk apskatu un salīdzinu nodaļā § 4. Šajā sadaļā apskatīsim formulējumus magnētiskajam laukam pie virsmas, kuru salīdzinot ar citiem formulējumiem iegūst vismazāko klūdu, kad to implementē skaitliski, kas sīkāk iztirzāts § 4.2.4.

Laplaza vienādojuma integrālo formu iegūstam no Grīna formulas (skatīt § A.3)

$$C(\mathbf{x}_0)\psi(\mathbf{x}_0) = \int \frac{\partial \psi}{\partial n_{\mathbf{x}}} \frac{1}{r} dS_{\mathbf{x}} - \int \psi(\mathbf{x}) \frac{\partial}{\partial n_{\mathbf{x}}} \frac{1}{r} dS_{\mathbf{x}} \quad r = |\mathbf{x} - \mathbf{x}_0| \quad (2.26)$$

kur  $C$  ir  $4\pi$ , ja  $\mathbf{x}_0$  atrodas iekšpusē,  $2\pi$  uz virsmas un  $0$  ārpus apgabala. Kad  $\mathbf{x}_0$  atrodas uz virsmas, tad labajā pusē esošais integrālis ir jāinterpretē kā Košī integrālis  $\mathcal{P} \int$ . Šajā darbā to interpretēju, kā virsmas integrāli ar izgrieztu  $\varepsilon$  apkārtni ap singularitāti.

Izvēloties par kopīgo normāles virzienu, kas vērstīs prom no piliena apgabala, rak-

stām robežintegrālvienādojumu pilienam

$$2\pi\psi^{(i)}(\mathbf{x}_0) = \int_{\partial D} \frac{\partial\psi^{(i)}}{\partial n_{\mathbf{x}}} \frac{1}{r} dS_{\mathbf{x}} - \mathcal{P} \int_{\partial D} \psi^{(i)}(\mathbf{x}) \frac{\partial}{\partial n_{\mathbf{x}}} \frac{1}{r} dS_{\mathbf{x}} \quad (2.27)$$

un tā apkārtējam apgabalam

$$\begin{aligned} 2\pi\psi^{(e)}(\mathbf{x}_0) &= - \int_{\partial D} \frac{\partial\psi^{(e)}}{\partial n_{\mathbf{x}}} \frac{1}{r} dS_{\mathbf{x}} + \mathcal{P} \int_{\partial D} \psi^{(e)}(\mathbf{x}) \frac{\partial}{\partial n_{\mathbf{x}}} \frac{1}{r} dS_{\mathbf{x}} + 4\pi\psi_0(\mathbf{x}_0) \\ \psi_0(\mathbf{x}_0) &= \frac{1}{4\pi} \int_{\partial\infty} \frac{\partial\psi^{(e)}}{\partial n_{\mathbf{x}}} \frac{1}{r} dS_{\mathbf{x}} - \frac{1}{4\pi} \int_{\partial\infty} \psi^{(e)}(\mathbf{x}) \frac{\partial}{\partial n_{\mathbf{x}}} \frac{1}{r} dS_{\mathbf{x}} \end{aligned} \quad (2.28)$$

Ievietojot robežnosacījumus (2.7) un saskaitot vienādojumu (2.28) ar  $\mu$  reizes vienādojumu (2.27) iegūst sekojošu robežintegrālvienādojumu potenciālam

$$\psi(\mathbf{x}_0) = \frac{2\psi_0(\mathbf{x}_0)}{\mu+1} - \frac{1}{2\pi} \frac{\mu-1}{\mu+1} \mathcal{P} \int_{\partial D} \psi(\mathbf{x}) \frac{\partial}{\partial n_{\mathbf{x}}} \frac{1}{|\mathbf{x}-\mathbf{x}_0|} dS_{\mathbf{x}} \quad (2.29)$$

Tālu no piliena magnētiskais lauks ir homogēns, kas vienāds ar  $\mathbf{H}_0$ , tādēļ  $\psi_0(\mathbf{x}) = \mathbf{H}_0 \cdot \mathbf{x}$  un risināmais robežintegrālvienādojums

$$\psi(\mathbf{x}_0) = \frac{2\mathbf{H}_0 \cdot \mathbf{x}_0}{\mu+1} - \frac{1}{2\pi} \frac{\mu-1}{\mu+1} \mathcal{P} \int_{\partial D} \psi(\mathbf{x}) \frac{\partial}{\partial n_{\mathbf{x}}} \frac{1}{|\mathbf{x}-\mathbf{x}_0|} dS_{\mathbf{x}} \quad (2.30)$$

Magnētiskā lauka intensitātes tangenciālo komponenšu atrašanai pēc šī vienādojuma atrisināšanas izmantojam galīgās differences, kā tas izklāstīts § 4.4.

Magnētiskā lauka normālās komponentes atrašanai izmantojam šādu formulējumu (skatīt § A.4)

$$C(\mathbf{x}_0)\mathbf{B}(\mathbf{x}_0) = \int \mu \mathbf{J} \times \nabla_{\mathbf{x}} \frac{1}{r} dV - \int \left[ (\mathbf{n}_{\mathbf{x}} \times \mathbf{B}) \times \nabla_{\mathbf{x}} \frac{1}{r} + (\mathbf{n}_{\mathbf{x}} \cdot \mathbf{B}) \nabla_{\mathbf{x}} \frac{1}{r} \right] dS_{\mathbf{x}} \quad (2.31)$$

kur  $\mathbf{J}$  ir brīvās strāvas, kuras apskatītajā modelī nav, bet  $C(\mathbf{x}_0)$  ir  $4\pi$  apgabala iekšpusē,  $2\pi$  uz virsmas, 0 ārpus apgabala.

Izvēloties normāles virzienu vērstu no piliena uz ārpusi, var iegūt robežintegrāl-

vienādojumu pilienam

$$2\pi \mathbf{B}^{(i)}(\mathbf{x}_0) = -\mathcal{P} \int_{\partial D} \left[ (\mathbf{n}_x \times \mathbf{B}^{(i)}) \times \nabla_x \frac{1}{r} + (\mathbf{n}_x \cdot \mathbf{B}^{(i)}) \nabla_x \frac{1}{r} \right] dS_x \quad (2.32)$$

un apkārtējam apgabalam

$$\begin{aligned} 2\pi \mathbf{B}^{(e)}(\mathbf{x}_0) &= +\mathcal{P} \int_{\partial D} \left[ (\mathbf{n}_x \times \mathbf{B}^{(e)}) \times \nabla_x \frac{1}{r} + (\mathbf{n}_x \cdot \mathbf{B}^{(e)}) \nabla_x \frac{1}{r} \right] dS_x + 4\pi \mathbf{B}_0(\mathbf{x}_0) \\ \mathbf{B}_0(\mathbf{x}_0) &= -\frac{1}{4\pi} \int_{\partial \infty} \left[ (\mathbf{n}_x \times \mathbf{B}^{(e)}) \times \nabla_x \frac{1}{r} + (\mathbf{n}_x \cdot \mathbf{B}^{(e)}) \nabla_x \frac{1}{r} \right] dS_x \end{aligned} \quad (2.33)$$

Saskaitot abus vienādojumus un izmantojot, ka magnētiskā indukcijas vektora normālā komponente ir nepārtraukta iegūstam

$$2\pi \mathbf{B}^{(i)}(\mathbf{x}_0) + 2\pi \mathbf{B}^{(e)}(\mathbf{x}_0) = 4\pi \mathbf{B}_0(\mathbf{x}_0) + \mathcal{P} \int_{\partial D} \left[ \mathbf{n}_x \times (\mathbf{B}^{(e)} - \mathbf{B}^{(i)}) \right] \times \nabla_x \frac{1}{r} dS_x \quad (2.34)$$

Iegūtā formula parāda analogiju, ka magnētisko lauku uz virsmas varam aprēķināt kā ārējā lauka un virsmas strāvu  $4\pi \mathbf{K} = \mathbf{n} \times (\mathbf{B}^{(e)} - \mathbf{B}^{(i)})$  radītā lauka summu.

Projicējot šo vienādojumu normāles virzienā, un ievietojot robežnosacījumu (2.5), iegūstam

$$H_n(\mathbf{x}_0) = \frac{\mathbf{H}_0 \cdot \mathbf{n}_{\mathbf{x}_0}}{\mu} + \frac{1}{4\pi} \frac{1-\mu}{\mu} \mathcal{P} \int_{\partial D} \mathbf{n}_{\mathbf{x}_0} \cdot \left[ (\mathbf{n}_x \times \mathbf{H}_t) \times \nabla_x \frac{1}{r} \right] dS_x \quad (2.35)$$

kur  $H_n(\mathbf{x}_0)$  ir magnētiskā lauka intensitāte virsmai tuvojoties no iekšpuses. Tātad redzam, ka labajā vienādojuma pusē ietilpst tikai lauka tangenciālās komponentes  $\mathbf{H}_t = (\mathbf{I} - \mathbf{n} \otimes \mathbf{n}) \mathbf{H}$ , kuras izrēķinām pēc (2.29) atrisināšanas, bet kreisajā pusē nepieciešamā lauka normālā komponente.

## 2.5 Īss kopsavilkums

Vispirms atrisinām robežintegrālvienādojumu (2.29) potenciālam, kuru atvasinot pēc galīgajām diferencēm, iegūstam magnētiskā lauka tangenciālās komponentes. Tās tālāk ievietojam robežintegrālvienādojumā (2.35), kuru nointegrējot iegūstam arī magnētiskā lauka normālo komponenti. Ievietojot šīs lauka komponentes vienādojumā (2.25), kur virsmas liekumu<sup>8</sup> atrodam skaitliski vai arī pārejam uz bezliekuma formulējumu (skatīt § A.5), iegūstam virsmas ātrumu. To attiecīgi integrējot

$$\frac{d\mathbf{x}_0}{dt} = \mathbf{u}(\mathbf{x}_0 \in \{\mathbf{x}\}, \{\mathbf{x}\}, \mu, \gamma, \eta, \mathbf{H}_0) \quad (2.36)$$

iegūstam virsmas konfigurāciju nākamajā laika solī.

Parametrus  $\mathbf{H}_0$ ,  $\mu$ ,  $\eta$ ,  $\gamma$  izvēlamies tā, lai kvalitatīvi apskatītu visas dotā modeļa iespējamās situācijas. No vienādojumiem (2.29) un (2.35) redzam, ka magnētiskā lauka intensitātei uz virsmas izpildās šādas īpašības

$$\mathbf{H}(\{\mathbf{x}\}, \mathbf{x}_0, \mu, \mathbf{H}_0) = C\mathbf{H}\left(\frac{\{\mathbf{x}\}}{C}, \frac{\mathbf{x}_0}{C}, \mu, C\mathbf{H}_0\right) \quad (2.37)$$

$$\mathbf{H}(\{\mathbf{x}\}, \mathbf{x}_0, \mu, C\mathbf{H}_0) = C\mathbf{H}(\{\mathbf{x}\}, \mathbf{x}_0, \mu, \mathbf{H}_0) \quad (2.38)$$

Savukārt apskatot vienādojumus (2.25), (2.36) var iegūt šādas īpašības virsmai piedrošiem punktiem

$$\mathbf{x}_0(\{\mathbf{x}\}, t, \mu, \gamma, \eta, \mathbf{H}_0) = C\mathbf{x}_0(C\{\mathbf{x}\}, t, \mu, C\gamma, \eta, C\mathbf{H}_0) \quad (2.39)$$

$$\mathbf{x}_0(\{\mathbf{x}\}, t, \mu, \gamma, \eta, \mathbf{H}_0) = \mathbf{x}_0(\{\mathbf{x}\}, Ct, \mu, \gamma, C\eta, \mathbf{H}_0) \quad (2.40)$$

$$\mathbf{x}_0(\{\mathbf{x}\}, t, \mu, \gamma, \eta, \mathbf{H}_0) = \mathbf{x}_0(\{\mathbf{x}\}, t, \mu, C^2\gamma, C^2\eta, C\mathbf{H}_0) \quad (2.41)$$

No kā seko, ka problēmas vispārīgajam atrisinājumu var izteikt ar šādu formulu

$$\mathbf{x}_0(\{\mathbf{x}\}, t, \mu, \gamma, \eta, \mathbf{H}_0) = R_0\mathbf{x}_0(R_0\{\mathbf{x}\}, \frac{\eta R_0}{\gamma}t, \mu, 1, 1, \sqrt{\frac{R_0}{\gamma}}\mathbf{H}_0) \quad (2.42)$$

---

<sup>8</sup> $k_m = \frac{1}{2}\nabla \cdot \mathbf{n}$ , skatīt § A.1.

kas ir no virsmas spraiguma  $\gamma$  un viskozitātes  $\eta$  neatkarīgs. No šejienas redzam, ka problēmas raksturīgais laiks ir  $\gamma/\eta R_0$ , bet raksturīgais parametrs

$$B_m = \frac{R_0 H_0^2}{\gamma} \quad (2.43)$$

ko šajā darbā saucam par magnētisko Bonda skaitli<sup>9</sup>.

---

<sup>9</sup>Literatūrā, piemēram [3], izmanto magnētiskā Bonda skaitļa definīciju  $B_m = \frac{R_0 M^2}{\gamma}$ .

# Nodaļa 3

## Virsmas ģeometrijas triangulēšana

Lai atrisinātu iepriekšējā nodaļā formulētos robežintegrālvienādojumus, virsma ir jāsadala elementos. Virsmas diskretizācijai ir jāspēj aprakstīt patieso virsmu un jāadaptējās virsmas deformācijām. Nemot vērā, ka ar trijstūrveida elementiem jau iepriekš ir veiktas brīvo virsmu simulācijas ar lielām deformācijām [19], [28], [18], tad tos arī izmantoju šajā darbā.

Šajā nodaļā apskatu sākotnējās triangulācijas iegūšanu, virsmas normāles un liekuma aprēķināšanu, kas veido minimumu integrālvienādojumu (2.25), (2.29), (2.35) simulācijai. Lai varētu veidot simulācijas ar lielām virsmas deformācijām, izmantoju triangulācijas stabilizācijas algoritmus, kuri apskatīti § 3.3 un testēti ar *Enright* ātruma sadalījumu. Visi šajā nodaļā apskatītie algoritmi un citi<sup>1</sup> atrodami *Surface-Geometry.jl* bibliotēkā<sup>2</sup>.

### 3.1 Virsmas triangulēšana

Par labu virsmas triangulāciju var uzskatīt tādu, kura vistuvāk atrodas patiesajai virsmai, kura sastāv no gandrīz regulāriem trijstūrveida elementiem un ir ar iespējamīn mazāku skaitu elementu, kur pēdējās divas īpašības uzlabo aprēķina precizitātes un

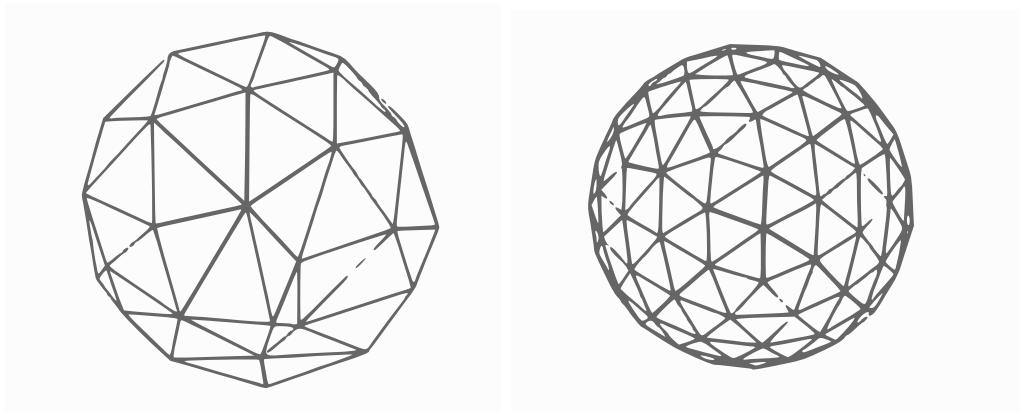
---

<sup>1</sup>Triangulācijas smalcināšana, dažādi iteratori ap virsotni, virsmas ieslēdzosā tilpuma reķināšana, patvaļīga ellipsoīda piedziņšana punktu kopai.

<sup>2</sup>Meklēt Github repozitorijā.

laika attiecību<sup>3</sup>. Bet no pirmās un trešās īpašības seko, ka elementu skaitam jābūt lielākam liela liekuma apgabalos [21].

Patvalīgas virsmas triangulācijai eksistē stabils algoritms no CGAL bibliotēkas [22], tomēr šis algoritms nedod labu triangulāciju<sup>4</sup>, kā tas redzams attēlā 3-1. Lai to uzlabotu, varētu veikt pārtriangulāciju, kur algoritms ar piemēriem atrodams [23], tādēļ varētu iegūt stabilu algoritmu labas triangulācijas ģenerēšanai. Tomēr, nēmot vērā, ka virsmas triangulācija man nepieciešama salīdzinoši vienkāršām un gludām virsmām, darbā izmantoju vienkāršāku algoritmu no bibliotēkas Distmesh [24], [25]. Distmesh virsmas ģenerēšanas algoritms kaut arī nestabils<sup>5</sup>, dod ļoti labas triangulācijas (skatīt attēlu 3-1).



Att. 3-1: CGAL un Distmesh ģenerēto triangulāciju salīdzinājums. Pa kreisi triangulācija ģenerēta ar CGAL, kura kā redzam ir ar trijstūriem ar šauriem leņķiem. Pa labi Distmesh ģenerētā triangulācija, kura ir praktiski ideāla.

## 3.2 Virsmas normāļu un liekuma aprēķināšana

Integrālvienādojumu (2.25), (2.29), (2.35) simulācijai ir nepieciešams zināt virsmas normāles un liekumu, kuri jāiegūst no virsmas triangulācijas. Vēlāk arī redzēsim, ka mūs interesē šie lielumi tieši triangulācijas virsotnēs.

Precīzākais normāļu atrašanas algoritms ir izstrādāts darbā [16], kurš piedzen

---

<sup>3</sup>Komentārus par to var atrast [16], [18], [17]

<sup>4</sup>Trijstūri ir tālu no regulāriem

<sup>5</sup>Triangulācijas ģenerēšanas programma var neapstāties

paraboloīdu virsotnes un apkārtējo kaimiņu virsotņu punktiem. Lai gan šī metode ir ļoti precīza, tā ir jutīgāka pret sliktu triangulāciju [19]<sup>6</sup>. Turklāt tai ir nepieciešams sākotnējais tuvinājums, lai noteiktu, kura puse virsmai ir iekšpusē un kura ir ārpuse.

Normāļu aprēķināšanai tādēļ izmantoju robustāku un vieglāk implementējamu algoritmu no [26]. Tās pamatā ir svērtā vidējā aprēķināšana starp kaimiņu skaldnēm, kur kā svaru koeficienti tiek izmantots trijstūra skaldnes leņķis pie virsotnes. Aprēķinātos normāles vektorus šai metodei pārbaudīju ar analītiskām elipsoīda normālēm,

$$\mathbf{n} \propto (x/a^2, y/b^2, z/c^2) \quad (3.1)$$

kuras sakrita arī pie ļoti izstieptiem elipsoīdiem.<sup>7</sup>

Savukārt virsmas liekumu rēķināju piedzenot paraboloīdu<sup>8</sup> tādā koordinātu sistēmā, kuras  $z$  ass virziens sakrīt ar iepriekš sarežģīto normāles virzienu<sup>9</sup>. Liekuma aprēķināšanas algoritmu pārbaudīju ar analītisko formulu elipsoīdam

$$k_m = \frac{x^2 + y^2 + z^2 - a^2 - b^2 - c^2}{2a^2b^2c^2(x^2/a^4 + y^2/b^4 + z^2/c^4)^{3/2}} \quad (3.2)$$

### 3.3 Virsmas ātruma sadalījuma integrēšana

Lai uzturētu labu triangulāciju integrējot virsmas ātruma sadalījumu, ir jāveic tās stabilizācija. Triangulācijas stabilizācija ir sadalāma divās daļās pasīvā un aktīvajā. Pasīvā stabilizācija cenšas uzturēt labu triangulāciju mainot ātrumu tangenciālo komponenti. Savukārt aktīvā stabilizācija veic triangulācijas ķirurgiju - pievieno/noņem virsotnes, maina saistību (*connectivity*), pārbīda virsotnes pa rekonstruēto virsmu<sup>10</sup>.

Literatūrā atrodamas dažādas metodes, kuras ar šiem uzdevumiem tiek galā [16], [17], [18], [27]. Jaunākā metode [27] ir pieejama kā atvērtā koda bibliotēka, kuru veiksmīgi izmantoju arī šajā darbā. Tomēr šajai metodei ir nepieciešami mazi lai-

<sup>6</sup>Šajā rakstā šo metodi uzlabo un turpina veiksmīgi izmantot pilieniem ar lielām deformācijām. Tomēr tā kļūst sarežģītāk implementējama.

<sup>7</sup>Piemēram  $a = 2$ ,  $b, c = 1/6$ .

<sup>8</sup>Virsotnei un tās kaimiņu virsotnēm (*first ring*).

<sup>9</sup>Šo metodi var uzskatīt arī kā vienkāršojumu [16].

<sup>10</sup>No triangulācijas ir jāiegūst gluda virsma.

ka soli, lai neizveidotos deģenerēta triangulācija, kas īpaši izteikti klūst tad, ja uz virsmas uzdod tikai normālā ātruma komponentes. Tādēļ, lai šo metodi izmantotu integrālvienādojumu simulācijai, ir nepieciešams sarēķināt arī tangenciālās ātruma komponentes. To darbā veicu ar pasīvās triangulācijas stabilizācijas algoritmu, kuras aizsākumi atrodami [16].

### 3.3.1 Virsmas ātrumu integrēšana

Testi parādījuši, ka Eilera metode ir nepietiekamas kārtas metode, lai integrētu virsmas ātrumu sadalījumu, ko parāda tilpuma nesaglabāšanās. Ievērojami labāk strādā Ādamsa Bašforda otrās kārtas metode, kura ir pievilcīga, jo nav nepieciešams rēķināt ātrumu starpsolī un tādēļ tā strādā tikpat ātri kā Eilera metode. Tomēr šī metode ir izmantojama tikai tad, ja virsotņu skaits nemainās, kas vairs nav spēkā, kad sāk lietot aktīvās triangulācijas stabilizācijas algoritmu. Tādēļ šajā darbā visur izmantoju Runges-Kutas (*Runge-Kutta*) otrās kārtas metodi. Bet, lai algoritms būtu stabils, ko aprēķinos redz kā *zāģi*, tad izmantoju laika soli, kas nemaina virsotņu koordinātes vairāk nekā raksturīgais triangulācijas elementa izmērs.

### 3.3.2 Pasīvā stabilizācija

Lai uzturētu labu triangulāciju ilgāku laiku, ātruma tangenciālajām komponentēm ir jābūt tādām, lai nākamajā laika solī triangulācija neveidotos deģenerēti elementi un lai tā būtu uzskatāma par iespējami labāku triangulāciju. Triangulācijas labumu kvantitatīvi definē ar enerģijas funkciju, kuru ar tangenciālo ātrumu komponentēm cenšamies minimizēt. Nemot vērā komentāru no [16]<sup>11</sup>, ka vienkāršākā metode ar atspēru enerģijas funkciju rada būtiskus ierobežojumus uz laika sola lielumu, tad izmantoju vienkāršotāku enerģijas funkciju no [28],

$$F = \sum_{(i,j)} \left[ \frac{d}{dt} (\mathbf{x}_i - \mathbf{x}_j)^2 \right]^2 + 0.4 \sum_{\Delta} \frac{1}{C_{\Delta}^2} \left( \frac{dC_{\Delta}}{dt} \right)^2 \quad C_{\Delta} = \frac{S_{\Delta}}{a^2 + b^2 + c^2} \quad (3.3)$$

---

<sup>11</sup>In the simplest version of the local grid tension method, internode springs are used to resist the mesh distortion, which often produces numerical stiffness with severe limitations on the time step.

kur pirmā summa atbild par virsotņu savstarpējo ātrumu minimizāciju<sup>12</sup>, bet otrā summas daļa neļauj trijstūra elementiem kļūt degenerētiem.

Lai minimizētu to pēc tangenciālajiem ātrumiem, vispirms atrodam šīs energijas funkcijas gradientu pēc ātruma  $\nabla_{v_i} F$  (skatīt § C). Lai saglabātu nosacījumu, ka normālās ātruma komponentes paliek nemainīgas, šo gradientu katrai virsotnei projicējam  $(\mathbf{I} - \mathbf{n} \otimes \mathbf{n}) \nabla_{v_i} F$  un lietojam, piemēram, konjugēto gradientu metodi minimuma atrašanai. Lai uzlabotu konvergences ātrumu, kā sākumnosacījumu nemam esošā soļa ātrumu normāles virzienā plus iepriekšējā soļa ātruma projekciju virsmas pieskarplaknē.

Lai stabilizāciju notestētu izmantoju Enright testu [27], kurā sfēra ar rādiusu 0.15 un centra koordinātēm  $(0.35, 0.35, 0.35)$  ievietota nesaspiežama šķidruma ātruma sadalījumā

$$\begin{aligned} v_x &= 2 \sin^2(\pi x) \sin(2\pi y) \sin(2\pi z) \sin(2/3\pi t) \\ v_y &= -\sin(2\pi x) \sin^2(\pi y) \sin(2\pi z) \sin(2/3\pi t) \\ v_z &= -\sin(2\pi x) \sin(2\pi y) \sin^2(\pi z) \sin(2/3\pi t) \end{aligned} \quad (3.4)$$

Tā kā no integrālvienādojuma (5.6) iegūstu tikai ātruma normālās komponentes, tad pasīvā stabilizācijas algoritma testā padodu tikai ātruma projekcijas normāles virzienā. Iegūtie testa rezultāti (attēls 3-2) parāda, ka izveidotais pasīvās stabilizācijas algoritms ļauj izvairīties no degenerētiem trijstūriem un virsotņu pārblīvēšanās.

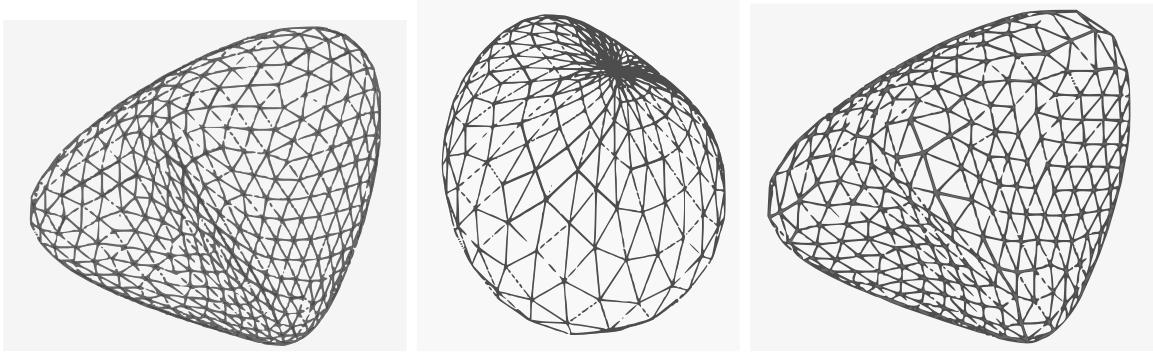
### 3.3.3 Aktīvā stabilizācija

Lai arī pasīvā stabilizācija spēj izsekot virsmas liekuma izmaiņām, tā nav spējīga apakstīt lielas virsmas deformācijas.<sup>13</sup> Tādēļ triangulāciju nepieciešams uzlabot veicot šķautņu kolapsēšanu (*edge collapse*), virsotņu pievienošanu, topoloģijas optimizēšanu

---

<sup>12</sup> To minimize globally, on average, the rates of change of the distances between neighbouring nodes.

<sup>13</sup> Tomēr lielas deformācijas ar pasīvo triangulācijas stabilizāciju ir iespējamas [19]. To panāk atlaujot trijstūra elementus ar vienu īsu bet divām garām malām. Šo algoritmu būtu vēlams notestēt, jo varētu tam varētu būt laba saderība ar ElTopo aktīvās stabilizācijas algoritmu, kur problēmas sagādā platlenķa trijstūri.



Att. 3-2: Pasīvās stabilizācijas algoritma tests ar Enright ātruma sadalījumu. Pa krei-  
si triangulācija laikā  $t = 1$ , kuru iegūst integrējot sākotnējo ievietotās sfēras virsotnes  
pēc ātruma sadalījumu (3.4). Vidū redzama triangulācija, ja katru reizi, kad veic āt-  
ruma integrēšanu to noprojicē virsmas normāles virzienā. Attēlā pa labi triangulācija,  
kad tangenciālos ātrumus atrod pēc energijas funkcijas (3.3) minimizācijas.

ar malas apmešanu (*edge flip*).

Minimāla šo algoritmu implementācija atrodama rakstā [18], kur panāk ka stiep-  
tam pilienam tiek uzturēta laba triangulācija, kura adaptējās virsmas liekumam. To-  
mēr šim algoritmam neatradu koda failus un tā uzrakstīšana bez sagataves varētu  
būt sarežģīts uzdevums, jo rakstā tas aprakstīts ar lieliem soliem. Tā vietā aktīvās  
triangulācijas stabilizācijai izmantoju atvērtā koda bibliotēku ElTopo [27].

Šī algoritma pamatā ir šķautņu kolapsēšana (*edge collapse*), kuras mazākas par uz-  
dotu lielumu  $l_{min}$ , šķautņu dalīšana, kuras lielākas par  $l_{max}$ , šķautņu apmešana (*edge  
flip*), ja tās garums kļūst īsāks, virsotņu pozīcijas relaksēšana (*null-space smoothing*),  
kur katru virsotni novieto viduspunktā starp visām pārejām<sup>14</sup>. Turklāt šīs operā-  
cijas ir veidotas tā lai neveidotos robi liela liekuma apgabalos, lai katra operācija  
neienestu tilpuma izmaiņu par uzdotu lielumu  $\gamma$ , lai operācijas rezultātā neizveidotatos  
trijstūris ar šaurāku lenķi par  $\alpha_{min}$  vai platāku lenķi par  $\alpha_{max}$ . Bibliotēka piedāvā arī  
sadursmju novēršanas algoritmu (*collision response algorithms*), bet to funkcijas jau  
pilda iepriekš izstrādātā pasīvā stabilizācija, tādēļ šī funkcija izmantota netiek.

Šajā darbā visām simulācijām izmantoju aktīvās stabilizācijas parametrus, kuri

---

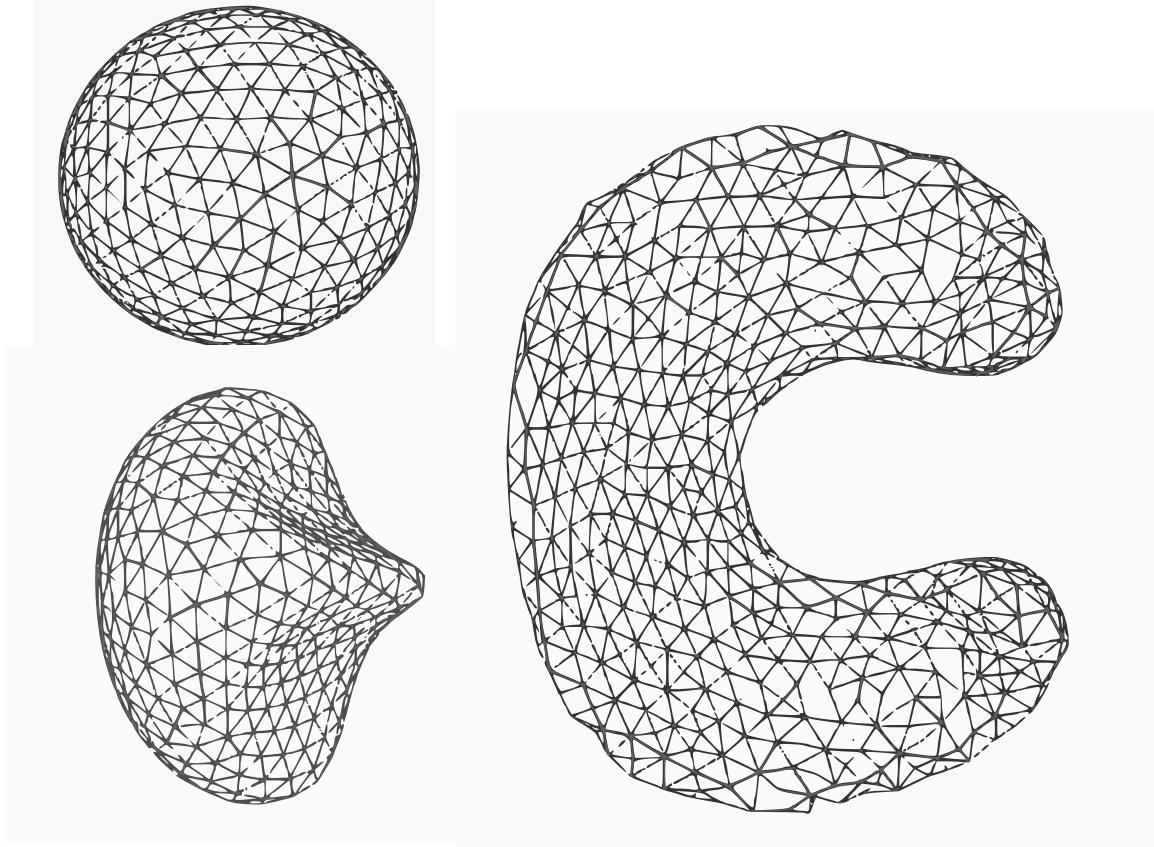
<sup>14</sup>Precīzāk *null-space of their associated quadric metric tensors*.

nedaudz atšķiras no [27] izmantotajiem,

$$l_{min} = 0.5l_0 \quad l_{max} = 1.5l_0 \quad \gamma = 0.1l_0^3 \quad (3.5)$$

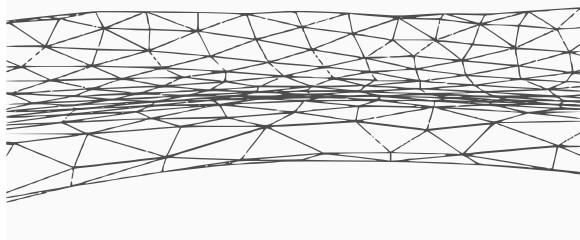
$$\alpha_{min} = 10^0 \quad \alpha_{max} = 160^0 \quad (3.6)$$

kur papildus atstāju ieslēgtu virsotņu pozīcijas relaksēšanu, bet atslēdzu sadursmju novēršanas algoritmu. Šo parametru testam izmantoju jau iepriekš definēto ātruma sadalījuma (3.4) normālās komponentes, kur tangenciālās sareķina ar pasīvo stabilizācijas algoritmu, kur rezultāti redzami attēlos 3-3 un 3-4.



Att. 3-3: Enright tests apvienotam pasīvās un aktīvās stabilizācijas algoritmam. Augšējā stūrī pa kreisi sākotnējā triangulācija, uz leju triangulācija laikā  $t = 1$  un pa labi triangulācija laikā  $t = 1.25$ .

Testos redzams, ka triangulāciju aktīvās stabilizācijas algoritms uztur labu, pat pie lieliem liekuma apgabaliem un pat spēju reproducēt rakstā [27] iegūto testu rezultātu. Atšķirība vērojama pie lieliem liekuma apgabaliem, kur sāk veidoties *zāģis*.



Att. 3-4: Enright testa apvienotam pasīvās un aktīvās stabilizācijas algoritmam skats no labās puses attēlā 3-3 laikā  $t = 1.25$ . Redzams, ka algoritms spēj uzturēt labu triangulāciju pat pie lieliem virsmas liekumiem.

Tā cēlonis no vienas puses ir pasīvās triangulācijas stabilizācijas neadaptēšanās virsmas liekumam, kur vēlams būtu, lai virsotņu tangenciālie ātrumi būtu minimāli un virsotņu blīvums lielāks.<sup>15</sup> No otras puses pasīvā triangulācijas stabilizācija izmanto tuvināti sareķinātas normāles, kad izvēlas ātruma virzienu tangenciālā virzienā. Tas savukārt perturbē ātrumu īstās virsmas normāles virzienā, kas kā sekas izsauc arī tilpuma nesaglabāšanos.<sup>16</sup>

---

<sup>15</sup>Uzlabojums varētu būt pasīvās stabilizācijas algoritms no [19].

<sup>16</sup>Risinājums būtu izmantot precīzāk sareķinātas normāles, kuras var iegūt no vienas puses ar rakst [19] algoritmu, bet no citas puses tās būtu precīzākas, ja uz šī lielā liekuma apgabala būtu vairāk virsotņu. Tātad liekuma adaptīvs pasīvās stabilizācijas algoritms šeit varētu strādāt.

# Nodaļa 4

## Magnētiskā lauka aprēķināšana

Lai izmantotu robežintegrālvienādojumu formulējumu piliena virsmas evolūcijas simulācija (2.25), ir nepieciešams zināt lauka normālās un tangenciālās komponentes visos virsmas punktos. Tās patvaļīgai virsmas ģeometrijai atradīsim risinot robežintegrālvienādojumus, kuriem ir iespējami vairāki formulējumi. Tā kā literatūrā nav vienošanās par labāko (*state of the art*) algoritmu Laplasa vienādojuma risināšanai un atvasinājuma atrašanai uz robežas, tad šajā nodaļa apskatīsim dažādus robežintegrālvienādojumu formulējumus un to skaitlisku risināšanu.

### 4.1 Robežintegrālvienādojumu diskretizācija

Robeželementu metodes diskretizācija sākas ar integrāla sadališanu pa parametrizējamiem elementiem jeb šajā gadījumā trijstūriem. Nākamajā solī izvēlas nezināmās funkcijas interpolāciju pa elementiem  $\psi(\mathbf{x}) \approx \sum_k \psi_k \phi_k(\mathbf{x})$ , piemēram, konstanta funkcijas vērtība uz elementa vai arī sarežģītāka kā lineāra, kvadrātiska vai vispārīgā gadījumā patvaļīga polinoma interpolācija. Kā piemēru šajā sadaļā apskatām robežintegrālvienādojuma potenciālam rēķināšanu uz virsmas pēc vienādojuma (2.29).

$$\psi(\mathbf{x}_0) = \frac{2\mathbf{H}_0 \cdot \mathbf{x}_0}{\mu + 1} - \frac{1}{2\pi} \frac{\mu - 1}{\mu + 1} \mathcal{P} \int_{\partial D} \psi(\mathbf{x}) \frac{\partial}{\partial n_x} \frac{1}{|\mathbf{x} - \mathbf{x}_0|} dS_{\mathbf{x}} \quad (4.1)$$

Izvēloties lineāru nezināmās funkcijas (potenciāla) interpolāciju pa trijstūra ele-

mentu un sadalot robežintegrāli vienādojumā (2.29) trijstūra elementos iegūst

$$\mathcal{P} \int_{\partial D} \psi \frac{\partial}{\partial n_{\mathbf{x}}} \frac{1}{|\mathbf{x} - \mathbf{x}_0|} dS_{\mathbf{x}} = \sum_i \sum_j \psi_j \int_{\Delta_i} \phi_j(\mathbf{x}) \frac{\partial}{\partial n_{\mathbf{x}}} \frac{1}{|\mathbf{x} - \mathbf{x}_0|} dS_{\mathbf{x}} \quad (4.2)$$

kur  $\phi_j(\mathbf{x})$  ir interpolācijas bāzes funkcijas. Integrālus pa trijstūriem integrēsim ar 3 punktu trapeces kvadratūru (*trapezodial rule*), ko plaši pielieto Stoksa robežintegrālvienādojumu piliena simulācijās [16], [17], [18]. Kaut arī šādā veidā tiek upurēta kvadratūras precizitāte, matricu vienādojumu var uzkonstruēt ievērojami ātrāk, tādēļ ielaisto kļūdu var kompensēt ar sīkāku diskretizāciju. Integrējot nesingulāros elementus ar trapeces kvadratūras formulu un izlaižot singulāru integrāļu aprēķināšanu,<sup>1</sup> iegūstam robežintegrāļa aprēķināšanas formulu dotam  $\mathbf{x}_0$ ,

$$\begin{aligned} & \int_{\partial D} \psi(\mathbf{x}) \frac{\partial}{\partial n_{\mathbf{x}}} \frac{1}{|\mathbf{x} - \mathbf{x}_0|} dS_{\mathbf{x}} \\ &= \sum_i \left( \psi_1 \frac{\partial}{\partial n_{\mathbf{x}_1}} \frac{1}{|\mathbf{x}_1 - \mathbf{x}_0|} + \psi_2 \frac{\partial}{\partial n_{\mathbf{x}_2}} \frac{1}{|\mathbf{x}_2 - \mathbf{x}_0|} + \psi_3 \frac{\partial}{\partial n_{\mathbf{x}_3}} \frac{1}{|\mathbf{x}_3 - \mathbf{x}_0|} \right) \Delta S_{i123} \end{aligned} \quad (4.3)$$

kur  $\Delta S_{i123}$  ir i-tā trijstūra laukums, bet  $\mathbf{x}_1$ ,  $\mathbf{x}_2$  un  $\mathbf{x}_3$  tā virsotņu koordinātes.

Šo formulu var pierakstīt arī vienkāršāk un efektīvāk pārejot no summēšanas pa trijstūriem uz summēšanu pa virsotnēm tā iegūstot<sup>2</sup>

$$\int_{\partial D} \psi(\mathbf{x}) \frac{\partial}{\partial n_{\mathbf{x}}} \frac{1}{|\mathbf{x} - \mathbf{x}_0|} dS_{\mathbf{x}} = \sum_j \psi_j \frac{\partial}{\partial n_{\mathbf{x}_j}} \frac{1}{|\mathbf{x}_j - \mathbf{x}_0|} \Delta S_j \quad (4.4)$$

kur  $\Delta S_j$  ir viena trešdaļa no laukuma visiem trijstūriem, kuriem ir kopīga virsotne  $j$ .

Lai atrastu interpolēto funkciju  $\psi_j$ , ir jāsastāda vienādojumu sistēma. Viena pieeja ir robežintegrālvienādojumu integrēt ar  $N$  neatkarīgām testa funkcijām, tā definējot sistēmu  $N$  nezināmiem  $\psi_j$ , kas ir Galerkina metodes pamatā. No citas pusē varam pieprasīt vienādojuma izpildīšanos uz virsma konkrētos punktos (kolokāciju metode),

---

<sup>1</sup>Vienkāršības labad singulāro integrāļu integrēšanu aizstājam ar 0, kas dos konverģējošu atrisinājumu, jo izlaistais elements kurā atrodas singularitāte var tikt uzskatīts kā *epsilona apkārtne* (skatīt § 2.4).

<sup>2</sup>Šī formula rakstīta gadījumam, kad visi integrāļi pa elementiem ir regulāri. Lai to izmantotu singulāriem integrāļiem, varam izskaitīt pēc summēšanas nepareizi nointegrētos trijstūra elementus un aizstāt ar vēlamo kvadratūru (šajā sadaļā 0).

kurus parasti izvēlās kā elementa virsotnes un tā iegūstot  $N$  vienādojumus. Tomēr kolokāciju punktus var novietot elementa iekšpusē pie elementa definējošām virsotnēm vai arī speciālās vietās (kāda polinoma saknes), kā to dara spektrālās metodes [29]<sup>3</sup>. Šādā gadījumā iegūst vairāk vienādojumus nekā ir nezināmie, tādēļ parasti izmanto mazāko kvadrātu metodi, lai atrastu  $\psi_j$ .

Šajā darbā kolokāciju punktus vienmēr novietoju uz trijsstūru virsotnēm un tā iegūstot tikpat vienādojumus cik ir nezināmie

$$\psi_i = \frac{2\mathbf{H}_0 \cdot \mathbf{x}_i}{\mu + 1} - \frac{1}{2\pi} \frac{\mu - 1}{\mu + 1} \sum_j \psi_j \frac{\partial}{\partial n_{\mathbf{x}_j}} \frac{1}{|\mathbf{x}_j - \mathbf{x}_i|} \Delta S_j \quad (4.5)$$

kur  $\psi_i = \psi(\mathbf{x}_j)$ . Doto lineāro vienādojumu sistēmu tad varam risināt ar jau julia programmēšanas valodā iebūvētajām standarta lineārās algebras bibliotēkām.

## 4.2 Potenciāla aprēķināšana

### 4.2.1 Robežintegrālvienādojuma regularizācija

Regularizācijai izmantosim sekojošu identitāti (skatīt § A.7)

$$\mathcal{P} \int \frac{\partial}{\partial n_{\mathbf{x}}} \frac{1}{4\pi |\mathbf{x} - \mathbf{x}_0|} dS_{\mathbf{x}} = \frac{1}{2} \quad (4.6)$$

Pareizinot šo vienādojumu ar  $2(\mu - 1)/(\mu + 1)\psi_{\mathbf{x}_0}$  un pieskaitot to (2.29) iegūstam regularizēto robežintegrālvienādojumu versiju

$$\psi(\mathbf{x}_0) \left( 1 + \frac{\mu - 1}{\mu + 1} \right) = \frac{2\mathbf{H}_0 \cdot \mathbf{x}_0}{\mu + 1} - \frac{1}{2\pi} \frac{\mu - 1}{\mu + 1} \int_{\partial D} [\psi(\mathbf{x}) - \psi(\mathbf{x}_0)] \frac{\partial}{\partial n_{\mathbf{x}}} \frac{1}{|\mathbf{x} - \mathbf{x}_0|} dS_{\mathbf{x}} \quad (4.7)$$

kur  $\mathcal{P} \int \rightarrow \int$ , jo integrālis uzskatāms par regulāru<sup>4</sup>. Iegūto integrālvienādojumu varam risināt piemēram ar § 4.1 apskatīto algoritmu.

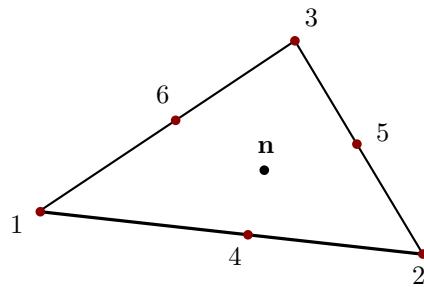
---

<sup>3</sup>Ar spektrālām metodēm var iegūt eksponenciālu konvergences ātrumu.

<sup>4</sup>Par regulāru šajā darbā saucu integrāli, kuram zemintegrāļa funkcijā nav singularitātes.

## 4.2.2 Regularizētā robežintegrālvienādojuma risināšana ar liektiem elementiem

Literatūrā daudz tiek apskatīti liektie elementi, kuri uzlabo konverģenci [32]. Piemēram [30] iesaka lietot regularizēto robežintegrālvienādojumu ar liektiem elementiem, kurus integrē ar Gausa kvadratūru trijstūrim [33]. Aprēķinā esmu izvēlējies kvadrātiskus elementus, kuri definēti ar 3 trijstūra virsotnēm un 3 virsotnēm uz trijstūra elementa malām.



Att. 4-1: Liekta kvadrātiska elementa raksturojošie parametri. Trijstūra virsotnes 1, 2, 3 un virsotnes uz trijstūra malām 4, 5, 6.

Apzīmējot ar  $\xi$  un  $\eta$  taisnlenķa trijstūra koordinātes (skatīt attēlu 4-2), tad, lai aprēķinātu jebkuru punktu uz liektā elementa, izmantojam formulu [30]

$$\begin{aligned}
 \mathbf{x} &= \sum x_i \phi_i(\xi, \eta) \\
 \phi_2 &= \frac{1}{1-\alpha} \xi \left( \xi - \alpha + \frac{\alpha - \gamma}{1-\gamma} \eta \right) \\
 \phi_3 &= \frac{1}{1-\beta} \eta \left( \eta - \beta + \frac{\beta + \gamma - 1}{\gamma} \xi \right) \\
 \phi_4 &= \frac{1}{\alpha(1-\alpha)} \xi(1 - \xi - \eta) \\
 \phi_5 &= \frac{1}{\gamma(1-\gamma)} \xi \eta \\
 \phi_6 &= \frac{1}{\beta(1-\beta)} \eta(1 - \xi - \eta) \\
 \phi_1 &= 1 - \phi_2 - \phi_3 - \phi_4 - \phi_5 - \phi_6
 \end{aligned} \tag{4.8}$$

$$\alpha = \frac{1}{1 - |\boldsymbol{x}_4 - \boldsymbol{x}_2|/|\boldsymbol{x}_4 - \boldsymbol{x}_1|}, \beta = \frac{1}{1 + |\boldsymbol{x}_5 - \boldsymbol{x}_2|/|\boldsymbol{x}_6 - \boldsymbol{x}_1|}, \gamma = \frac{1}{1 + |\boldsymbol{x}_5 - \boldsymbol{x}_2|/|\boldsymbol{x}_5 - \boldsymbol{x}_3|}. \quad (4.9)$$

Tā ņauj izteikt integrēšanu pa liekto elementu šādi

$$\int_{\Delta} f(\boldsymbol{x}) dS = \int_0^1 d\xi \int_0^{1-\xi} f[\boldsymbol{x}(\xi, \eta)] \left| \frac{\partial \boldsymbol{x}}{\partial \xi} \times \frac{\partial \boldsymbol{x}}{\partial \eta} \right| d\eta. \quad (4.10)$$

kur labo pusi integrē ar trijstūra Gausa kvadratūru. Manā implementācijā apskatu lineāru nezināmās funkcijas interpolāciju starp trijstūra virsotnēm un izmantoju 3 punktu Gausa kvadratūru, kas samazināja klūdu nepalielinot mainīgo skaitu (skatīt § 4.2.4 ).

### 4.2.3 Singulāra robežintegrālvienādojuma risināšana ar adaptīvo integrēšanu

Ja integrālvienādojumam nav atrodama regularizētā forma, tad singulāros elementus var integrēt ar specializētu kvadratūru. Šādiem gadījumiem literatūrā atrodama lokālā regularizācijas procedūra [34], kurā integrējamo funkciju izvirza Teilora rindā un iegūst regularizētu integrāli pa elementu. Citāda pieeja ir singulāro elementu integrēt ar specializētu kvadratūru, piemēram, [35], [36] un [37], kur pēdējo izmantoju šajā sadaļā. Kā piemēru apskatīsim singulārā integrālvienādojuma (2.29) risināšanu.

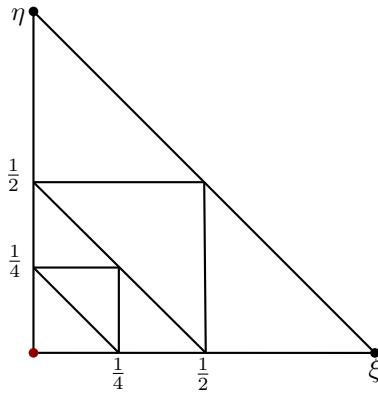
Plakana elementa ieguldījums pie singularitātes ir  $0^5$ , tādēļ ir jāizmanto augstākas kārtas virsmas aproksimācija [30], ko panākam ar kvadrātiskajiem elementiem § 4.2.2. Lai aprēķinu vienkāršotu, tas ir, padarītu ātrāku, nesingulāros elementus integrējam ar trapeces likumu kā § 4.1<sup>6</sup>. Savukārt singulārajiem elementiem izmantojam adaptīvo kvadratūru [37].

Šīs adaptīvās kvadratūras pamatā ir integrējamā trijstūra sagriešana 4 mazākos, kur 3 integrēti ar nesingulāro kvadratūru, bet atlikušo atkal dala līdz trijstūris pie singularitātes, kļūst neievērojami mazs (skatīt 4-2). Nesingulārie trijstūri tiek integ-

---

<sup>5</sup>Trijstūra normāle ir perpendikulāra vektoram, kas sākās no virsotnes un atrodas uz virsmas.

<sup>6</sup>Novērtos, ka aprēķinos tas arī uzlabo precizitāti nekā tad, ja izmanto Gausa kvadratūru.



Att. 4-2: Adaptīvās kvadratūras shēma, kad veiktas 2 dalīšanas iterācijas. Singulārajam trijstūrim ar virsotni  $\xi = 0, \eta = 0$  piekārto vērtību 0, bet pārējos integrē ar 3 punktu Gausa kvadratūru.

rēti ar 3 punktu Gausa kvadratūru, un trijstūru dalīšana notiek maksimāli 50 reizes, bet tiek pārtraukta, kad konverģence pārstāj būt vienmērīga.

#### 4.2.4 Potenciāla rēķināšanas algoritmu testi

Magnetostatikas tuvinājumā elipsoīdā ar lineāru magnetizāciju, kas ievietots ārējā magnētiskā laukā  $\mathbf{H}_0$ , ir homogēns lauks ar lauka intensitāti [40]

$$H_i^{(i)} = \frac{H_{0i}}{1 + (\mu - 1)n_i} \quad (4.11)$$

kur  $n_i$  ir demagnetizācijas koeficients. Elipsoīdam tas ir izteikts kā

$$n_i = \frac{1}{2}abc \int_0^\infty \frac{ds}{(s + a^2)R_s} \quad R_s^2 = (s + a^2)(s + b^2)(s + c^2) \quad (4.12)$$

kur integrēšanu viegli var veikt skaitliski. Var piebilst, ka demagnetizācijas koeficientu summa kopā dod  $\sum_i n_i = 1$  un tādēļ sfērai tas ir vienāds ar  $n_i = 1/3$  un cilindram  $n_{1,2} = 1/2$ . Šajā sadaļā aprēķināšana pārbaudei izmantojam magnētisko potenciālu  $\mathbf{H} = \nabla\psi$ . Ja uzdots robežnosacījums bezgalībā  $\psi \rightarrow \mathbf{H}_0 \cdot \mathbf{x}$ , tad uz elipsoīda virsmas tas ir

$$\psi_{elips} = x_i H_i^{(i)} = \frac{H_{0i}x_i}{1 + (\mu - 1)n_i}. \quad (4.13)$$

Aprēķinu precizitātes novērtēšanai katrai triangulācijas virsotnei rēķinu relatīvo klūdu  $r = \frac{|calculated - theoretical|}{|theoretical|}$ . Lai to visu kopu varētu apstrādāt, apskatu vidējo relatīvo klūdu

$$\langle r \rangle = \frac{1}{N} \sum_i^N \frac{|calculated_i - theoretical_i|}{|theoretical_i|} \quad (4.14)$$

kura dod ieskatu par relatīvo klūdu katrā virsotnē. Tomēr šis novērtējums nav pieiekams, jo tas nedod ziņas par korelāciju starp relatīvo klūdu un lieluma absolūto vērtību, kas ir svarīgi, jo integrālvienādojumā (2.25) interesants ir tieši kumulatīvais klūdu ieguldījums. Tā pētišanai rēķinu lielumu

$$q = \frac{\sum_i^N |calculated_i - theoretical_i|}{\sum_i^N |theoretical_i|} \quad (4.15)$$

kas ir lielāks (mazāks) par vidējo relatīvo klūdu, ja lielākas (mazākas) relatīvas klūdas ir ar absolūto vērtību lielākiem (mazākiem) lielumiem.

Darbu izstrādājot esmu ievērojis, ka, piemēram, izlaižot singulārā elementa integrēšanu, relatīva klūda par aptuveni vienādu lielumu pieaug visām virsotnēm, kas tādējādi varētu kalpot par indikatoru nepilnīgai singularitātes apstrādei. Lai to redzētu izmantoju relatīvo klūdu kopas sadalījuma kvantiles, kuras definētas ar

$$P(r_\xi) < \xi \quad (4.16)$$

kur  $P$  šajā gadījumā ir normēts histogrammas kumulatīvais sadalījums relatīvam klūdām. Piemēram,  $r_{1/2}$  atbilst mediānai, bet  $r_{1/4}$  ir relatīvā klūda par kuru ir mazāka vismaz  $1/4$  no sarēķinātajiem lielumiem triangulācijas virsotnēs. Izmantojot visas šīs klūdu metrikas ar dažādiem aprēķināšanas algoritmiem un triangulācijām, iegūst tabulu 4.1.

Tabulā 4.1 varam redzēt, ka vislielākā relatīvā klūda ir formulējumam, kura singulāro elementa integrēšanu izlaiž § 4.1, bet tomēr tā samazinās palielinot elementu skaitu.<sup>7</sup> Regularizētais formulējums (4.7) strādā ievērojami labāk un tā aprēķina pre-

---

<sup>7</sup>Elementu skaitu var daudzkāršot ar multipolu metodi būtiski nepalielinot algoritma izpildes

Metode	$a$	$b$	$c$	$N$	$\langle r \rangle, \%$	$q, \%$	$r_{1/4}, \%$	$r_{1/2}, \%$	$r_{3/4}, \%$
singulārs izlaižot singulāro elementu § 4.1	1	1	1	150	11.2789	11.2352	11.1987	11.2984	11.3516
	1	1	1	472	6.5781	6.564	6.5272	6.5772	6.6306
	1	1	1	1828	3.4078	3.3979	3.378	3.4053	3.4263
	2	$\frac{1}{4}$	$\frac{1}{4}$	684	39.7824	42.4724	35.956	37.3057	41.0605
	$\frac{1}{4}$	1	1	408	3.4175	1.4273	0.6912	1.2341	3.9719
regularizēts ar trapeces kvadratūru § 4.2.1	1	1	1	150	2.0235	1.8669	1.8422	1.8628	1.8897
	1	1	1	472	0.5754	0.5786	0.5687	0.5799	0.5856
	1	1	1	1828	0.1569	0.1472	0.1455	0.1471	0.149
	2	$\frac{1}{4}$	$\frac{1}{4}$	684	0.4568	0.334	0.3162	0.3497	0.3606
	$\frac{1}{4}$	1	1	408	4.3939	1.265	0.7105	1.153	2.1992
regularizēts ar liektiem elementiem § 4.2.2	1	1	1	150	0.5109	0.5184	0.5148	0.5179	0.5215
	1	1	1	472	0.1712	0.1718	0.17	0.1718	0.1735
	1	1	1	1828	0.0475	0.0473	0.0471	0.0473	0.0474
	2	$\frac{1}{4}$	$\frac{1}{4}$	684	0.3339	0.3699	0.261	0.2827	0.3441
	$\frac{1}{4}$	1	1	408	1.7381	0.4027	0.1346	0.3704	0.6941
singulārs ar adaptīvo kvadratūru § 4.2.3	1	1	1	150	0.3429	0.2027	0.1432	0.2176	0.2814
	1	1	1	472	0.3114	0.3071	0.2743	0.3152	0.3401
	1	1	1	1828	0.2406	0.2239	0.2132	0.2275	0.2378
	2	$\frac{1}{4}$	$\frac{1}{4}$	684	7.9342	9.7833	5.6156	6.2738	8.3436
	$\frac{1}{4}$	1	1	408	2.6536	0.7327	0.115	0.4004	2.9295

Tabula 4.1: Potenciāla rēķināšanas algoritmu salīdzinājums pie  $\mu = 10$ . Elipsoīda pusasis  $a, b, c$ , triangulācijas virsotņu skaits  $N$ , vidējā relatīvā klūda  $\langle r \rangle$  (4.14), relatīvās klūdas raksturojošs lielums  $q$  (4.15), relatīvo klūdu kvantiles  $r_{1/4}, r_{1/2}, r_{3/4}$  (4.16).

cizitātē ievērojami uzlabojas izmantojot liektos elementus, kā tas bija arī sagaidāms [32], [35]. Redzams, ka veiksmīgi strādā arī singulārais formulejums (2.29), kad singulāro elementu integrē ar adaptīvo kvadratūru § 4.2.3. Tomēr lielas grūtības šim algoritmam sagādā izstiepts elipsoīds.

Ņemot vērā, ka liektajiem elementiem ir vajadzīgas papildvirsotnes uz trijstūra malām (skatīt attēlu 4-1), tad brīvo virsmu simulācijām robustāks § 4.2.1 algoritms, kur integrēšanu veic pa plakaniem elementiem<sup>8</sup>. Turklat algoritms ar liektajiem elementiem strādā par vidēji 6 reizēm ilgāk, jo integrēšanas punktu ir vairāk. No tabulas 4.1 redzam, ka šim algoritmam ir konvergences ātrums aptuveni  $r \propto N^{-1}$ .<sup>9</sup> Relatīvās laiku [31].

<sup>8</sup>Ja triangulācijas topoloģija aprēķina laikā nemainās, tad implementācija ir samērā vienkārša arī liektiem elementiem.

<sup>9</sup>Un ņemot vērā, ka algoritms ir ar  $O(N^2)$  sarežģītību, tad relatīvā klūda no aprēķinam nepieciešamā laika ir  $r \propto t^{-1/2}$ .

klūdas ir neatkarīgas no potenciāla absolūtās vērtības, jo  $r \approx q$  un spēj labi risināt problēmu dažādām gludām ģeometrijām.

## 4.3 Lauka normālās komponentes aprēķināšana

### 4.3.1 Singulārais formulējums

Elektriskā vai magnētiskā lauka normālās komponentes atrašanai 2 dimensijās un aksiālsimetriskā gadījumā iepriekš ir izmantots singulārais formulējums [7], [8], [9], [10], kuru izvedīsim šajā sadalā. Izmantosim sekojošu identitāti (skatīt § A.4)

$$C(\mathbf{x}_0)\mathbf{B}(\mathbf{x}_0) = \int \mu \mathbf{J} \times \nabla_{\mathbf{x}} \frac{1}{r} dV - \int \left[ (\mathbf{n}_{\mathbf{x}} \times \mathbf{B}) \times \nabla_{\mathbf{x}} \frac{1}{r} + (\mathbf{n}_{\mathbf{x}} \cdot \mathbf{B}) \nabla_{\mathbf{x}} \frac{1}{r} \right] dS_{\mathbf{x}} \quad (4.17)$$

kur  $C(\mathbf{x}_0)$  ir  $8\pi$ , ja  $\mathbf{x}_0$  atrodas iekšpusē,  $4\pi$  uz virsmas un 0, ja ārpus integrējamās virsmas noslēgtā apgabala, bet  $\mathbf{J}$  ir brīvās strāvas, kuras apskatītajā modelī nav.

Izvēloties normāles virzienu vērstu no piliena uz ārpusi, var iegūt robežintegrālvienādojumu pilienam

$$2\pi \mathbf{B}^{(i)}(\mathbf{x}_0) = - \int_{\partial D} \left[ (\mathbf{n}_{\mathbf{x}} \times \mathbf{B}^{(i)}) \times \nabla_{\mathbf{x}} \frac{1}{r} + (\mathbf{n}_{\mathbf{x}} \cdot \mathbf{B}^{(i)}) \nabla_{\mathbf{x}} \frac{1}{r} \right] dS_{\mathbf{x}} \quad (4.18)$$

un apkārtējam apgabalam

$$2\pi \mathbf{B}^{(e)}(\mathbf{x}_0) = + \int_{\partial D} \left[ (\mathbf{n}_{\mathbf{x}} \times \mathbf{B}^{(e)}) \times \nabla_{\mathbf{x}} \frac{1}{r} + (\mathbf{n}_{\mathbf{x}} \cdot \mathbf{B}^{(e)}) \nabla_{\mathbf{x}} \frac{1}{r} \right] dS_{\mathbf{x}} + 4\pi \mathbf{B}_0(\mathbf{x}_0) \quad (4.19)$$

$$\mathbf{B}_0(\mathbf{x}_0) = - \frac{1}{4\pi} \int_{\partial \infty} \left[ (\mathbf{n}_{\mathbf{x}} \times \mathbf{B}^{(e)}) \times \nabla_{\mathbf{x}} \frac{1}{r} + (\mathbf{n}_{\mathbf{x}} \cdot \mathbf{B}^{(e)}) \nabla_{\mathbf{x}} \frac{1}{r} \right] dS_{\mathbf{x}} \quad (4.20)$$

Pieskaitot ar  $\mu$  pareizinātu vienādojumu (4.19) vienādojumam (4.18) un ievietojot robežnosacījumus (2.5) iegūst vienādojumu magnētiskā lauka intensitātes normālajai

komponentei

$$H_n(\mathbf{x}_0) = \frac{2\mathbf{H}_0 \cdot \mathbf{n}_{\mathbf{x}_0}}{\mu + 1} + \frac{1}{2\pi} \frac{\mu - 1}{\mu + 1} \mathcal{P} \int_{\partial D} H_n(\mathbf{x}) \frac{\partial}{\partial n_{\mathbf{x}_0}} \frac{1}{r} dS_{\mathbf{x}}. \quad (4.21)$$

Iegūtā integrālvienādojuma forma ir ļoti līdzīga potenciāla integrālvienādojumam (2.29), ar būtisku atšķirību, ka zem integrāla vienādojumā (4.21) atvasināšana ir pēc ārejās normāles  $\mathbf{n}_{\mathbf{x}_0}$ . Tādēļ līdzīgu regularizācijas procedūru, kā § 4.2.1, veikt nevaram.

Kā iepriekš jau redzējām rēķinot singulāro potenciāla integrālvienādojumu § 4.2.3 un § 4.2.4, singulārie elementi ir jāintegrē, lai būtiski samazinātu aprēķinātā lauka normālās komponentes klūdu.

### 4.3.2 Vāji singulārs formulējums

Citāda pieeja balstās uz vāji singulāru robežintegrālvienādojuma risināšanu, ko iegūstam no Grīna formulas (skatīt § A.7)

$$2\pi\psi(\mathbf{x}_0) = \int \frac{\partial\psi}{\partial n_{\mathbf{x}}} \frac{1}{r} dS_{\mathbf{x}} - \int \psi(\mathbf{x}) \frac{\partial}{\partial n_{\mathbf{x}}} \frac{1}{r} dS_{\mathbf{x}} \quad (4.22)$$

Izmantojot sakarību  $\mathcal{P} \int \frac{\partial}{\partial n_{\mathbf{x}}} \frac{1}{r} dS_{\mathbf{x}} = \frac{1}{2}$  (skatīt § A.7) regularizētu integrālvienādojumu

$$\int \frac{\partial}{\partial n_{\mathbf{x}}} \frac{1}{|\mathbf{x} - \mathbf{x}_0|} (\psi(\mathbf{x}) - \psi(\mathbf{x}_0)) dS_{\mathbf{x}} = \int \frac{1}{|\mathbf{x} - \mathbf{x}_0|} \frac{\partial\psi}{\partial n_{\mathbf{x}}} dS_{\mathbf{x}} \quad (4.23)$$

Kreisā vienādojuma puse ir viegli sareķināma ar 3 punktu trapezes integrēšanas formulu *trapezodial rule*, jo singulārā punktā integrējamā funkcija vienāda ar 0. Savukārt labajai pusei izmantojam 3 punktu trapezes integrēšanas formulu nesingulārajiem elementiem, bet vāji singulārajiem veiksim polāro transformāciju.

Apzīmējot ar  $q(\mathbf{x})$  patvalīgu funkciju, kuru vēlamies integrēt pa singulāra trijstūra elementu, un ar  $\mathbf{x}_1$  uz trijstūra virsotnes atrodošos singularitātes koordināti, varam

integrēšanai pa vāji singulāru trijstūra elementu veikt ar sekojošu transformāciju [30]

$$\begin{aligned} \int_{\Delta} \frac{q(\mathbf{x})}{|\mathbf{x} - \mathbf{x}_1|} dS_{\mathbf{x}} &= h_S \int_0^1 \int_0^{1-\xi} \frac{q[\mathbf{x}(\xi, \eta)]}{|(\mathbf{x}_2 - \mathbf{x}_1)\xi + (\mathbf{x}_3 - \mathbf{x}_1)\eta|} d\eta d\xi \\ &= \frac{h_S}{|\mathbf{x}_2 - \mathbf{x}_1|} \int_0^{\pi/2} \frac{\int_0^{R(\chi)} q(\rho, \chi) d\rho}{\sqrt{\cos^2 \chi + B \sin 2\chi + C \sin^2 \chi}} d\chi \end{aligned} \quad (4.24)$$

kur

$$\begin{aligned} \xi &= \rho \cos \chi & \eta &= \rho \sin \chi & h_S &= |(\mathbf{x}_2 - \mathbf{x}_1) \times (\mathbf{x}_3 - \mathbf{x}_1)| \\ B &= \frac{(\mathbf{x}_3 - \mathbf{x}_1) \cdot (\mathbf{x}_2 - \mathbf{x}_1)}{|\mathbf{x}_2 - \mathbf{x}_1|^2} & C &= \frac{|\mathbf{x}_3 - \mathbf{x}_1|^2}{|\mathbf{x}_2 - \mathbf{x}_1|^2} & R(\chi) &= \frac{1}{\cos \chi + \sin \chi} \end{aligned} \quad (4.25)$$

Skaitliski šo divkāršo integrāli rēķinu ar divkāršu Gausa kvadratūru, kur izvēlos 10 punktus integrēšanai pēc  $\rho$  un 10 punktus integrēšanai pēc  $\chi$ .

### 4.3.3 Pārrēķinot no lauka tangenciālās komponentes

Iepriekš ieguvām integrālvienādojumu, kas ļauj pārrēķināt tangenciālā lauka komponentes uz lauka normālajām komponentēm, kā Bio-Savāra integrāli (2.35):

$$H_n(\mathbf{x}_0) = \frac{\mathbf{H}_0 \cdot \mathbf{n}_{\mathbf{x}_0}}{\mu} + \frac{1}{4\pi} \frac{1-\mu}{\mu} \int_{\partial D} \mathbf{n}_{\mathbf{x}_0} \cdot \left[ (\mathbf{n}_{\mathbf{x}} \times \mathbf{H}) \times \nabla_{\mathbf{x}} \frac{1}{r} \right] dS_{\mathbf{x}} \quad (4.26)$$

Labajā vienādojuma pusē ietilpst tikai lauka tangenciālās komponentes, kuras protam precīzi sareķināt izmantojot skalāro potenciālu § 4.2.4. Tomēr šādā formā uzrakstīts integrālvienādojums ir singulārs, tādēļ pirms tā skaitliskas integrēšanas veiksim regularizāciju.<sup>10</sup>

Izmantojot vektoralgebras sakarības  $\mathbf{a} \times (\mathbf{b} \times \mathbf{c}) + \mathbf{b} \times (\mathbf{c} \times \mathbf{a}) + \mathbf{c} \times (\mathbf{a} \times \mathbf{b}) = 0$ ,  $\mathbf{a} \times (\mathbf{b} \times \mathbf{c}) = \mathbf{b}(\mathbf{a} \cdot \mathbf{c}) - \mathbf{c}(\mathbf{a} \cdot \mathbf{b})$  varam pārrakstīt labā pusē esošo integrāli šādi

$$H_n(\mathbf{x}_0) = \frac{\mathbf{H}_0 \cdot \mathbf{n}_{\mathbf{x}_0}}{\mu} + \frac{1}{4\pi} \frac{1-\mu}{\mu} \int_{\partial D} \mathbf{n}_{\mathbf{x}_0} \cdot \left[ \mathbf{H}_t \frac{\partial}{\partial n_{\mathbf{x}_0}} \frac{1}{r} + \mathbf{H}_t \times (\nabla_{\mathbf{x}} \frac{1}{r} \times \mathbf{n}_{\mathbf{x}}) \right] dS_{\mathbf{x}} \quad (4.27)$$

---

<sup>10</sup>Citāda procedūra atrodama [38], kurā būtu jārēķina virsmas liekums integrāļa regularizācijai.

kur  $\mathbf{H}_t := (\mathbf{I} - \mathbf{n} \otimes \mathbf{n})\mathbf{H}$ . Kur tagad viendāojumu varam regularizēt ar formulām (skatīt § A.7 un § A.8)

$$\mathcal{P} \int \frac{\partial}{\partial n_{\mathbf{x}}} \frac{1}{4\pi r} dS_{\mathbf{x}} = \frac{1}{2} \quad \mathcal{P} \int \frac{\mathbf{n} \times \mathbf{r}}{r^3} dS_{\mathbf{x}} = 0 \quad (4.28)$$

un iegūstam integrālformulu vāji singulārā formā

$$H_n(\mathbf{x}_0) = \frac{\mathbf{H}_0 \cdot \mathbf{n}_{\mathbf{x}_0}}{\mu} + \frac{1}{4\pi} \frac{1-\mu}{\mu} \int_{\partial D} n_{\mathbf{x}_0} \cdot \left[ [\mathbf{H}_t(\mathbf{x}) - \mathbf{H}_t(\mathbf{x}_0)] \frac{\partial}{\partial n_{\mathbf{x}}} \frac{1}{r} \right. \\ \left. + [\mathbf{H}_t(\mathbf{x}) - \mathbf{H}_t(\mathbf{x}_0)] \times (\nabla_{\mathbf{x}} \frac{1}{r} \times \mathbf{n}_{\mathbf{x}}) \right] dS_{\mathbf{x}} \quad (4.29)$$

Skaitliski nesingulāros elementus integrējam ar trapezes formulu (*trapezodial rule*), bet vāji singulāros elementus integrējam ar formulu (4.24).

#### 4.3.4 Lauka normālās komponentes aprēķināšanas algoritmu salīdzinājums

Novērtējot iegūto algoritmu skaitlisko klūdu pie dažādām triangulācijām iegūstam tabulu 4.2. Tajā redzam, ka lauka rēķināšanas algoritms § 4.3.1, kas iepriekš izmantots 2D [7], [8] un 3D aksiālsimetriskos aprēķinos [9], [10], strādā slikti saspilstam rotācijas elipsoīdam. Vāji singulāram formulējumam § 4.3.2 arī ir samērā lielas relatīvās klūdas, kuras klūst ievērojami lielākas, kad potenciāls pats ir ar klūdu<sup>11</sup>. Savukārt ieņētā pārrēķināšanas formula (4.29) strādā ievērojami labāk, tādēļ to izmantoju piliena simulācijās.

---

<sup>11</sup>Šīs sadalītas testos izmantojam precīzas ellipsoīda normāles un potenciālu vai lauka tangencialās komponentes.

Metode	$a$	$b$	$c$	$N$	$\langle r \rangle, \%$	$q, \%$	$r_{1/4}, \%$	$r_{1/2}, \%$	$r_{3/4}, \%$
singulārs	1	1	1	150	11.2789	11.2352	11.1987	11.2984	11.3516
izlaižot singulāro elementu § 4.3.1	1	1	1	472	6.5781	6.564	6.5272	6.5772	6.6306
	1	1	1	1828	3.4078	3.3979	3.378	3.4053	3.4263
	2	$\frac{1}{4}$	$\frac{1}{4}$	684	39.7824	45.7148	35.956	37.3057	41.0605
	$\frac{1}{4}$	1	1	408	3.4175	1.8754	0.6912	1.2341	3.9719
	1	1	1	150	1.2398	1.1202	1.0544	1.1442	1.2205
ar adaptīvo kvadratūru § 4.3.1	1	1	1	472	0.4747	0.4718	0.4355	0.4812	0.5063
	1	1	1	1828	0.262	0.2456	0.2346	0.2491	0.2599
	2	$\frac{1}{4}$	$\frac{1}{4}$	684	9.2851	11.3891	5.8845	6.4635	8.1103
	$\frac{1}{4}$	1	1	408	23.6586	4.1127	0.7374	1.1262	2.3453
	1	1	1	150	11.5372	6.1046	5.4386	6.0368	7.1854
vāji singulārs § 4.3.2	1	1	1	472	3.6541	3.4344	2.9087	3.4468	4.0617
	1	1	1	1828	2.7216	1.7262	1.1919	1.7034	2.2521
	2	$\frac{1}{4}$	$\frac{1}{4}$	684	4.7204	6.2412	1.8778	2.7887	4.6461
	$\frac{1}{4}$	1	1	408	11.0894	8.3032	5.0108	6.5863	14.2929
	1	1	1	150	1.5314	1.2548	1.1718	1.2486	1.3227
Bio-Savāra § 4.4	1	1	1	472	0.1464	0.1267	0.0989	0.1199	0.1553
	1	1	1	1828	0.1098	0.0833	0.0717	0.0869	0.0978
	2	$\frac{1}{4}$	$\frac{1}{4}$	684	0.9691	4.0344	0.221	0.3339	0.6823
	$\frac{1}{4}$	1	1	408	3.52	1.4893	0.1942	0.2487	3.4761

Tabula 4.2: Lauka normālās komponentes aprēķināšanas algoritmu salīdzinājums pie  $\mu = 10$ . Elipsoīda pusasis  $a, b, c$ , triangulācijas virsotņu skaits  $N$ , vidējā relatīvā klūda  $\langle r \rangle$  (4.14), relatīvās klūdas raksturojošs lielums  $q$  (4.15), relatīvo klūdu kvantiles  $r_{1/4}, r_{1/2}, r_{3/4}$  (4.16).

## 4.4 Lauka normālās un tangenciālās komponentes aprēķināšanas algoritms

Lai no potenciāla atrastu tangenciālās lauka komponentes, izmantošu galīgās differences. No piliena ārpuses vai iekšpuses magnētiskais lauks ir izsakāms kā skalāra potenciāla gradients  $\nabla\psi$ . Tātad tuvināti varu izteikt potenciāla vērtības ap izvēlētu virsotni  $\mathbf{x}_0$  sekojošā veidā

$$\nabla\psi \cdot (\mathbf{x}_i - \mathbf{x}_0) = \psi(\mathbf{x}_i) - \psi(\mathbf{x}_0) \quad (4.30)$$

Ņemot vēra visas izejošās šķautnes no  $\mathbf{x}_0$ , iegūstam vienādojumu sistēmu, kuru risinām ar mazāko kvadrātu metodi. Tangenciālās lauka komponentes pēc atrisināšanas

varam iegūt projicējot  $\nabla\psi$  uz virsmas pieskarplaknes

$$\mathbf{H}_t = |(\mathbf{I} - \mathbf{n} \otimes \mathbf{n})\nabla\psi| \quad (4.31)$$

Šo metodi tomēr nav iespējams izmantot lauka normālās komponentes aprēķināšanai, jo pēc risināmajiem vienādojumiem potenciāla atvasinājums normāles virzienā piedzīvo lēcienu (skatīt robežnosacījumus (2.7)). Tādēļ tās atrodu pēc iegūtās pārrēķināšanas formulas § 4.3.3. Nemot vērā, ka reālā situācijā normāles vektorus sareķinu pēc § 3.2, bet lauka normālās komponentes rēķinu no iepriekš iegūtajām tangencialajām komponentēm<sup>12</sup>, tad veicu apvienoto algoritmu testu (skatīt tabulu 4.3 un attēlu 4-3).

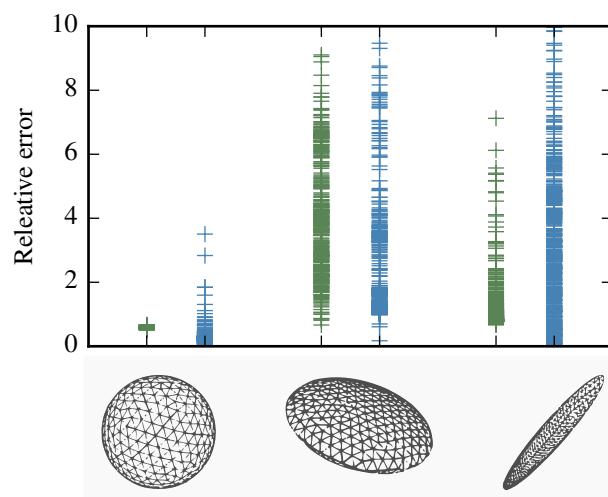
Redzams, ka kļūdas lauka normālajai komponentei ievērojami pieaug, tomēr tā vēl joprojām ir pietiekami maza, kā to redzēsim piliena stiepšanas testā § 5.3. Tomēr dažas relatīvās kļūdas virsotnēs ir lielākas par 10%, kā redzams attēlā 4-3, kuru cēloni būtu nepieciešams izpētīt.

Komponente	$a$	$b$	$c$	$N$	$< r >, \%$	$q, \%$	$r_{1/4}, \%$	$r_{1/2}, \%$	$r_{3/4}, \%$
Tangencialā	1	1	1	150	1.8647	1.87	1.832	1.8633	1.9031
	1	1	1	472	0.5828	0.5832	0.5676	0.5807	0.5954
	1	1	1	1828	0.1478	0.148	0.1406	0.1465	0.1534
	2	$\frac{1}{4}$	$\frac{1}{4}$	684	1.4421	1.4484	0.8047	0.9041	1.1566
	$\frac{1}{4}$	1	1	408	3.9178	4.4132	2.4102	3.6653	5.1118
Normālā	1	1	1	150	0.3181	0.1139	0.0604	0.0995	0.2054
	1	1	1	472	0.286	0.2477	0.215	0.2536	0.278
	1	1	1	1828	0.3317	0.1807	0.1608	0.181	0.2042
	2	$\frac{1}{4}$	$\frac{1}{4}$	684	3.7536	6.5952	1.1116	2.661	4.5408
	$\frac{1}{4}$	1	1	408	6.9656	3.5073	1.2856	1.8911	4.2858

Tabula 4.3: Lauka normālās un tangencialās komponentes aprēķināšana algoritmus apvienojot pie  $\mu = 10$ . Elipsoīda pusasis  $a, b, c$ , triangulācijas virsotņu skaits  $N$ , vidējā relatīvā kļūda  $< r >$  (4.14), relatīvās kļūdas raksturojošs lielums  $q$  (4.15), relatīvo kļūdu kvantiles  $r_{1/4}, r_{1/2}, r_{3/4}$  (4.16).

---

<sup>12</sup>Iepriekšējos testos lietotas precīzas ellipsoīda normāles un lauka sadalījums uz virsmas



Att. 4-3: Lauka normālās un tangenciālās komponentes relatīvā kļūda (%) katrā triangulācijas virsotnē apvienotajā algoritmā pie  $\mu = 10$ . Zilais stabīņš ir relatīvās kļūdas tangenciālajām un zaļais normālajām lauka komponentēm. Kā testa piemēri izmantota sfēras triangulācija ar virsotņu skaitu  $N = 472$ , saspiests rotācijas elipsoīds  $a, b, c = 1/4, 1, 1$  ar  $N = 408$  un izstiepts  $a, b, c = 2, 1/4, 1/4$  ar  $N = 684$ .

## Nodaļa 5

# Virsmas ātruma aprēķināšana ar robežintegrālvienādojumu metodi

Zinot uz virsmu iedarbojušos spēka normālo komponenti, varam aprēķināt viskozā piliena dinamiku pēc Stoksa robežintegrālvienādojuma. Tā skaitiskai izmantošanai veicu regularizāciju un ar ekvivalentiem pārveidojumiem atbrīvojos no virsmas liekuma tiešas reķināšanas. Simulācijas algoritmu pārbaudu ar elipsoīda relaksācijas dinamiku uz sfēru kapilāro spēku darbības rezultātā un ar līdzvara figūru aprēķinu konstantā laukā.

### 5.1 Stoksa robežintegrālvienādojuma regularizācija un pārveidošana bezliekumu formā

Kā iepriekš jau redzējām § 2.3, virsmas ātrumu pilienam varam izteikt ar robežintegrālvienādojumu (2.25)

$$\mathbf{u}(\mathbf{x}_0) = \frac{1}{8\pi\eta} \int_{\partial D} \Delta f \left( \frac{\mathbf{I}}{r} + \frac{\mathbf{r} \otimes \mathbf{r}}{r^3} \right) \cdot \mathbf{n}_{\mathbf{x}} dS_{\mathbf{x}} \quad \mathbf{r} = \mathbf{x} - \mathbf{x}_0 \quad (5.1)$$

kura izmantošanai ir jāzin tikai virsmas spēks normāles virzienā, kas šajā gadījumā ir

$$\Delta f = -\nabla \cdot \mathbf{n} + \frac{\mu(\mu-1)}{8\pi} H_n^2 + \frac{\mu-1}{8\pi} H_t^2 \quad (5.2)$$

Lai to aprēķinātu, ir jāzina magnētiskā lauka intensitāte uz virsmas (tuvojoties no piliena iekšpusēs) un virsmas vidējais liekums  $k_m = \frac{1}{2}\nabla \cdot \mathbf{n}$  (skatīt § 4.1), ko aprēķinām ar lokālu paraboloida piedziņanu (skatīt § 3.2).

Ērtākai skaitliskai integrēšanai atbrīvosimies no singulāro integrāļu rēķināšanas veicot regularizāciju. Izmantosim sekojošu identitāti (skatīt § A.6)

$$\int \left( \frac{\mathbf{I}}{r} + \frac{\mathbf{r} \otimes \mathbf{r}}{r^3} \right) \cdot \mathbf{n}_x dS_x = 0 \quad (5.3)$$

kura ļauj integrālvienādojumu pārrakstīt regulārā formā<sup>1</sup>

$$\mathbf{u}(\mathbf{x}_0) = \frac{1}{8\pi\eta} \int_{\partial D} (\Delta f_{\mathbf{x}} - \Delta f_{\mathbf{x}_0}) \left( \frac{\mathbf{I}}{r} + \frac{\mathbf{r} \otimes \mathbf{r}}{r^3} \right) \cdot \mathbf{n}_x dS_x \quad (5.4)$$

Šo integrālvienādojumu literatūrā [17], [16], [18] integrē ar trapeces integrēšanas formulu (*trapezodial rule*) tāpat kā to jau darījām ar magnētiskā lauka aprēķinu § 4.1.

Lai padarītu algoritmu robustāku un samazinātu degenerētas triangulācijas ietekmi, atbrīvosimies no virsmas liekuma tiešas rēķināšanas, kas nosaka spraiguma ietekmi. Izmantosim sekojošu identitāti (skatīt § A.5)

$$\begin{aligned} - \int \nabla \cdot \mathbf{n}_x \mathbf{n}_{x_0} \cdot \left( \frac{\mathbf{I}}{r} + \frac{\mathbf{r} \otimes \mathbf{r}}{r^3} \right) \cdot \mathbf{n}_x dS_x &= \\ &= \int \frac{\mathbf{r} \cdot [\mathbf{n}_x + \mathbf{n}_{x_0}]}{r^3} \left( 1 - \frac{3(\mathbf{r} \cdot \mathbf{n}_x)(\mathbf{r} \cdot \mathbf{n}_{x_0})}{r^2} \right) dS_x \end{aligned} \quad (5.5)$$

---

<sup>1</sup>Šo singularitāti var integrēt arī skaitliski (skatīt § 4.3.2), bet šādi izskaitot algoritms ievērojami vienkāršojas.

ko izmantojot varam rakstīt robežintegrālvienādojumu ar magnētisko lauku

$$u_n(\mathbf{x}_0) = \frac{1}{8\pi\eta} \int_{\partial D} (\Delta \tilde{f}_{\mathbf{x}} - \Delta \tilde{f}_{\mathbf{x}_0}) \mathbf{n}_{\mathbf{x}_0} \cdot \left( \frac{\mathbf{I}}{r} + \frac{\mathbf{r} \otimes \mathbf{r}}{r^3} \right) \cdot \mathbf{n}_{\mathbf{x}} dS_{\mathbf{x}} \\ + \frac{\gamma}{8\pi\eta} \int_{\partial D} \frac{\mathbf{r} \cdot [\mathbf{n}_{\mathbf{x}} + \mathbf{n}_{\mathbf{x}_0}]}{r^3} \left( 1 - \frac{3(\mathbf{r} \cdot \mathbf{n}_{\mathbf{x}})(\mathbf{r} \cdot \mathbf{n}_{\mathbf{x}_0})}{r^2} \right) dS_{\mathbf{x}} \quad (5.6)$$

kur  $\Delta \tilde{f}$  ir magnētiskā lauka radītais spēks normāles virzienā

$$\Delta \tilde{f} = \frac{\mu(\mu-1)}{8\pi} H_n^2 + \frac{\mu-1}{8\pi} H_t^2 \quad (5.7)$$

bet  $u_n$  ir ātrums virsmas normāles virzienā. Abi integrāli ir regularizēti, kuri singulārā punktā dod 0 ieguldījumu, tādēļ to skaitliskai integrēšanai atkal varam izmantot trapeces integrēšanas formulu.

## 5.2 Relaksācija par sfēru

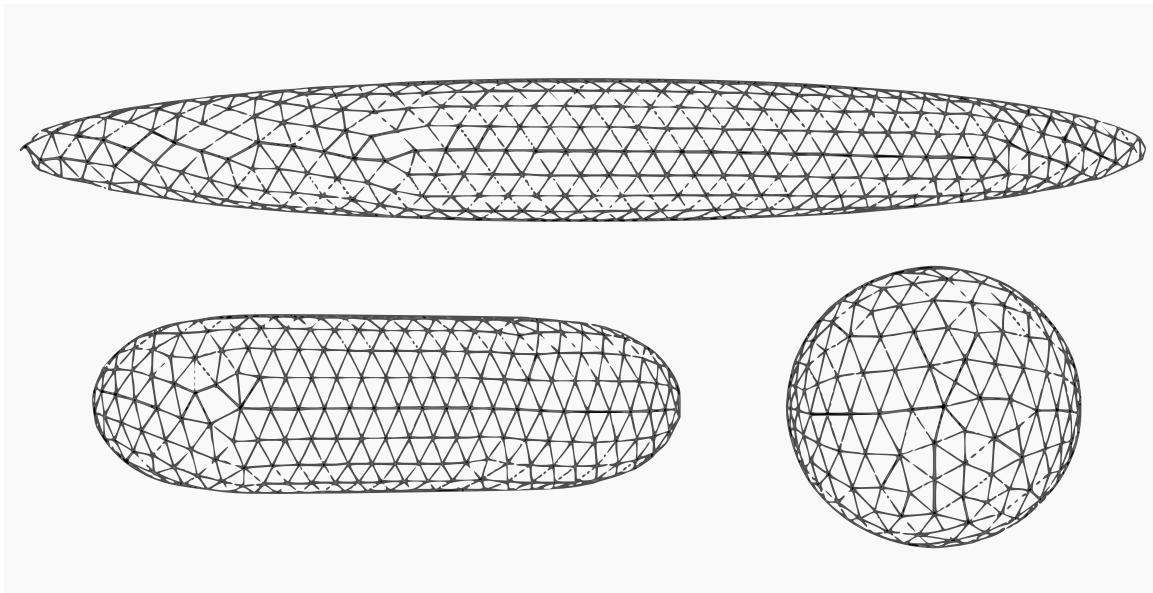
Virsmas spraiguma dēļ piliens bez ārēja uzlikta lauka cenšas ieņemt sfērisku formu. Laujot relaksēt loti izstieptam ellipsoīdam, redzamā dinamika kvalitatīvi atbilst [39] (skatīt attēlu 5-1). Kvantitatīvi mazām deformācijām pieņemot, ka piliens ieņem tikai elipsoīda formu visā relaksācijas procesā, ir iegūstama šāda formula [39]

$$D = D_0 e^{-t/\tau} \quad \tau = \frac{R_0(16\eta_2 + 19\eta_1)(3\eta_2 + 2\eta_1)}{40\gamma(\eta_1 + \eta_2)} \quad (5.8)$$

kur  $D$  ir deformētība definēta kā pusasu starpības attiecība pret pusasu summu, bet  $R_0$  ir piliena sfēras rādiuss relaksētā stāvoklī. Apskatot gadījumu, kad  $\eta_1^{(e)} = \eta_2^{(i)} = \eta$  un kad sākotnējais stāvoklis ir elipsoīds  $R_0 = (abc)^{1/3}$ , tad relaksācijas laiks vienkāršojās par

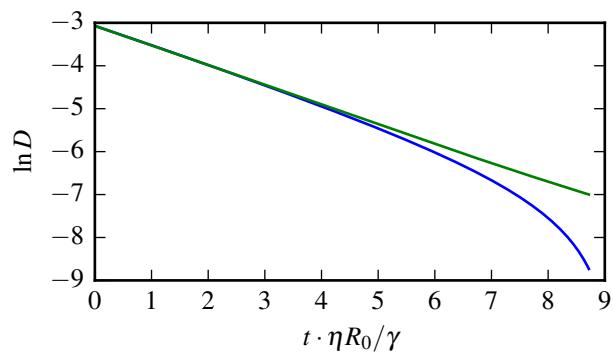
$$\tau = \frac{35\eta}{16\gamma} (abc)^{1/3} \quad (5.9)$$

Lai iegūtu relaksācijas dinamikas aprēķinu, varam izmantot formulējumu (5.4) vai bezliekuma formulējumu ( $\Delta \tilde{f} = 0$ ) (5.6). Par sākumnosacījumu izmantojot elip-



Att. 5-1: ļoti izstiepta piliena relaksācijas simulācija. Laikā  $t = 0$  ir elipsoīds ar pusasīm  $a, b, c = 2, 1/4, 1/4$  (augšā), kas kapilāro spēku dēļ relaksē un ieņem cigāra formu laikā  $t = 2\frac{\eta R_0}{\gamma}$  (apakšā pa kreisi) un tuvu sfēriskai formai laikā  $t = 5\frac{\eta R_0}{\gamma}$  (apakšā pa labi).

soīdu ar parametriem  $a = 1.1, b = 1, c = 1$  un veicot skaitlisku simulāciju, iegūstu deformētības  $D$  atkarību no laika  $t$ , kas redzams attēlā 5-2 abiem formulējumiem.



Att. 5-2: Maz izstiepta elipsoīda deformētības atkarība no laika relaksējot. Sākotnējā triangulācija ar  $a, b, c = 1.1, 1, 1$  un  $N = 480$ . Zilā līnija ir skaitliska simulācija, kura izmanto formulējumu (5.4), bet zaļā līnija izmanto bezliekumu formulējumu (5.6).

Sākotnēji abas simulācijas relaksē par sfēru eksponenciāli ar slīpuma koeficientu  $-35/16$ , kā tas bija sagaidāms no analītiskā rezultāta, tādēļ abi formulējumi dod ticamus skaitliskus rezultātus un var tikt izmantoti simulācijās bez ārēja uzlikta lau-

ka. Bet pie ilgākām simulācijām bezliekuma formulējums strādā labāk, kurš paliek eksponenciāls. Tam par iemeslu varētu būt neprecīzā liekuma sarēķināšana § 3.2.

### 5.3 Līdzvara figūras konstantā laukā

Konstantā ārējā magnētiskā laukā sfēriski piliens izstiepjas ieņemot līdzvara stāvokli. Pienemot, ka piliens pieņem elipsoīda formu, tad pēc enerģijas minimuma principa iegūst sekojošu parametrisku formulu elipsoīda pusasu attiecībai un ārējam magnētiskajam laukam [5] (skatīt arī § 6.2.3 līdzīgam izvedumam)

$$a/b = K^{-1} \quad H_0^2 R / \gamma = g(K) \quad (5.10)$$

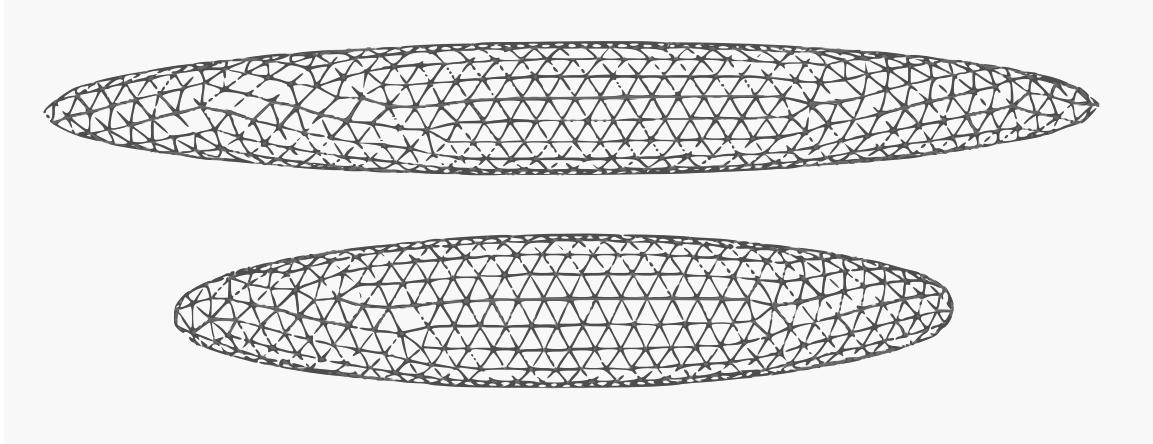
$$\varepsilon = (1 - K^2)^{1/2} \quad n = K^2(-2\varepsilon + \frac{1}{2\varepsilon^3} \log \frac{1+\varepsilon}{1-\varepsilon}) \quad \alpha = \frac{1}{\mu-1}$$

$$g(K) = \frac{8\pi}{\mu} (n + \alpha)^2 \varepsilon^2 K^{-4/3} (1 + 2K^2 + (1 - 4K^2)\varepsilon^{-1} K^{-1} \arcsin \varepsilon) \\ \times \left( -6 + (2 + K^2)\varepsilon^{-1} \log \frac{1+\varepsilon}{1-\varepsilon} \right)^{-1}$$

kura pārbaudīta arī aksialsimetriskos aprēķinos [11], ka strādā labi zem pusasu attiecības  $a/b < 7$ .

Skaitliskajai simulācijai izmantojam bezliekumu formulējumu (5.6) un magnētisko lauku uz virsmas rēķinām tāpat kā § 4.4. Ātruma lauku integrēju pēc Runges-Kutas otrās kārtas shēmas kā to jau apskatīju § 3.3. Lai uzturētu blīvāku triangulāciju pie elipsoīda smailēm, par sākotnējo triangulāciju izvēlos elipsoīdu ar pusasu parametriem  $a = 2, b = 1/4, c = 1/4$ , kurai ļauju sarauties ar ārēji uzliktu lauku, kā tas redzams attēlā 5-3.

Lai iegūtu līdzvara figūras deformāciju, aprēķins tiek apstādināts pie galīga laika  $t_0$ , kad redzamas izmaiņas vairs nenotiek. Tas ienes klūdu, kuru varam novērtēt zinot cik ātri notiek parametru izmaiņas, kad veicam apstādināšanu. Tomēr zinot,



Att. 5-3: Piemērs elipsoīda relaksācijai ar ārēju uzliktu lauku pie  $\mu = 10$ .

ka nomērītie lielumi (deformācija  $D$  vai pusasu attiecība  $a/b$ ) tiecās uz konstantu vērtību, varam kļūdu novērtēt ar asymptotisko izvirzījumu.

Apzīmēsim ar  $f(t)$  interesējošos parametrus, piemēram, pusasu attiecību  $a/b$  vai deformāciju  $D$ . Pēc asymptotiskā izvirzījuma varam funkcijas vērtību aprēķināt ar sekojošu formulu

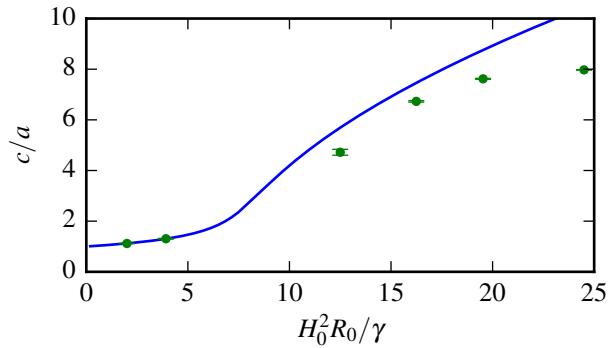
$$f(t) = f(t_0) - t_0^2 \frac{df}{dt} \Big|_{t_0} \left( \frac{1}{t} - \frac{1}{t_0} \right) + O(t^{-2}) \quad (5.11)$$

Tiecinot  $t \rightarrow \infty$  iegūstam

$$f(t \rightarrow \infty) - f(t_0) = t_0 \frac{df}{dt} \Big|_{t_0} \quad (5.12)$$

kur kreisā puse ir kļūda, kura ieviesusies novērtētajos parametros dēļ galīgā atvēlētā simulācijas laika  $t_0$ . Izvēloties dažādas magnētiskā lauka vērtības, novērtēju  $t_0$ , kad vizuālas izmaiņas vairs aprēķinā nenotiek. Tālāk novērtēju pusasu attiecību  $c/a$  un tā kļūdu ar formulu (5.12), kuras redzamas attēlā 5-4 ar teorētisko līkni (5.11).

Redzams, ka kļūdas robežās nosimulētās līdzsvara figūru pusasu attiecības atbilst analītiskajam rezultātam elipsoīdam, tātad skaitliskais algoritms ir izmantojams magnētiskā šķidruma piliena simulācijai ārējā laukā. Tomēr kļūdu intervāliem ir tendence pieaugt pie lielāka lauka, kur cēlonis varētu būt magnētiskā lauka aprēķināšanas kļū-



Att. 5-4: Pusasu attiecības atkarība no Bonda skaitļa pie  $\mu = 10$ . Līkne atbilst rotācijas elipsoīda piliena pusasu attiecībai (5.11), kas labi spēj aprakstīt eksperimentu pusasu attiecībai zem 7 [11].

da.<sup>2</sup> Tomēr tās arī varētu būt sekas tam, ka piliens vairs nav uzskatāms par rotācijas elipsoīdu [11].

---

<sup>2</sup>Tā mazinās līdz ar triangulācijas smalkumu, ko pārbaudījis neesmu jo aprēķini būtu jāveic dienām.

# Nodaļa 6

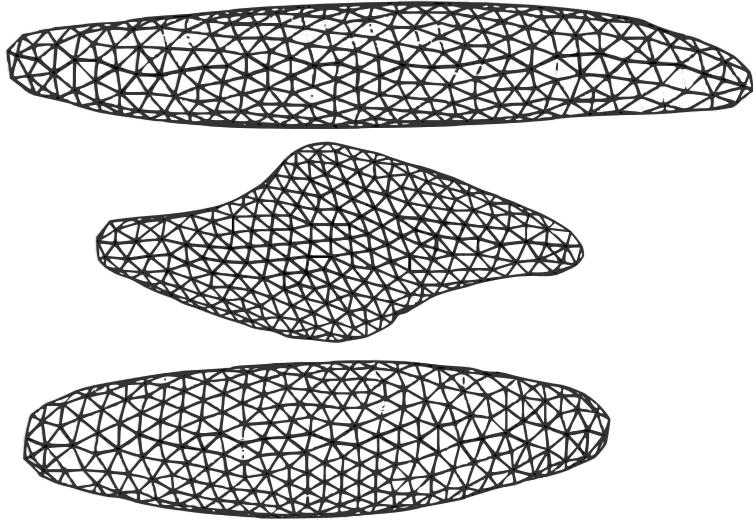
## Simulācijas ar rotējošu lauku

Šajā nodaļā aplūkosim magnētiskā šķidruma pilienu ievietotu rotējošā laukā. Kvalitatīvi apskatīsim pāreju no lēna uz ātru lauku, kā arī atkarību no lauka intensitātes. Ātra mazas intensitātes lauka gadījumā iegūsim analītisku formulu saspieduma atkarībai no lauka intensitātes, ar ko pārbaudīsim simulāciju. Bet lielā laukā demonstrēšu plakantārpa un zvaigznes līdzvara figūras iegūšanu.

### 6.1 Līdzvara figūru atkarība no frekvences

Zemas frekvences gadījumā (sinhronais režīms) piliens paspēj izsekot magnētiskā lauka virzienam, kura forma nav aksiālsimetriska, jo tā gali noliecās (skatīt attēlu 6-1). Pie lielākas lauka frekvences piliens vairs laukam izsekot nespēj un veic svārstības rotējošā lauka plaknē (asinhronais režīms). Tomēr vidēji laikā šīs svārstības rezultējās rotācijā, kas ir rotējošā lauka virziena. Novērojams, ka arī šajā gadījumā līdzvara figūras gali noliecās (skatīt vidējo simulāciju attēlā 6-1)

Vēl lielākas frekvences gadījumā piliens iegūst aksiālsimetrisku formu (attēla 6-1 apakšējā simulācija) un vairs nerotē un nesvārstās. To tad darbā saucu par augstfrekvences lauku, kas no simulāciju sērijas pie Bonda skaitļa 35 novērtējams ar frekvenci, kura pārsniedz  $8\eta R_0/\gamma$ .



Att. 6-1: Magnētiskā šķidruma piliena formas atkarība no frekvences pie Bonda skaita 35 un pie  $\mu = 10$ . Pirmais attēls atbilst situācijai, kad piliens spēj izsekot lauka rotācijai ar frekvenci  $\omega = 0.5\eta R_0/\gamma$ , otrs attēls, kad lauks apsteidz piliena rotācijas ātrumu ( $\omega = 1\eta R_0/\gamma$ ) un ir novērojamas svārstības. Pēdējais gadījums atbilst pilienam augstfrekvences gadījumā  $\omega >> 8\eta R_0/\gamma$ .

## 6.2 Rotējošs augstfrekvences lauks

### 6.2.1 Skaitliska metode augstfrekvences lauka simulācijām

Augstfrekvences laukā deformācijas viena perioda laikā klūst mazākas palielinot lauka frekvenci. Tātad varam vienkāršoti uzskatīt, ka deformācijas izsauc lauka vidējās vērtības un varam veikt vidējošanu pa periodu vienādojumiem (2.25) un (2.36)

$$\frac{d \langle \mathbf{x}_0 \rangle}{dt} = \frac{1}{8\pi\eta} \int_{\partial D} G_{ji} n_i \left( \gamma \nabla \cdot \mathbf{n} - \frac{\mu(\mu-1)}{8\pi} \langle H_n^2 \rangle - \frac{\mu-1}{8\pi} \langle H_t^2 \rangle \right) dS_{\mathbf{x}} \quad (6.1)$$

kur  $\langle H_n^2 \rangle$  un  $\langle H_t^2 \rangle$  ir lauka vidējās vērtības.

Lai atrastu lauka tangenciālās un normālās komponentes ātri rotējošā laukā atceņsimies, ka vēlamies atrisināt Laplasa vienādojumu (2.6), bet šoreiz ar robežnosacī-

jumu bezgalībā

$$\psi(\mathbf{x} \rightarrow \infty) = \mathbf{H}_0 \cdot \mathbf{x} = H_0 x \cos t + H_0 y \sin t \quad (6.2)$$

Nemot vērā, ka viena perioda laikā piliena virsma mainās maz, tad varam atrisinājumu meklēt formā

$$\psi(\mathbf{x}_0, t) = \psi_x(\mathbf{x}_0) \cos t + \psi_y(\mathbf{x}_0) \sin t \quad (6.3)$$

Ievietojot šo formu vienādojumos (2.6) un (2.7), bet ar robežnosacījumu bezgalībā (6.2) iegūst, ka  $\psi_x$  ir atrisinājums tiem pašiem vienādojumiem ar uzliktu lauku bezgalībā  $\mathbf{H}_0 = H_0 \hat{\mathbf{x}}$  un tāpat  $\psi_y$  ir atrisinājums ar lauku  $\mathbf{H}_0 = H_0 \hat{\mathbf{y}}$ . Tas ļauj izmantot bez modificēšanas jau iepriekš iegūtos algoritmus, kuri rēķina lauka normālās un tangenciālās komponentes uz virsmas.

Lauka normālās komponentes pie rotējoša lauka uz virsmas pēc iepriekš teiktā tad var izteikt šādi

$$H_n(\mathbf{x}_0, t) = H_{nx}(\mathbf{x}_0) \cos t + H_{ny}(\mathbf{x}_0) \sin t \quad (6.4)$$

Vidējojot pa periodu kvadrāta vidējo vērtību

$$\langle H_n^2 \rangle = \frac{H_{nx}^2 + H_{ny}^2}{2} \quad (6.5)$$

Un tāpat kvadrāta vidējā vērtība izsakāma arī lauka tangenciālajai komponentei

$$\langle H_t^2 \rangle = \frac{H_{tx}^2 + H_{ty}^2}{2} \quad (6.6)$$

### 6.2.2 Skaitliskas simulācijas augstfrekvences laukā

Mazas lauka intensitātes gadījumā piliens saplok par rotācijas figūru (turpmāk sauktu par pankūku), kā tas redzams attēla 6-2 pirmajā rindā. Tomēr pie vēl lielāka lauka pankūka klūst nestabila un pārtop par trīsasu figūru (turpmāk sauksu par pla-

kantārpu) (attēla 6-2 2. rinda). Savukārt pie Bonda skaitļa, kas lielāks par  $Bm > 50$  (pie  $\mu = 10$ ) pankūka pārtop nevis par plakantārpu, bet kļūst par jūraszvaigzni<sup>1</sup> (skatīt attēlu 6-5), kas arī ir novērots eksperimentāli [3], [2] un tā parādot spontāno simetrijas laušanu rotācijas figūrai un skaitliskā algoritma efektivitāti.<sup>2</sup>

No simulācijām dažādiem magnētiskā Bonda skaitļiem novēroju, ka tārpa nestabilitāte no pankūkas aug, ja magnētiskais Bonda skaitlis ir lielāks par 20 (pie  $\mu = 10$ ), kura redzama attēlā 6-3. Šī figūra tuvināti var tikt uzskatīta par trīsasu elipsoīdu, jo tā divas projekcijas atbilst elipsēm, bet pretskatā ir mazas novirzes no tās. Tomēr, lai saprastu, cik šīs novirzes ir mazas, būtu jāsarēķina elipsoīda energija, ko šajā darbā darījis neesmu.

Ja par sākotnējo triangulāciju izvēlās elipsoīdu un uzliek lielu augstfrekvences lauku, tad līdzvara figūra ieņem plakantārpa nevis zvaigznes formu (skatīt attēlu 6-4), tātad iespējams pastāv histerēze. Tomēr pastāv iespējamība, ka šī simulācija no tārpa nesabruk par zvaigzni, jo triangulācija ir pārāk rupja un jo aktīvās stabilizācijas algoritms § 3.3.3 veic virsotņu pozīciju relaksēšanu, kas neļauj nestabilitātēm augt.

### 6.2.3 Enerģijas minimums aksiālsimetriskam elipsoīdam

Simulācijās redzam, ka ātri rotējošā laukā piliens tiecās ieņemt rotācijas elipsoīda formu. Šajā stāvoklī pilienam ir minimāla virsmas spraiguma un magnetostatiskās energijas summa. Virsmas energija ir proporcionāla virsmas laukumam, bet visas telpas magnētiskā lauka energijas izmaiņa tajā ievietojot magnētisko materiālu palielinās par [40]

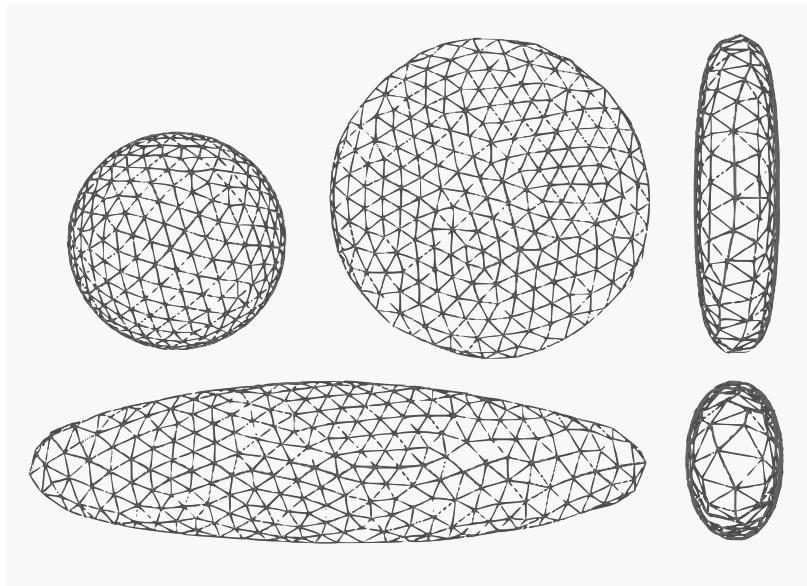
$$E_M = \frac{1}{8\pi} \int_D (1 - \mu) \mathbf{H} \cdot \mathbf{H}_0 dV \quad (6.7)$$

kur integrēšana ir pa ievietotā ķermēna tilpumu un kur  $\mathbf{H}_0$  un  $\mathbf{H}$  ir magnētiskais lauks bez un ar magnētisko pilienu.

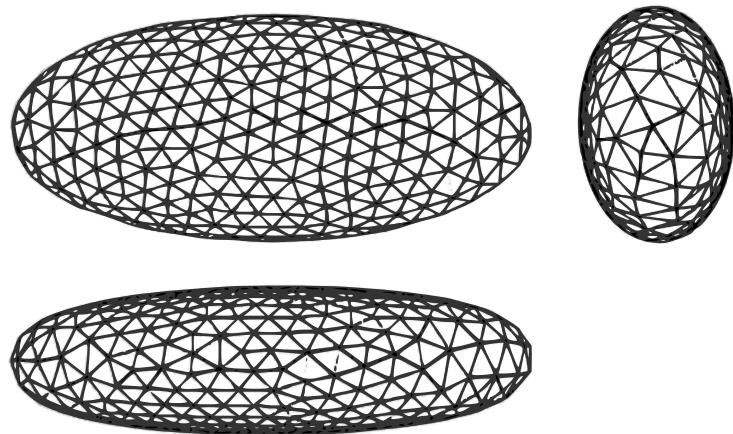
---

<sup>1</sup>Kuru ieguvis arī esmu dinamikas simulācijā pie  $\omega = 30\eta R_0/\gamma$ .

<sup>2</sup>Šīs simulācijas vēl nav sasniegušas savu līdzvara stāvokli, kuras šajā darbā ieguvis neesmu, jo simulācijas ir pārāk lēnas. Katra no tām ilgst vairāk par 24h. Ievērojamu ātruma pieaugumu varētu iegūt izmantojot adaptīvu laika soli un pārrakstot robežintegrālvienādojumu rēķināšanu, kā paralēlu algoritmu (skatīt § D.5).

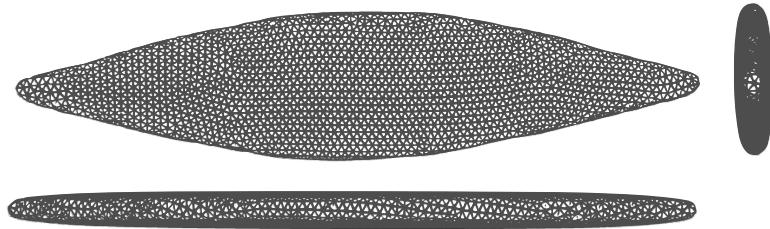


Att. 6-2: Magnētiskā šķidruma piliena dinamika augstfrekvences laukā pie magnētiskā Bonda skaitļa 35 un pie  $\mu = 10$ . Pa kreisi ir sākotnējā triangulācija, kura uzliktā laukā saplok par pankūku (attēls pa vidu un sānskats pa labi). Ja lauks ir pietiekami spēcīgs (kā tas ir šajā gadījumā) pankūka sabrūk par plakantārpu (attēls apakšā ar sānskatu pa labi).

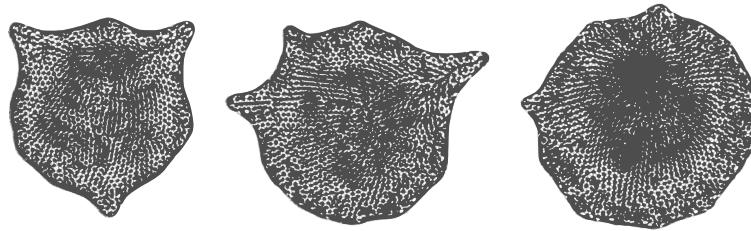


Att. 6-3: Magnētiskā šķidruma piliena līdzsvara figūra augstfrekvences laukā pie Bonda skaitļa 22.3. un pie  $\mu = 10$ .

Ja ar  $a, b, c$  apzīmēsim pusasu garumus un divas no pusasīm ir vienādas  $a = b$ , bet



Att. 6-4: Magnētiskā šķidruma piliena līdzsvara figūra augstfrekvences laukā pie Bonda skaitļa 72 un pie  $\mu = 10$ . Kā sākotnējā triangulācija izvēlēta šim gadījumam ir izstiepts elipsoīds ar parametriem  $a, b, c = 2, 1/4, 1/4$ .



Att. 6-5: Magnētiskā šķidruma piliena zvaigžņu simulācijas augstfrekvences laukā pie  $\mu = 10$ . Pa kreisi ar magnētisko Bonda skaitli 50.4 vidū 68.6 un pa labi 89.6.

pusasu attiecību apzīmēsim ar  $K = a/c$ , tad virsmas laukumam ir analītiska formula

$$c < a \quad S = 2\pi a^2 \left( 1 - \frac{1}{2\varepsilon K^2} \log \left( \frac{1+\varepsilon}{1-\varepsilon} \right) \right) \quad \varepsilon = \sqrt{1 - 1/K^2} \quad (6.8)$$

$$c > a \quad S = 2\pi c^2 \left( 1 + \frac{K}{\varepsilon} \sin^{-1} \varepsilon \right) \quad \varepsilon = \sqrt{1 - K^2} \quad (6.9)$$

Bet elipsoīda lauks ir izsakāms pēc formulas

$$H_i = \frac{H_{0i}}{1 + (1 - \mu)n_i} \quad (6.10)$$

kur  $n_i$  ir demagnetizācijas koeficients. Elipsoīdam, kuram divi pusasu garumi ir vienādi  $a = b$ , demagnetizācijas koeficients ir izsakāms ar sekojošu formulu [40]

$$c < a \quad n_z = \frac{1 + \varepsilon^2}{\varepsilon^3} (\varepsilon - \tan^{-1} \varepsilon) \quad \varepsilon = \sqrt{K^2 - 1} \quad (6.11)$$

$$c > a \quad n_z = \frac{K^2}{2\varepsilon^3} (-2\varepsilon + \log \left( \frac{1+\varepsilon}{1-\varepsilon} \right)) \quad \varepsilon = \sqrt{1 - K^2} \quad (6.12)$$

Pārējie demagnetizācijas koeficienti izriet no  $n_x + n_y + n_z = 1$  un simetrijas īpašībām.

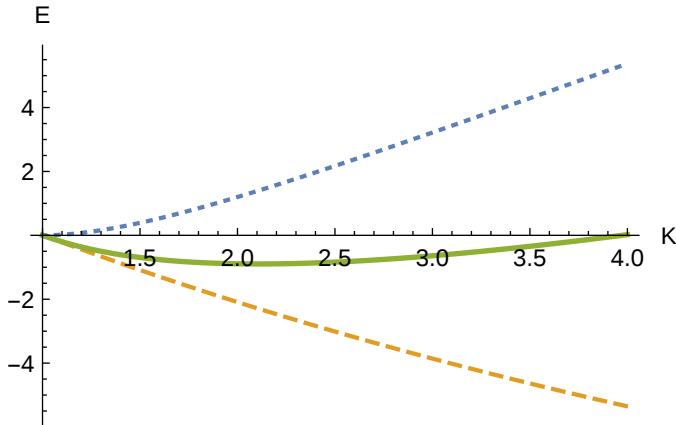
Ja magnētiskais lauks rotē  $xy$  plaknē, lauks elipsoīda iekšpusē ir

$$H_x = \frac{H_0}{1 + (\mu - 1)(1 - n_z)/2} \quad (6.13)$$

kur pieņemam, ka apkārtējai videi nav magnētisko īpašību. Ievietojot šo sakarību formulā (6.7) iegūstam

$$E_M = -\frac{VH_0^2}{8\pi} \frac{1}{8\pi} \frac{1}{\alpha + (1 - n_z)/2} \quad \alpha = \frac{1}{\mu - 1} \quad (6.14)$$

kur  $V = 4/3\pi R_0^3$  ir piliens tilpums. Virsmas un magnētiskās energijas ieguldījums atkarībā no elipsoīda pusasu attiecības pie Bonda skaitļa 15 redzams attēlā 6-6.



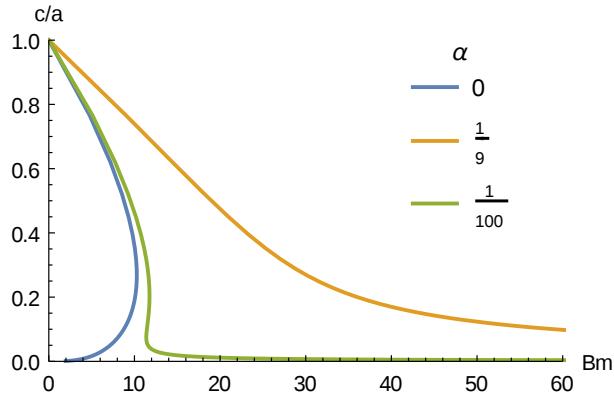
Att. 6-6: Piliens enerģijas atkarība saspieduma pie magnētiskā Bonda skaitļa 15 un pie  $\mu = 10$ . Ar sīki raustīto līniju attēlota virsmas enerģija, ar rupji raustītu līniju magnetizācijas enerģija un ar zaļu līniju summārā enerģija.

Minimizējot pilno enerģiju  $E_T = E_M + E_S$  pie konstanta tilpuma attiecībā pret pusasu attiecību  $K$  iegūstam sekojošu parametrisku izteiksmi

$$\frac{H_0^2 R_0}{\gamma} = \pi \frac{(\varepsilon(-1 + 2\alpha\varepsilon^2) + K^2 \tan^{-1} \varepsilon)^2 (2\varepsilon K(1 + 2K^2) + (1 - 4K^2) \log \frac{K+\varepsilon}{K-\varepsilon})}{K^{7/3} \varepsilon^4 (-3\varepsilon + (2 + K^2) \tan^{-1} \varepsilon)} \quad (6.15)$$

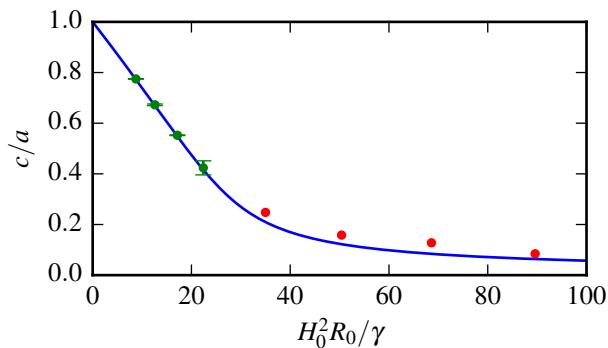
kura pie dažādām  $\alpha$  vērtībām ir redzama attēlā 6-7.

Iegūtos rezultātus izmantoju skaitliskās simulācijas pārbaudei pie  $\mu = 10$  (skatīt



Att. 6-7: Minimālās enerģijas pusasu attiecības atkarība no magnētiskā Bonda skaitļa pie  $\mu = 10$ .

attēlu 6-8). Pie maziem Bonda skaitļiem piliens atrodas līdzsvarā un tam ir aksīlsimetriskā forma (zali punkti grafikā). Bet pie lielākiem laukiem piliens vispirms ieņem aksīlsimetrisku formu (pankūku), no kurās tas spontāni sabrūk par trīsas figūru (plakantārpu). Šo labilo rotācijas figūras stāvokli attēlā 6-8 esmu atlicis ar sarkaniem punktiem. Lielā Bonda skaitļu diapazonā redzama laba atbilstība starp skaitlisko simulāciju un iegūto analītisko rezultātu, kas vēlreiz apstiprina skaitliskā algoritma precīzitāti.

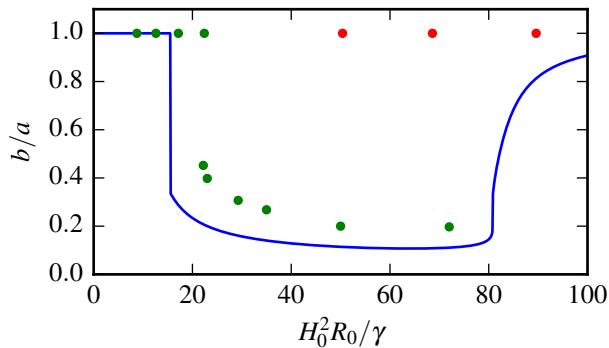


Att. 6-8: Magnētiskā šķidruma piliena skaitliskās simulācijas rotācijas pusasu atkarība no Bonda skaitļa pie  $\mu = 10$ . Ar zaliem punktiem ir atzīmētas tās simulācijas, kurās rotācijas figūra ir stabila. Sarkanie punkti ir tie gadījumi, kad divas rotācijas figūra tālāk sabrūk tālāk, lai ieņemtu plakantārpa formu (kā tas ir redzams 6-2).

#### 6.2.4 Enerģijas minimums trīsasu elipsoīdam

Pie magnētiskā Bonda skaitļa  $B_m < 20$  (pie  $\mu = 10$ ) magnētiskā šķidruma piliens augstfrekvences laukā saspiežas, ko teicami apraksta rotācijas elipsoīda saspiešanās (skatīt attēlu 6-8). Pie lielākiem Bonda skaitļiem notiek spontānā simetrijas laušana pankūkai uz plakantārpu (skatīt attēlus 6-2, 6-3), bet aptuveni no Bonda skaitļa  $B_m > 50$  pankūka sabrūk par jūraszvaigzni (skatīt 6-5). Tomēr, ja par sākotnējo triangulāciju izvēlās elipsoīdu, tad pie laukiem  $B_m > 50$  šis rotācijas elipsoīds pārvēršas par plakantārpu.

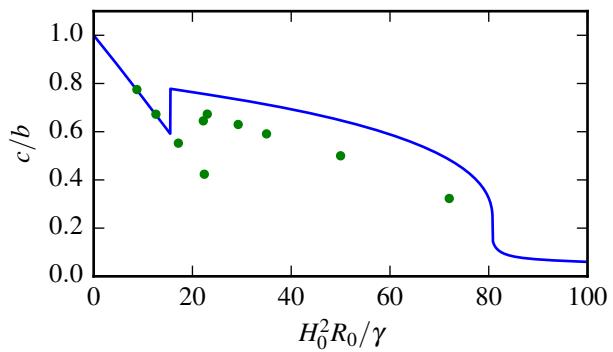
Līdzīgi kā to darījām iepriekšējā sadaļā rotācijas elipsoīdam var apskatīt vispārīgu trīsasu elipsoīda enerģijas minimumu, kas darīts bakalaura darbā [6]. Simetrijas asis šajā gadījumā iepriekš nav definētas, jo spontānā simetrijas laušana pankūkai notiek patvalīgā virzienā lauka rotācijas plaknē. Tādēļ šos virzienus un pusas garumus nācās novērtēt katrai simulācijai individuāli<sup>3</sup>, kuri atlikti attēlā 6-9 un 6-10.



Att. 6-9: Magnētiskā šķidruma piliena skaitlisko simulāciju pusas attiecības lauka rotācijas plaknē atkarība no Bonda skaitļa pie  $\mu = 10$ . Ar sarkaniem punktiem atzīmētas zvaigžņu simulācijas, kurām sākotnējā triangulācija ir izvēlēta par sfēru. Ar zilu līniju ir pusas attiecību, kura būtu sagaidāma trīsasu elipsoīda formas pilienam [6].

Atšķirības cēlonis galvenokārt varētu būt piliena formas neatbilstība trīsasu elipsoīdam, ko visizteiktāk redzam sānskata (kā piemēru skatīt attēla 6-3 apakšu). Bet cik šī atšķirība ir liela, būtu novērtejama aprēķinot simulācijas formas un atbilstošā elipsoīda enerģiju, kas ir ārpus šī maģistra darba rāmjiem.

<sup>3</sup>Ar zīmuli un lineālu



Att. 6-10: Magnētiskā šķidruma piliena skaitlisko simulāciju pusasu attiecības perpendikulāri lauka rotācijas plaknei atkarība no magnētiskā Bonda skaitla pie  $\mu = 10$ . Ar zilu līniju ir pusasu attiecība ( $c$  ir perpendikulārs magnētiskā lauka rotācijas plaknei), kura būtu sagaidāma trīsasu elipsoīda formas pilienam [6].

# Nodaļa 7

## Secinājumi

- Efektīvai magnētiskā vai elektriskā lauka aprēķināšanai nevar izmantot analogus robežintegrālvienādojumus, kuri iepriekš izmantoti 2D un aksiālsimetriskā 3D gadījumā. Salīdzinot parādīts, ka vislabāk strādā regularizēti robežintegrālvienādojumi, kurus var integrēt ar trapeces kvadratūru.
- Piliena dinamikas simulācijām zemfrekvences laukā labāk par katru atsevišķi strādā apvienotais pasīvās un aktīvās stabilizācijas algoritms. Tomēr pasīvās stabilizācijas algoritms ienes lielas tilpuma izmaiņas neprecīzi sarežģināto normāles vektoru dēļ.
- Lēni rotējošā laukā novērota piliena sekošana tā virzienam (sinhronais gadījums). Novērots, ka piliens vidēji seko līdzi lauka rotācijas virzienam, kad iestājās asinhronais režīms, bet kļūst nekustīgs pie lauka rotācijas frekvencēm  $\omega \gg 8\eta R_0/\gamma$ .
- Novērots, ka augstfrekvences lauks spēj izsaukt spontāno simetrijas sabrukšanu no pankūkas uz plakantārpu un no pankūkas uz jūraszvaigzni. Tomēr, lai nestabilitātes augtu un lai varētu iegūt jūraszvaigznes līdzvara figūru, jāizmanto smalka triangulācija.
- Atrasts, ka piliens pie zemkritiskām magnētiskā Bonda skaitļa vērtībām ieņem formu, kura tuvināti atbilst rotācijas elipsoīdam. Pie virskritiskām vērtībām pi-

liena forma tuvināti labi atbilst trīsasu elipsoīdam, bet tuvinājums kļūst sliktāks pie vēl lielākām magnētiskā Bonda skaitļa vērtībām.

- Piliena saspiešanos par pankūku augstfrekvences laukā teicami apraksta rotācijas elipsoīda minimums. Savukārt trīsasu elipsoīda enerģijas minimums kvalitatīvi labi apraksta spontāno simetrijas sabrukšanu no pankūkas uz plakantārpu un piliena formu pie virskritiskām Bonda skaitļa vērtībām.

# Pielikums A

## Integralvienādojumu teorēmas

### A.1 Divergēnce no normāles vektora

Ja normāles vektoru uzskata, kā līmeņvirsmas gradientu, tad tam uz virsmas definēta divergēnce. Parādīsim līdzīgi kā to dara [41], ka divergēnce no normāles vektora ir vienāda ar virsmas vidējo liekumu  $k_m$  jeb

$$k_m = \frac{1}{2} \nabla \cdot \mathbf{n} \quad (\text{A.1})$$

Izvēlamies perpendikulāras  $x$  un  $y$  asis virsmas pieskarplaknē un apskatām normālos šķēlumus ar virsmu. Ar  $\mathbf{n}$  apzīmēsim virsmas normāles vektoru un ar  $\mathbf{t}_x$  un  $\mathbf{t}_y$  pieskares vektorus līknēm, kuras veidojās šķelot virsmu ar  $xz$  un  $yz$  plakni, tad vidējo virsmas liekumu  $k_m$  varam definēt ar šādu formulu<sup>1</sup>

$$2k_m = \left| \frac{d\mathbf{t}_x}{dx} \right| + \left| \frac{d\mathbf{t}_y}{dy} \right| \quad (\text{A.2})$$

Nemot vērā, ka  $\frac{\partial n_x}{\partial z} = 0$ ,  $\frac{\partial n_y}{\partial z} = 0$ , un izsakot  $\mathbf{t}_x = \mathbf{n} \times \hat{y}$ ,  $\mathbf{t}_y = \mathbf{n} \times \hat{x}$  iegūstam virsmas liekumam sekajošu formulu

$$2k_m = \frac{\partial n_x}{\partial x} + \frac{\partial n_y}{\partial y} \quad (\text{A.3})$$

---

<sup>1</sup>Apskatām speciālgadījumu, kad liekumi ir pozitīvi.

Tā kā  $\frac{\partial n_x}{\partial z} = 0$ ,  $\frac{\partial n_y}{\partial z} = 0$ , tad pēc  $\nabla(\mathbf{n} \cdot \mathbf{n}) = 0$  iegūst, ka  $\frac{\partial n_z}{\partial z} = 0$  un augstāko formulu varam pierakstīt sekojošā veidā

$$2k_m = \nabla \cdot \mathbf{n} \quad (\text{A.4})$$

## A.2 Stoksa vienādojuma integrālā forma

Stoksa vienādojuma integrālformai izmantoju speciālgadījumus no [41], kurus visus var apvienot vienā formulā

$$C(\mathbf{x}_0)u_j(\mathbf{x}_0) = -\frac{1}{\eta} \int G_{ji}(\mathbf{x}, \mathbf{x}_0) \sigma_{ik} n_{\mathbf{x}k} dS_{\mathbf{x}} + \int u_i(\mathbf{x}) T_{ijk}(\mathbf{x}, \mathbf{x}_0) n_{\mathbf{x}k} dS_{\mathbf{x}} \quad (\text{A.5})$$

$$G_{ij}(\mathbf{x}, \mathbf{x}_0) = \frac{\delta_{ij}}{r} + \frac{r_i r_j}{r^3} \quad T_{ijk}(\mathbf{x}, \mathbf{x}_0) = -6 \frac{r_i r_j r_k}{r^5} \quad r = \mathbf{x} - \mathbf{x}_0 \quad (\text{A.6})$$

kur  $C(\mathbf{x}_0)$  ir  $8\pi$ , ja  $\mathbf{x}_0$  atrodas iekšpusē,  $4\pi$  uz virsmas un 0, ja ārpus noslēgtā apgaibala.

## A.3 Robežintegrālvienādojums potenciālam

Šajā sadaļā pierādīsim apgalvojumu, ka harmoniska funkcija  $\psi(\mathbf{x}_0)$  apmierina sekojošu vienādojumu

$$C(\mathbf{x}_0)\psi(\mathbf{x}_0) = \int \frac{\partial \psi}{\partial n_{\mathbf{x}}} \frac{1}{r} dS_{\mathbf{x}} - \int \psi(\mathbf{x}) \frac{\partial}{\partial n_{\mathbf{x}}} \frac{1}{r} dS_{\mathbf{x}} \quad r = |\mathbf{x} - \mathbf{x}_0| \quad (\text{A.7})$$

kur  $C(\mathbf{x}_0)$  ir  $4\pi$ , ja  $\mathbf{x}_0$  atrodas integrējamās robežas iekšpusē,  $2\pi$  uz virsmas un 0 ārpus ieslēdzotās virsmas.

No Grīna formulas iegūstam šādu identitāti

$$\int \left( \frac{\partial \psi}{\partial n_{\mathbf{x}}} \frac{1}{r} - \psi(\mathbf{x}) \frac{\partial}{\partial n_{\mathbf{x}}} \frac{1}{r} \right) dS_{\mathbf{x}} = \int \left( \frac{1}{r} \Delta_{\mathbf{x}} \psi - \psi \Delta_{\mathbf{x}} \frac{1}{r} \right) dV_{\mathbf{x}} \quad (\text{A.8})$$

Izvēloties  $\mathbf{x}_0$  ārpus integrējamā apgabala un pieprasot  $\Delta\psi = 0$  iegūstam

$$\int \left( \frac{\partial\psi}{\partial n_{\mathbf{x}}} \frac{1}{r} - \psi(\mathbf{x}) \frac{\partial}{\partial n_{\mathbf{x}}} \frac{1}{r} \right) dS_{\mathbf{x}} = 0 \quad (\text{A.9})$$

ar ko pierādām vienu gadījumu no (A.7).

Lai iegūtu identitāti uz virsmas, tuvinām  $\mathbf{x}_0$  integrēšanas virsmai no kuras izgriežam mazu pussfēru ar rādiusu  $\varepsilon$ , un attiecīgi integrāli (A.9) pierakstīt šādi

$$\int = \mathcal{P} \int + \int_{\varepsilon} \quad (\text{A.10})$$

kur  $\int_{\varepsilon}$  ir

$$\int_{\varepsilon} = \int_{\varepsilon} \left( \frac{\partial\psi}{\partial n_{\mathbf{x}}} \frac{1}{r} + \psi(\mathbf{x}) \frac{\mathbf{n}_{\mathbf{x}} \cdot \mathbf{r}}{r^3} dS_{\mathbf{x}} \right) \quad (\text{A.11})$$

Pārejot uz sfērisko koordinātu sistēmu un nemot vērā, ka  $\psi(\mathbf{x})$  uzskatāms konstants pa visu pussfēru

$$\int_{\varepsilon} = \int_{\varepsilon} \left( \frac{\partial\psi}{\partial n_{\mathbf{x}}} \frac{1}{\varepsilon} + \psi(\mathbf{x}_0) \frac{-1}{\varepsilon^2} \varepsilon^2 \sin(\theta) \right) d\theta d\phi = -\psi(\mathbf{x}_0) 2\pi \int_0^{\pi/2} \sin(\theta) d\theta = -2\pi\psi(\mathbf{x}_0) \quad (\text{A.12})$$

Apvienojot formulas (A.9), (A.10) un (A.12) pieradām otru formulu

$$2\pi\psi(\mathbf{x}_0) = \int \frac{\partial\psi}{\partial n_{\mathbf{x}}} \frac{1}{r} dS_{\mathbf{x}} - \mathcal{P} \int \psi(\mathbf{x}) \frac{\partial}{\partial n_{\mathbf{x}}} \frac{1}{r} dS_{\mathbf{x}} \quad r = |\mathbf{x} - \mathbf{x}_0| \quad (\text{A.13})$$

Lai iegūtu trešo identitāti, kad  $\mathbf{x}_0$  ir integrējamās virsmas iekšpusē, veicam formulā (A.12) integrēšanu pa visu sfēru.

## A.4 Robežintegrālvienādojums magnētiskajam laukam

Robežintegrālvienādojumu var rakstīt arī magnētiskajam laukam [37], [42]<sup>2</sup>

$$C(\mathbf{x}_0)\mathbf{B}(\mathbf{x}_0) = \int \mu \mathbf{J} \times \nabla_{\mathbf{x}} \frac{1}{r} dV - \int \left[ (\mathbf{n}_{\mathbf{x}} \times \mathbf{B}) \times \nabla_{\mathbf{x}} \frac{1}{r} + (\mathbf{n}_{\mathbf{x}} \cdot \mathbf{B}) \nabla_{\mathbf{x}} \frac{1}{r} \right] dS_{\mathbf{x}} \quad (\text{A.14})$$

kur  $\mathbf{J}$  ir brīvās strāvas, kuras apskatītajā modelī nav, bet  $C(\mathbf{x}_0)$  ir  $4\pi$  apgabala iekšpusē,  $2\pi$  uz virsmas, 0 ārpus apgabala.

Pienemsim par zināmu no [40] dotu formulu<sup>3</sup>

$$\mu \int_V \mathbf{J} \times \nabla_{\mathbf{x}} \frac{1}{r} dV_{\mathbf{x}} - \int \left[ (\mathbf{n}_{\mathbf{x}} \times \mathbf{B}) \times \nabla_{\mathbf{x}} \frac{1}{r} + (\mathbf{n}_{\mathbf{x}} \cdot \mathbf{B}) \nabla_{\mathbf{x}} \frac{1}{r} \right] dS_{\mathbf{x}} = 0 \quad (\text{A.15})$$

ja  $\mathbf{x}_0$  atrodas ārpus integrējamā apgabala. Tuvinot  $\mathbf{x}_0$  uz virsmu, izdalām no virsmas integrāļa mazu pussfēru ar rādiusu  $\varepsilon$

$$\int = \mathcal{P} \int + \int_{\varepsilon} \quad (\text{A.16})$$

Ar formulu  $\mathbf{r} \times (\mathbf{B} \times \mathbf{n}) = (\mathbf{r} \cdot \mathbf{n})\mathbf{B} - (\mathbf{B} \cdot \mathbf{n})\mathbf{r} + \mathbf{B} \times (\mathbf{r} \times \mathbf{n})$  varam pārrakstīt zemintegrāļa izteiksmi un veikt integrēšanu pa pussfēru iegūstot

$$\begin{aligned} & \int_{\varepsilon} (\mathbf{n}_{\mathbf{x}} \times \mathbf{B}) \times \nabla_{\mathbf{x}} \frac{1}{r} + (\mathbf{n}_{\mathbf{x}} \cdot \mathbf{B}) \nabla_{\mathbf{x}} \frac{1}{r} dS_{\mathbf{x}} \\ &= - \int_{\varepsilon} \frac{1}{r^3} ((\mathbf{r} \cdot \mathbf{n}_{\mathbf{x}})\mathbf{B} + \mathbf{B} \times (\mathbf{r} \times \mathbf{n}_{\mathbf{x}})) dS_{\mathbf{x}} = \mathbf{B}(\mathbf{x}_0) \int_{\varepsilon} \frac{1}{r^2} dS_{\mathbf{x}} = 2\pi \mathbf{B}(\mathbf{x}_0) \end{aligned} \quad (\text{A.17})$$

kur izmantojām  $\mathbf{r} \cdot \mathbf{n} = -r$  un  $\mathbf{r} \times \mathbf{n} = 0$ . Apvienojot formulas (A.15), (A.16) un

---

<sup>2</sup>Formula (23) pārskata rakstā [42], kur to iegūst regularizējot hipersingulāro integrālvienādojumu, kuru iegūst nemot gradientu no potenciāla integrālvienādojuma.

<sup>3</sup>Skatīt [40] 267. lpp 32. problēmu.

(A.17) iegūstam

$$2\pi \mathbf{B}(\mathbf{x}_0) = \int \mu \mathbf{J} \times \nabla_{\mathbf{x}} \frac{1}{r} dV - \mathcal{P} \int \left[ (\mathbf{n}_{\mathbf{x}} \times \mathbf{B}) \times \nabla_{\mathbf{x}} \frac{1}{r} + (\mathbf{n}_{\mathbf{x}} \cdot \mathbf{B}) \nabla_{\mathbf{x}} \frac{1}{r} \right] dS_{\mathbf{x}} \quad (\text{A.18})$$

ar ko pierādām formulu (A.14), kad  $\mathbf{x}_0$  atrodas uz virsmas. Gadījumā, kad  $\mathbf{x}_0$  atrodas virsmas iekšpusē, pussfēras vietā integrējam pa visu sfēru.

## A.5 Atbrīvošanās no virsmas liekuma

Līdzīgi, kā to dara [17] pierādīsim sekojošu formulu

$$\begin{aligned} - \int \nabla \cdot \mathbf{n}_{\mathbf{x}} \mathbf{n}_{\mathbf{x}_0} \cdot \left( \frac{\mathbf{I}}{r} + \frac{\mathbf{r} \otimes \mathbf{r}}{r^3} \right) \cdot \mathbf{n}_{\mathbf{x}} dS_{\mathbf{x}} &= \\ &= \int \frac{\mathbf{r} \cdot [\mathbf{n}_{\mathbf{x}} + \mathbf{n}_{\mathbf{x}_0}]}{r^3} \left( 1 - \frac{3(\mathbf{r} \cdot \mathbf{n}_{\mathbf{x}})(\mathbf{r} \cdot \mathbf{n}_{\mathbf{x}_0})}{r^2} \right) dS_{\mathbf{x}} \quad (\text{A.19}) \end{aligned}$$

Patvalīgām vektora funkcijām  $\mathbf{A}$  un  $\mathbf{B}$  izpildās sekojoša identitāte

$$\int \frac{\partial}{\partial x_j} (A_i B_j) n_i dS_{\mathbf{x}} = \int \frac{\partial}{\partial x_j} (A_j B_i) n_i dS_{\mathbf{x}} \quad (\text{A.20})$$

kur integrējam pa patvalīgu noslēgtu virsmu. Izvēlamies  $A_i = G_{ik}(\mathbf{x}, \mathbf{x}_0) = \frac{\delta_{ik}}{r} + \frac{r_i r_k}{r^3}$  un  $B_i = n_i$  (Vispārīgāku pieeju, kad  $\gamma$  nav konstants skatīt [17]) un izmantojot, ka  $\nabla(\mathbf{n} \cdot \mathbf{n}) = 0$  un  $\nabla \cdot \mathbf{G}(\mathbf{x}, \mathbf{x}_0) = 0$ , iegūstam

$$\int \frac{\partial G_{ik}}{\partial x_j} n_j n_i dS_{\mathbf{x}} + \int G_{ik}(\mathbf{x}, \mathbf{x}_0) \frac{\partial n_j}{\partial x_j} n_i dS_{\mathbf{x}} = 0 \quad (\text{A.21})$$

Ko varam pārrakstīt arī šādi

$$\int G_{ik}(\mathbf{x}, \mathbf{x}_0) n_i \nabla \cdot \mathbf{n} dS_{\mathbf{x}} = \int \frac{r_j}{r^3} \left[ 1 - 3 \frac{(\mathbf{r} \cdot \mathbf{n}_{\mathbf{x}})^2}{r^2} \right] dS_{\mathbf{x}} \quad (\text{A.22})$$

kas parāda, ka varam atbrīvoties no tiešas virsmas liekuma rēķināšanas. Šī forma ir derīga gadījumiem, kad  $\mathbf{x}_0$  atrodas ārpus integrējamā apgabala. Kad tuvinām

$\mathbf{x}_0$  virsmai, tad otrs summas loceklis integrāli ir regulārs, jo divi normāles vektori skalāri reizinās ar rādiusvektorū, kamēr pirmais ir singulārs.

Lai no šīs singularitātes atbrīvotos apskatam regularizāciju šādai singularitātei

$$\begin{aligned} - \int \frac{\partial}{\partial n_{\mathbf{x}_0}} \frac{1}{r} dS_{\mathbf{x}} &= \int \mathbf{n}_{\mathbf{x}_0} \cdot \nabla_{\mathbf{x}} \frac{1}{r} dS_{\mathbf{x}} = \int (\mathbf{n}_{\mathbf{x}} + \mathbf{n}_{\mathbf{x}_0}) \nabla_{\mathbf{x}} \frac{1}{r} dS_{\mathbf{x}} \\ &= \int (\mathbf{n}_{\mathbf{x}} + \mathbf{n}_{\mathbf{x}_0}) \cdot \frac{1}{r} dS_{\mathbf{x}} - \int \mathbf{n}_{\mathbf{x}} \cdot \nabla_{\mathbf{x}} \frac{1}{r} dS_{\mathbf{x}} \quad (\text{A.23}) \end{aligned}$$

Tagad tuvinot  $\mathbf{x}_0$  virsmai ar regularizācijas formulu (A.29) iegūstam<sup>4</sup>

$$\int \mathbf{n}_{\mathbf{x}_0} \cdot \nabla_{\mathbf{x}} \frac{1}{r} dS_{\mathbf{x}} = \int (\mathbf{n}_{\mathbf{x}} + \mathbf{n}_{\mathbf{x}_0}) \cdot \frac{1}{r} dS_{\mathbf{x}} + 2\pi \quad (\text{A.24})$$

Lai iegūtu regularizēto formu kādu izmanto [16] un šajā darbā, izmantojam

$$\int \frac{\mathbf{r} \cdot \mathbf{n}_{\mathbf{x}} (\mathbf{r} \cdot \mathbf{n}_{\mathbf{x}_0})}{r^5} dS_{\mathbf{x}} = \int \nabla_{\mathbf{x}} \cdot \frac{\mathbf{r} (\mathbf{r} \cdot \mathbf{n}_{\mathbf{x}_0})^2}{r^5} dV = 0 \quad (\text{A.25})$$

ja  $\mathbf{x}_0$  atrodas ārpus integrējamā apgabala. Tuvinot  $\mathbf{x}_0$  virsmai iegūstam (integrējot singularitāti)

$$\mathcal{P} \int \frac{\mathbf{r} \cdot \mathbf{n}_{\mathbf{x}} (\mathbf{r} \cdot \mathbf{n}_{\mathbf{x}_0})^2}{r^5} dS_{\mathbf{x}} = \frac{2\pi}{3} \quad (\text{A.26})$$

Saliekot formulas (A.24),(A.25),(A.26) formulā (A.22) piereizinātu ar normāles vektoru punktā  $\mathbf{x}_0$  iegūstam

$$\int \nabla \cdot \mathbf{n} G_{ik}(\mathbf{x}, \mathbf{x}_0) n_i(\mathbf{x}) n_k(\mathbf{x}_0) dS_{\mathbf{x}} = \int \frac{\mathbf{r} \cdot (\mathbf{n}_{\mathbf{x}} + \mathbf{n}_{\mathbf{x}_0})}{r^3} \left[ 1 - \frac{3(\mathbf{r} \cdot \mathbf{n}_{\mathbf{x}})(\mathbf{r} \cdot \mathbf{n}_{\mathbf{x}_0})}{r^2} \right] dS_{\mathbf{x}} \quad (\text{A.27})$$

---

<sup>4</sup>Šādu regularizācijas formulu mēgināju izmantot singulārā integrālvienādojuma lauka normālai komponentei (4.21) regularizācijai (šajā darbā neiekļāvu), kura manā izpildījumā deva lielākas kļūdas nekā neregularizētā versija.

## A.6 Stoksa robežintegrālvienādojuma regularizācijas formula

Stoksa robežintegrālvienādojumu regularizācijai plaši izmanto identitāti [17]

$$\int \left( \frac{\mathbf{I}}{r} + \frac{\mathbf{r} \otimes \mathbf{r}}{r^3} \right) \cdot \mathbf{n}_x dS_x = 0 \quad (\text{A.28})$$

Tās pierādišanai pārejam uz integrāli pa tilpumu, kur zemintegrāla izteiksme ir vienāda ar 0. Atšķirībā no citām regularizācijas formulām šī ir ar vāju (integrējamu) singularitāti, tādēļ nav nepieciešams atsevišķi to apstrādāt, kad tuvinām  $\mathbf{x}_0$  virsmai.

## A.7 Potenciāla robežintegrālvienādojuma regularizācijas formula

Šajā sadaļā pierādīsim sekojošu identitāti, līdzīgi kā to dara [30]

$$\mathcal{P} \int \frac{\partial}{\partial n_x} \frac{1}{4\pi|\mathbf{x} - \mathbf{x}_0|} dS_x = \frac{1}{2} \quad (\text{A.29})$$

Kad  $\mathbf{x}_0$  atrodas ārpus integrējamā apgabala, tad varam pāriet uz integrāli pa tilpumu

$$\int \frac{\partial}{\partial n_x} \frac{1}{r} dS_x = \int \Delta_x \frac{1}{r} dV = 0 \quad (\text{A.30})$$

Tuvinot no ārpuses  $\mathbf{x}_0$  virsmai varam šo integrāli sadalīt kā Košī integrāla un integrāla ap mazu pussfēru ar rādiusus  $\varepsilon$  summu

$$\int \frac{\partial}{\partial n_x} \frac{1}{r} dS_x = \mathcal{P} \int \frac{\partial}{\partial n_x} \frac{1}{r} dS_x + \int_{\varepsilon} \frac{\partial}{\partial n_x} \frac{1}{r} dS_x \quad (\text{A.31})$$

Integrāli ap pussfēru varam aprēķināt pārejot uz sfērisko koordinātu sistēmu

$$\int_{\varepsilon} \frac{\partial}{\partial n_x} \frac{1}{r} dS_x = - \int_{\varepsilon} \frac{\mathbf{n}_x \cdot \mathbf{r}}{r^3} dS_x = - \int_{\varepsilon} \frac{1}{\varepsilon^2} \varepsilon^2 \sin(\theta) dS_x = -2\pi \quad (\text{A.32})$$

Apvienojot (A.32), (A.31) iegūstam nepieciešamo formulu (A.29).

## A.8 Magnētiskā lauka robežintegrālvienādojuma regularizācija

Pierādīsim sekojošu formulu, kura iepriekš izmantota [38]

$$\mathcal{P} \int \frac{\mathbf{n} \times \mathbf{r}}{r^3} dS_x = 0 \quad (\text{A.33})$$

Vispirms apskatām gadījumu, kad  $\mathbf{x}_0$  atrodas ārpus integrējamā apgabala, tad komponentēs izrakstīts integrālis  $\int \frac{\mathbf{n}_x \times \mathbf{r}}{r^3} dS_x$  ir

$$\int \epsilon_{ijk} n_j \frac{\epsilon_{ijk} r_k}{r^3} dS_x = \int \frac{\partial}{\partial x_j} \frac{\epsilon_{ijk} r_k}{r^3} dS_x = \int \left[ \nabla_{\mathbf{x}} \times \frac{\mathbf{r}}{r^3} \right]_i dS_x = 0 \quad (\text{A.34})$$

Tuvinot  $\mathbf{x}_0$  virsmai sadalām to divos integrāļos

$$\int \frac{\mathbf{n} \times \mathbf{r}}{r^3} dS_x = \mathcal{P} \int \frac{\mathbf{n} \times \mathbf{r}}{r^3} dS_x + \int_{\varepsilon} \frac{\mathbf{n} \times \mathbf{r}}{r^3} dS_x \quad (\text{A.35})$$

kur otrs ir pa izgrieztu pussfēru. Tā kā uz sfēras normāles vektors ir kolineārs ar rādiusvektorū, tad otrs integrālis ir 0, un iegūstam (A.33), kas bija jāpierāda.

# Pielikums B

## Vektorpotenciāla integrālvienādojuma formulējums

Literatūrā ir atrodams arī vektorpotenciāla formulējums elektromagnētiskajam laukam, kas izmantots [43] un [37] un ko šeit arī apskatu. Tā izmantošana lauka normālās komponentes atrašanai uz virsmas sekotu no identitātes

$$B_n \Delta S = \int \mathbf{B} \mathbf{n}_x dS_x = \int \nabla_x \times \mathbf{A} \cdot \mathbf{n}_x dS = \int \mathbf{A} dl \quad (\text{B.1})$$

kur integrēšana būtu ap mazu kontūru ap virsotni.

Vektorpotenciāla integrālvienādojumu varam iegūt šādas formulas [43] un [37]

$$C(\mathbf{x}_0) \mathbf{A}(\mathbf{x}_0) = \int \left[ (\mathbf{B} \times \mathbf{n}_x) \frac{1}{r} - \mathbf{A} \frac{\mathbf{r} \cdot \mathbf{n}_x}{r^3} + \mathbf{A} \times \frac{\mathbf{n}_x \times \mathbf{r}}{r^3} \right] dS_x \quad (\text{B.2})$$

kur  $C(\mathbf{x}_0)$  ir  $4\pi$  iekšpusē un  $2\pi$ , kad  $\mathbf{x}_0$  atrodas uz robežas. Rakstot robežintegrlāvienādojumu vektorpotenciālam piliena apgabalam un apkārt pielienam esošajai telpai:

$$2\pi \mathbf{A}^{(i)}(\mathbf{x}_0) = \mathcal{P} \int_{\partial D} \left[ (\mathbf{B}^{(i)} \times \mathbf{n}_x) \frac{1}{r} - \mathbf{A}^{(i)} \frac{\mathbf{r} \cdot \mathbf{n}_x}{r^3} + \mathbf{A}^{(i)} \times \frac{\mathbf{n}_x \times \mathbf{r}}{r^3} \right] dS_x \quad (\text{B.3})$$

$$2\pi \mathbf{A}^{(e)}(\mathbf{x}_0) = -\mathcal{P} \int_{\partial D} \left[ (\mathbf{B}^{(e)} \times \mathbf{n}_{\mathbf{x}}) \frac{1}{r} - \mathbf{A}^{(e)} \frac{\mathbf{r} \cdot \mathbf{n}_{\mathbf{x}}}{r^3} + \mathbf{A}^{(e)} \times \frac{\mathbf{n}_{\mathbf{x}} \times \mathbf{r}}{r^3} \right] dS_{\mathbf{x}} + 4\pi \mathbf{A}_0(\mathbf{x}_0) \quad (\text{B.4})$$

$$\mathbf{A}_0(\mathbf{x}_0) = \frac{1}{4\pi} \int_{\partial\infty} \left[ (\mathbf{B} \times \mathbf{n}_{\mathbf{x}}) \frac{1}{r} - \mathbf{A} \frac{\mathbf{r} \cdot \mathbf{n}_{\mathbf{x}}}{r^3} + \mathbf{A} \times \frac{\mathbf{n}_{\mathbf{x}} \times \mathbf{r}}{r^3} \right] dS_{\mathbf{x}} \quad (\text{B.5})$$

Ievietojot robežnosacījumus (2.5), no kā seko ka arī vektorpotenciāls uz piliena virsmas ir nepārtraukts, iegūst šādu robežintegrālvienādojumu vektorpotenciālam:

$$\mathbf{A}(\mathbf{x}_0) = \frac{2\mu \mathbf{A}_0(\mathbf{x}_0)}{\mu + 1} + \frac{1}{2\pi} \frac{\mu - 1}{\mu + 1} \mathcal{P} \int_{\partial D} \mathbf{A} \frac{\mathbf{r} \cdot \mathbf{n}_{\mathbf{x}}}{r^3} - \mathbf{A} \times \frac{\mathbf{n}_{\mathbf{x}} \times \mathbf{r}}{r^3} dS_{\mathbf{x}} \quad (\text{B.6})$$

Šim integrālvienādojumam ir iespējams veikt regularizāciju ar § A.7 un § A.8, tomēr šo vektorpotenciāla integrālvienādojumu skaitliski implementējis neesmu, tādēļ grūti spriest par šo operāciju efektivitāti.

# Pielikums C

## Pasīvās stabilizācijas enerģijas funkcijas minimizācija

Lai varētu minimizēt pasīvās stabilizācijas enerģijas funkciju

$$F = \sum_{(i,j)} \left[ \frac{d}{dt} (\mathbf{x}_i - \mathbf{x}_j)^2 \right]^2 + 0.4 \sum_{\Delta} \frac{1}{C_{\Delta}^2} \left( \frac{dC_{\Delta}}{dt} \right)^2 \quad C_{\Delta} = \frac{S_{\Delta}}{a^2 + b^2 + c^2} \quad (\text{C.1})$$

ir jāiegūst tās atvasinājumi pret ātrumiem virsmas plaknē<sup>1</sup>, ko tālāk var ievietot piemēram konjugētā gradiента metodē [16].

Vispirms iegūsim atvasinājumu  $dC_{\Delta}/dt$

$$C_{\Delta} = \frac{1}{4} \sqrt{1 - 2 \frac{a^4 + b^4 + c^4}{(a^2 + b^2 + c^2)^2}} \quad (\text{C.2})$$

$$\begin{aligned} \frac{dC_{\Delta}}{dt} = & -\frac{1}{16C_{\Delta}} \frac{4}{(a^2 + b^2 + c^2)^3} [\mathbf{x}_a \cdot \mathbf{v}_a (a^2(a^2 + b^2 + c^2) - a^4 - b^4 - c^4) \\ & + \mathbf{x}_b \cdot \mathbf{v}_b (b^2(a^2 + b^2 + c^2) - a^4 - b^4 - c^4) \\ & + \mathbf{x}_c \cdot \mathbf{v}_c (c^2(a^2 + b^2 + c^2) - a^4 - b^4 - c^4)] \end{aligned} \quad (\text{C.3})$$

---

<sup>1</sup>Lai neizmainītu ātruma virzienu normāles virzienā

kur ievedām apzīmējumus

$$\mathbf{x}_a = \mathbf{x}_2 - \mathbf{x}_1 \quad \mathbf{x}_b = \mathbf{x}_3 - \mathbf{x}_2 \quad \mathbf{x}_c = \mathbf{x}_1 - \mathbf{x}_2 \quad (\text{C.4})$$

Ievedot jaunus apzīmējumus, varam pārrakstīt  $dC_\Delta/dt$  sekojošā veidā

$$\begin{aligned} \frac{dC_\Delta}{dt} &= -A\mathbf{x}_a \cdot \mathbf{v}_a - B\mathbf{x}_b \cdot \mathbf{v}_b - C\mathbf{x}_c \cdot \mathbf{v}_c & (\text{C.5}) \\ A &= \frac{1}{4C_\Delta(a^2 + b^2 + c^2)^3} (a^2(a^2 + b^2 + c^2) - a^4 - b^4 - c^4) \\ B &= \frac{1}{4C_\Delta(a^2 + b^2 + c^2)^3} (b^2(a^2 + b^2 + c^2) - a^4 - b^4 - c^4) \\ C &= \frac{1}{4C_\Delta(a^2 + b^2 + c^2)^3} (c^2(a^2 + b^2 + c^2) - a^4 - b^4 - c^4) & (\text{C.6}) \end{aligned}$$

Nākamais solis ir aprēķināt šīs funkcijas gradientu pret trijstūra virsotņu ātrumiem

$$\begin{aligned} \nabla_{\mathbf{v}_1} \frac{dC_\Delta}{dt} &= A\mathbf{x}_a - C\mathbf{x}_c \\ \nabla_{\mathbf{v}_2} \frac{dC_\Delta}{dt} &= B\mathbf{x}_b - A\mathbf{x}_a \\ \nabla_{\mathbf{v}_3} \frac{dC_\Delta}{dt} &= C\mathbf{x}_c - B\mathbf{x}_b & (\text{C.7}) \end{aligned}$$

Pirmās summas gradients izsakāms ir ar formulu

$$\nabla_{\mathbf{v}_k} \sum_{(i,j)} \left[ \frac{d}{dt} \mathbf{x}_{ij}^2 \right]^2 = -8 \sum_j [\mathbf{x}_{jk} \cdot (\mathbf{v}_j - \mathbf{v}_k)] \mathbf{x}_{jk} \quad (\text{C.8})$$

kur labajā pusē summēšana ir ap virsotnes  $k$  kaimiņu virsotnēm.

No katras trijstūra tādēļ varam iegūt sekojošu ieguldījumu gradientam

$$\begin{aligned} \nabla_{\mathbf{v}_j} F &= \sum_{\Delta_i} \tilde{A}\mathbf{x}_a - \tilde{C}\mathbf{x}_c \\ \tilde{A} &= 2 \cdot 0.4 \frac{dC_\Delta}{dt} \frac{A}{C_\Delta^2} - 8\mathbf{x}_a \cdot \mathbf{v}_a & (\text{C.9}) \end{aligned}$$

$$\tilde{C} = 2 \cdot 0.4 \frac{dC_\Delta}{dt} \frac{C}{C_\Delta^2} - 8\mathbf{x}_c \cdot \mathbf{v}_c \quad (\text{C.10})$$

Un visbeidzot katrā virsotnē gradientu projicē pēc formulas

$$\mathbf{f}_j = (\mathbf{I} - \mathbf{n}_j \otimes \mathbf{n}_j) \nabla_{\mathbf{v}_j} F \quad (\text{C.11})$$

ko tālāk var padod, piemēram, konjugētā gradienta minimizācijas algoritmam.

# Pielikums D

## Koda izvilkumi

Simulāciju kodā ir izmantoti 3 dažādi iteratori triangulācijas virsotnei. Lai noskaidrotu kādas ir kaimiņu virsotnes izmanto VertexVRing. Lai atrastu trijstūrus, kurai ir dotā virsotne, izmanto FaceVRing. Savukārt pretējo trijstūra malu atrašanai apdotu virsotni izmanto DoubleVertexVRing. Šīs operācijas kā arī citas ir uzrakstītas *SurfaceGeometry.jl* bibliotēkā ar atbilstošiem testa piemēriem.<sup>1</sup>

### D.1 Potenciāla aprēķināšana

function PotentialSimple(points,faces,mu,H0)  
**ievade** triangulācijas virsotnes *points* un orientēti trijstūri *faces*. Materiāla  $\mu$  un magnētiskā lauka intensitāte tālu no piliena  $\mathbf{H}_0$ .  
**izvade** potenciāls uz virsmas, risināts ar § 4.2.1.

```
1 function PotentialSimple(points , faces , hmag,H0; regularize=true , normals=nothing )
2
3     if normals==nothing
4         normals = Array(Float64 , size(points)...)
5         NormalVectors!(normals , points , faces , i->FaceVRing(i , faces ))
6     end
7
8     A = zeros(Float64 , size(points ,2) , size(points ,2))
9
10    vareas = zeros(Float64 , size(points ,2))
11    for i in 1:size(faces ,2)
```

---

<sup>1</sup>Šīs operācijas katru reizi veic meklēšanu visā orientēto trijstūru masīvā, tādēļ ir lēnas. Labāka pieeja ir iepriekš glabāt kādu daļu par triangulāciju masīvā, ar kuriem var paātrināt šīs operācijas vairāk par 10 reizēm. Tomēr šajos aprēķinos kā atklāja koda profilešana, tas ir maznozīmīgs uzlabojums.

```

12     v1,v2,v3 = faces[:,i]
13     area = norm(cross(points[:,v2]-points[:,v1],points[:,v3]-points[:,v1]))/2
14     vareas[v1] += area/3
15     vareas[v2] += area/3
16     vareas[v3] += area/3
17 end
18
19 for xkey in 1:size(points,2)
20
21     x = points[:,xkey]
22     nx = normals[:,xkey]
23
24     for ykey in 1:size(points,2)
25         if xkey==ykey
26             continue
27         end
28
29         y = points[:,ykey]
30         ny = normals[:,ykey]
31
32         A[ykey,xkey] = dot(y-x,ny)/norm(y-x)^3 * vareas[ykey]
33     end
34 end
35
36 B = zeros(Float64,size(points,2))
37 for xkey in 1:size(points,2)
38     B[xkey] = 2*dot(H0,points[:,xkey])/(hmag+1)
39 end
40
41
42 if regularize==true
43     A = A'
44     psi = (eye(A)*(1-(hmag-1)/(hmag+1)) - 1/2/pi * (hmag-1)/(hmag+1) * (A - diagm(
45         Float64[sum(A[i,:]) for i in 1:size(A,2)]))) \ B
46 else
47     A = A*(hmag-1)/(hmag+1)/2/pi
48     A = A'
49     psi = (eye(A) - A)\B
50 end
51 return psi
52 end

```

## D.2 Tangencialā lauka komponenšu atrašana ar galīgajām diferencēm

function HField(points,faces,psi)

**ievade** triangulācijas virsotnes *points* un orientēti trijstūri *faces*. Un potenciāla vērtība triangulācijas virsotnēs *psi*.

**izvade** Magnētiskā lauka intensitātes vektors katrā virsotnē.<sup>2</sup>

```

1 function HField(points,faces,psi)
2
3     H = Array(Float64,size(points)...)
```

---

<sup>2</sup>Derīgas ir tikai tangencialās komponentes

```

5      for xkey in 1:size(points,2)
6
7      x = points[:,xkey]
8      psix = psi[xkey]
9
10     vvec = Float64[]
11     dphi = Float64[]
12
13     for ykey in VertexVRing(xkey,faces)
14         y = points[:,ykey]
15
16         vvec = [vvec; y-x]
17         dphi = [dphi; psi[ykey]-psix]
18     end
19
20     A, B = vvec, dphi
21
22     A = transpose(reshape(A,3,div(length(A),3)))
23     H[:,xkey] = inv(transpose(A)*A)*transpose(A)*B
24
25 end
26
27 return H
28 end

```

### D.3 Lauka normālās komponentes aprēķināšana

function strquad(q,x1,x2,x3)

**ievade** funkcija  $q(\xi, \eta)$ , singulārā trijstūra virsotne  $x_1$  un pārējās  $x_2, x_3$ .

**izvade** vāji singulāra kvadratūra trijstūra elementam (4.24).

function NormalFieldCurrent(points,faces,Ht,mu,H0)

**ievade** triangulācijas virsotnes *points* un orientēti trijstūri *faces*. Lauka tangenciālās komponentes katrā virsotnē  $\mathbf{H}_t$ <sup>3</sup>. Materiāla  $\mu$  un magnētiskā lauka intensitāte tālu no piliena  $\mathbf{H}_0$ .

**izvade** Magnētiskā lauka normālā komponente katrā virsotnē izmantojot § 4.3.3.

```

1 using FastGaussQuadrature
2 const NP = 10
3 const t, w = gausslegendre(NP)
4 function strquad(q::Function,x1,x2,x3)
5
6     B = dot(x3-x1,x2-x1)/norm(x2-x1)^2
7     C = norm(x3-x1)^2/norm(x2-x1)^2
8     hS = norm(cross(x2-x1,x3-x1))
9
10    s = 0
11    for i in 1:NP
12        Chi = pi/4*(1 + t[i])
13
14        R = 1/(cos(Chi) + sin(Chi))
15        si = 0

```

---

<sup>3</sup>Funkcijā izprojicē ārā iespējamās lauka normālās komponentes.

```

16      for j in 1:NP
17          rho = R/2*(1 + t[j])
18          si += q(rho*cos(Chi),rho*sin(Chi))*w[j]
19      end
20
21      s += si*R/2 / sqrt(cos(Chi)^2 + B*sin(2*Chi) + C*sin(Chi)^2) * w[i]
22  end
23  s *= pi/4
24
25  return s*hS/norm(x2 - x1)
26 end
27
28 function NormalFieldCurrent(points,faces,Ht,hmag,H0; normals=nothing)
29
30     if normals==nothing
31         normals = Array(Float64,size(points)... )
32         NormalVectors!(normals,points,faces,i->FaceVRing(i,faces))
33     end
34
35     for xkey in 1:size(points,2)
36         nx = normals[:,xkey]
37         Ht[:,xkey] = (eye(3) - nx*nx')*Ht[:,xkey]
38     end
39
40     vareas = zeros(Float64,size(points,2))
41     for i in 1:size(faces,2)
42         v1,v2,v3 = faces[:,i]
43         area = norm(cross(points[:,v2]-points[:,v1],points[:,v3]-points[:,v1])) / 2
44         vareas[v1] += area/3
45         vareas[v2] += area/3
46         vareas[v3] += area/3
47     end
48
49
50     function qs(xi,eta,v1,v2,v3,x,nx,Htx)
51         y = (1 - xi - eta)*points[:,v1] + xi*points[:,v2] + eta*points[:,v3]
52         Hty = (1 - xi - eta)*Ht[:,v1] + xi*Ht[:,v2] + eta*Ht[:,v3]
53         ny = (1 - xi - eta)*normals[:,v1] + xi*normals[:,v2] + eta*normals[:,v3]
54         s = -dot(nx,cross(Hty - Htx,cross(ny,-(y-x)/norm(y-x)^2)))
55     end
56
57     Hn = Array(Float64,size(points,2))
58
59     for xkey in 1:size(points,2)
60
61         nx = normals[:,xkey]
62         x = points[:,xkey] + eps*nx
63         Htx = Ht[:,xkey]
64
65         s = 0
66         for ykey in 1:size(points,2)
67             !(xkey==ykey) || continue
68             y = points[:,ykey]
69             ny = normals[:,ykey]
70             Hty = Ht[:,ykey]
71
72             s += dot(nx,-(Hty-Htx)*dot((y-x)/norm(y-x)^3,ny)) * vareas[ykey]
73             s += -dot(nx,cross(Hty-Htx,cross(ny,-(y-x)/norm(y-x)^3))) * vareas[ykey]
74         end
75
76         #### Making a proper hole
77         for (v2,v3) in DoubleVertexVRing(xkey,faces)
78             area = norm(cross(points[:,v2]-x,points[:,v3]-x))/2
79
80             ny = normals[:,v2]
81             y = points[:,v2]
82             s -= -dot(nx,cross(Ht[:,v2]-Htx,cross(ny,-(y-x)/norm(y-x)^3))) * area/3
83
84             ny = normals[:,v3]
85             y = points[:,v3]

```

```

86         s -= -dot(nx,cross(Ht[:,v3]-Htx,cross(ny,-(y-x)/norm(y-x)^3))) * area/3
87
88     ### Singular triangle integration
89     s += strquad((xi,eta) -> qs(xi,eta,xkey,v2,v3,x,nx,Htx),x,points[:,v2],points
90                  [:,v3])
91
92     Hn[xkey] = dot(H0,nx)/hmag + 1/4/pi * (1-hmag)/hmag * s
93
94
95     return Hn
96

```

## D.4 Virsmas ātruma aprēķināšana

`function InterfaceSpeedZinchenko(points,faces,fn,eta,gamma)`

**ievade** triangulācijas virsotnes *points* un orientēti trijstūri *faces*. Uz virsmu iedarbojošs spēks  $\tilde{f}$  katrā virsotnē. Piliena viskozitāte  $\eta$  un tā virsmas spraigums  $\gamma$ .

**izvade** Virsmas ātrums katrā virsotnē normāles virzienā ar (5.6).

```

1  function InterfaceSpeedZinchenko(points ,faces ,forcen ,etaP ,gammap)
2
3     normals = Array(Float64 ,size(points)...)
4     NormalVectors!(normals ,points ,faces ,i->FaceVRing(i ,faces))
5
6     vareas = zeros(Float64 ,size(points ,2))
7     for i in 1:size(faces ,2)
8         v1,v2,v3 = faces[:,i]
9         area = norm(cross(points[:,v2]-points[:,v1] ,points[:,v3]-points[:,v1]))/2
10        vareas[v1] += area/3
11        vareas[v2] += area/3
12        vareas[v3] += area/3
13    end
14
15    velocityn = zeros(Float64 ,size(points ,2))
16
17    for xkey in 1:size(points ,2)
18
19        x = points[:,xkey]
20        nx = normals[:,xkey]
21        fx = forcen[xkey]
22
23        s = 0
24        for ykey in 1:size(points ,2)
25            if ykey==xkey
26                continue
27            end
28
29            y = points[:,ykey]
30            ny = normals[:,ykey]
31            fy = forcen[ykey]
32
33            s += vareas[ykey]*1./8./pi/etaP* dot(y-x ,nx+ny)/norm(y-x)^3*(1-3*dot(y-x ,nx)*
34                           dot(y-x ,ny)/norm(y-x)^2) * gammap
35
36            s += vareas[ykey]*1./8./pi/etaP* ( dot(nx ,ny)/norm(x-y) + dot(nx ,x-y)*dot(ny ,x
37                           -y)/norm(x-y)^3)*(fy - fx)
38
39    end

```

```

37         velocityn[xkey] = s
38     end
39
40     return velocityn
41 end

```

## D.5 Piemērs paralēlai implementācijai ar julia

Šī sadaļa parāda, kā var izmantot julia programmēšanas valodu, lai rakstītu paralēli izpildāmu robežintegrālvienādojumu kodu.<sup>4</sup> Tomēr kods ir sens, tādēļ detaļas nekomentēšu.<sup>5</sup>

```

1 @everywhere function fillcolumn!(a::AbstractArray{Float64,1},xkey::Int,faces::
2   AbstractArray{Int,2},points::AbstractArray{Float64,2})
3
4   nfaces = size(faces,2)
5   nverticies = size(points,2)
6
7   v1,v2,v3 = faces[:,xkey]
8   p1, p2, p3 = points[:,v1], points[:,v2], points[:,v3]
9   x = (p1 + p2 + p3)/3
10  normalx = cross(p2-p1,p3-p1)
11  normalx /= norm(normalx)
12
13  for ykey in 1:nfaces
14    if xkey==ykey
15      a[ykey] = 0
16    else
17      v1y,v2y,v3y = faces[:,ykey]
18      y1, y2, y3 = points[:,v1y], points[:,v2y], points[:,v3y]
19      f(y::Array{Float64,1}) = dot(normalx,x-y)/norm(x-y)^3
20      a[ykey] = trquad2(f,y1,y2,y3) # !!! Not in here
21    end
22  end
23
24 @everywhere function chunkcalc!(A::AbstractArray{Float64,2},faces::AbstractArray{Int,2},
25   points::AbstractArray{Float64,2},xrange)
26   @show xrange
27   for xkey in xrange
28     fillcolumn!(A,xkey,faces,points)
29   end
30
31 function magnetic_field_trquad(faces::AbstractArray{Int,2},points::AbstractArray{
32   Float64,2},hmag::Float64,H0::Array{Float64,1})
33
34   nfaces = size(faces,2)
35   nverticies = size(points,2)
36
37   A = zeros(Float64,nfaces,nfaces)
38   A = convert(SharedArray,A)
39
40   @sync begin
41     for (i,p) in enumerate(procs(A))
42       @spawnat p begin
43         a = Array(Float64,nfaces)
44         for xkey in i:length(procs(A)):nfaces

```

---

<sup>4</sup>Ir testēts, ka pie diviem atlautiem kodoliem iegūst 2 reižu īsāku aprēķina laiku.

<sup>5</sup>Kodam ir šeit ir parauga nozīme, ja tomēr izdomāju efektīvi lietot daudzkodolu procesoru.

```
45         println("xkey=$xkey")
46         fillcolumn!(a,xkey,faces,points)
47         A[:,xkey] = a[:,]
48     end
49 end
50 end
51
52 A = A*(hmag-1)/(hmag+1)/2/pi
53 B = zeros(Float64,nfaces)
54
55 return A \ B
56
57 end
```

# Pateicības

Lielu paldies saku savai mammai par rūpēm un pacietību darbu veidojot, par komatu kļudu labošanu un citām mulķīgām piezīmēm, kas lāva paskatīties uz darbu ar vieglumu. Paldies Andrejam Cēberam par noticešanu manām spejām veikt šo darbu, par vērtīgām sarunām, kurās kristalizējās veicamie uzdevumi, par brīvību tos veikt savā kārtībā, par finansiālu atbalstu šī darba izstrādes laikā, par iespēju strādāt no savām mājām Valmierā, kur varēju viegli nodarboties ar saviem vaļaspriekiem prieka un izmisuma situācijās. Pateicos arī Elīnai Potaņinai par iedrošinājumu sākt strādāt pie A. Cēbera, kura pieredze ir nenovērtējama.

# Literatūra

- [1] Chandrasekhar, S. (1965). The Stability of a Rotating Liquid Drop. Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences, 286(1404), 1–26. doi:10.1098/rspa.1965.0127
- [2] Bacri, J.-C., Cebers, A. O., & Perzynski, R. (1994). Behavior of a magnetic fluid microdrop in a rotating magnetic field. Physical Review Letters, 72(17), 2705–2708. doi:10.1103/physrevlett.72.2705
- [3] Guntars Kitenebergs (2015). Hydrodynamic instabilities in microfluidic magnetic fluid flows. University of Latvia [phd thesis]
- [4] Rosensweig, R. E. (1979). Fluid Dynamics and Science of Magnetic Liquids. Advances in Electronics and Electron Physics, 103–199. doi:10.1016/s0065-2539(08)60306-6
- [5] Bacri, J. C., & Salin, D. (1982). Instability of ferrofluid magnetic drops under magnetic field. J. Physique Lett., 43(17), 649–654. doi:10.1051/jphyslet:019820043017064900
- [6] Raimonds Narnickis (2016). Magnētiska piliena līdzsvara figūru skaitlisks aprēķins augstfrekvences rotējošā magnētiskā laukā. Latvijas Universitāte. [bsc thesis]
- [7] A.O. Tsebers (1983). Numerical simulation of MHD instability in the free surface of a gripped drop of magnetic liquid. Magnetohydrodynamics 19, 360.

- [8] A.O. Tsebers (1986). Numerical modeling of the dynamics of a drop of magnetizable liquid in constant and rotating magnetic fields. *Magnetohydrodynamics* 22, 345.
- [9] Baygents, J. C., Rivette, N. J., & Stone, H. A. (1998). Electrohydrodynamic deformation and interaction of drop pairs. *Journal of Fluid Mechanics*, 368, 359–375. doi:10.1017/s0022112098001797
- [10] Lac, E., & Homsy, G. M. (2007). Axisymmetric deformation and stability of a viscous drop in a steady electric field. *Journal of Fluid Mechanics*, 590. doi:10.1017/s0022112007007999
- [11] Afkhami, S., Tyler, A. J., Renardy, Y., Renardy, M., St. Pierre, T. G., Woodward, R. C., & Riffle, J. S. (2010). Deformation of a hydrophobic ferrofluid droplet suspended in a viscous medium under uniform magnetic fields. *Journal of Fluid Mechanics*, 663, 358–384. doi:10.1017/s0022112010003551
- [12] Garton, C. G., & Krasucki, Z. (1964). Bubbles in Insulating Liquids: Stability in an Electric Field. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 280(1381), 211–226. doi:10.1098/rspa.1964.0141
- [13] Stone, H. A., Lister, J. R., & Brenner, M. P. (1999). Drops with conical ends in electric and magnetic fields. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 455(1981), 329–347. doi:10.1098/rspa.1999.0316
- [14] Sherwood, J. D. (1988). Breakup of fluid droplets in electric and magnetic fields. *Journal of Fluid Mechanics*, 188(-1), 133. doi:10.1017/s0022112088000667
- [15] Ghaffari, A., Hashemabadi, S. H., & Bazmi, M. (2015). CFD simulation of equilibrium shape and coalescence of ferrofluid droplets subjected to uniform magnetic field. *Colloids and Surfaces A: Physicochemical and Engineering Aspects*, 481, 186–198. doi:10.1016/j.colsurfa.2015.04.038

- [16] Zinchenko, A. Z., Rother, M. A., & Davis, R. H. (1997). A novel boundary-integral algorithm for viscous interaction of deformable drops. *Phys. Fluids*, 9(6), 1493. doi:10.1063/1.869275
- [17] Pozrikidis, C. (2001). Interfacial Dynamics for Stokes Flow. *Journal of Computational Physics*, 169(2), 250–301. doi:10.1006/jcph.2000.6582
- [18] Cristini, V., Bławdziewicz, J., & Loewenberg, M. (2001). An Adaptive Mesh Algorithm for Evolving Surfaces: Simulations of Drop Breakup and Coalescence. *Journal of Computational Physics*, 168(2), 445–463. doi:10.1006/jcph.2001.6713
- [19] Zinchenko, A. Z., Rother, M. A., & Davis, R. H. (1999). Cusping, capture, and breakup of interacting drops by a curvatureless boundary-integral algorithm. *Journal of Fluid Mechanics*, 391, 249–292. doi:10.1017/s0022112099005285
- [20] Velankar, S., Zhou, H., Jeon, H. K., & Macosko, C. W. (2004). CFD evaluation of drop retraction methods for the measurement of interfacial tension of surfactant-laden drops. *Journal of Colloid and Interface Science*, 272(1), 172–185. doi:10.1016/j.jcis.2003.09.030
- [21] Heckbert, P. S., & Garland, M. (1999). Optimal triangulation and quadric-based surface simplification. *Computational Geometry*, 14(1-3), 49–65. doi:10.1016/s0925-7721(99)00030-9
- [22] Boissonnat, J.-D., & Oudot, S. (2006). Provably good sampling and meshing of Lipschitz surfaces. *Proceedings of the Twenty-Second Annual Symposium on Computational Geometry - SCG '06*. doi:10.1145/1137856.1137906
- [23] Botsch, M., & Kobelt, L. (2004). A remeshing approach to multiresolution modeling. *Proceedings of the 2004 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing - SGP '04*. doi:10.1145/1057432.1057457
- [24] Persson, P.-O., & Strang, G. (2004). A Simple Mesh Generator in MATLAB. *SIAM Review*, 46(2), 329–345. doi:10.1137/s0036144503429121

- [25] Persson P.O. Mesh Generation for Implicit Geometries (2005). Ph.D. thesis Massachusetts Institute of Technology
- [26] Jin, S., Lewis, R. R., & West, D. (2005). A comparison of algorithms for vertex normal computation. *Visual Comput.*, 21(1-2), 71–82. doi:10.1007/s00371-004-0271-1
- [27] Brochu, T., & Bridson, R. (2009). Robust Topological Operations for Dynamic Explicit Surfaces. *SIAM Journal on Scientific Computing*, 31(4), 2472–2493. doi:10.1137/080737617
- [28] Zinchenko, A. Z., & Davis, R. H. (2013). Emulsion flow through a packed bed with multiple drop breakup. *Journal of Fluid Mechanics*, 725, 611–663. doi:10.1017/jfm.2013.197
- [29] Muldowney, G. P., & Higdon, J. J. L. (1995). A spectral boundary element approach to three-dimensional Stokes flow. *Journal of Fluid Mechanics*, 298(-1), 167. doi:10.1017/s0022112095003260
- [30] Pozrikidis, C., & Chen, J.-T. (2003). Practical Guide to Boundary Element Methods with the Software Library BEMLIB. *Appl. Mech. Rev.*, 56(3), B45. doi:10.1115/1.1566406
- [31] Sakuma, T., Schneider, S., & Yasuda, Y. (n.d.). Fast Solution Methods. Computational Acoustics of Noise Propagation in Fluids - Finite and Boundary Element Methods, 333–366. doi:10.1007/978-3-540-77448-8\_13
- [32] Atkinson, K. E., & Chien, D. (1995). Piecewise Polynomial Collocation for Boundary Integral Equations. *SIAM Journal on Scientific Computing*, 16(3), 651–681. doi:10.1137/0916040
- [33] Dunavant, D. A. (1985). High degree efficient symmetrical Gaussian quadrature rules for the triangle. *Int. J. Numer. Meth. Engng.*, 21(6), 1129–1148. doi:10.1002/nme.1620210612

- [34] Guiggiani, M., & Gigante, A. (1990). A General Algorithm for Multidimensional Cauchy Principal Value Integrals in the Boundary Element Method. *J. Appl. Mech.*, 57(4), 906. doi:10.1115/1.2897660
- [35] Willis, D., Peraire, J., & White, J. (2006). A Quadratic Basis Function, Quadratic Geometry, High Order Panel Method. 44th AIAA Aerospace Sciences Meeting and Exhibit. doi:10.2514/6.2006-1253
- [36] Karafiat, A. (1996). Adaptive integration techniques for almost singular functions in the boundary element method. *Computers & Mathematics with Applications*, 32(5), 11–30. doi:10.1016/0898-1221(96)00132-0
- [37] Brebbia, C. A. (Ed.). (1989). Electromagnetic Applications. doi:10.1007/978-3-642-83680-0
- [38] Pozrikidis, C. (2000). Theoretical and computational aspects of the self-induced motion of three-dimensional vortex sheets. *Journal of Fluid Mechanics*, 425, 335–366. doi:10.1017/s0022112000002202
- [39] Luciani, A., Champagne, M. F. and Utracki, L. A. (1997), Interfacial tension coefficient from the retraction of ellipsoidal drops. *J. Polym. Sci. B Polym. Phys.*, 35: 1393–1403. doi:10.1002/(SICI)1099-0488(19970715)35:9<1393::AID-POLB9>3.0.CO;2-N
- [40] Stratton, J. A. (1941). Electromagnetic Theory. doi:10.1002/9781119134640
- [41] Pozrikidis, C. (1992). Boundary Integral and Singularity Methods for Linearized Viscous Flow. doi:10.1017/cbo9780511624124
- [42] Tanaka, M., Sladek, V., & Sladek, J. (1994). Regularization Techniques Applied to Boundary Element Methods. *Appl. Mech. Rev.*, 47(10), 457. doi:10.1115/1.3111062
- [43] Adriaens, J. P., Delince, F., Dular, P., Genon, A., Legros, W., & Nicole, A. (1991). Vector potential boundary element method for three di-

imensional magnetostatic. IEEE Transactions on Magnetics, 27(5), 3808–3810.

doi:10.1109/20.104931