

LATVIJAS UNIVERSITĀTE
FIZIKAS UN MATEMĀTIKAS FAKULTĀTE
MATEMĀTIKAS NODAĻA

**CILINDRU VIRKNES APTECĒŠANAS MATEMĀTISKAIS
MODELIS**

MAĢISTRA DARBS

Autors: **Maksims Marinaki**

Stud. apl. mm07026

Darba vadītājs: prof. Dr.math. Harijs Kalis

RĪGA 2013

Anotācija

Darbā tiek apskatīts bezgalīgas cilindru virknes aptecēšanas stacionārais uzdevums magnētiskā lauka iedarbībā divās dimensijās. Viskoza nespīējama šķidruma plūsmu apraksta Psi-Zēta sistēma. Iegūtas attiecīgo eliptisko robežproblēmu vājās nostādnes un uzkonstruēts iterāciju process, katrā solī pielietojot galīgo elementu metodi.

Atslēgas vārdi: MHD, Navjē-Stoksa vienādojumi, psi-zēta sistēma, galīgo elementu metode

Abstract

In this thesis the two-dimensional steady viscous incompressible magnetohydrodynamical flow past infinite rows of cylinders is considered. The flow is modeled by using the vorticity-stream Navier-Stokes equations' reformulation. The discrete versions of weak formulations for individual elliptic problems are derived and the iterative method by using linear finite elements is constructed.

Keywords: MHD, Navier-Stokes equations, stream-vorticity formulation, finite element method

Saturs

Apzīmējumi	3
Ievads	5
1. Modeļa izveide	7
1.1. Cilindru virkne	9
2. Psi-Zēta sistēmas analīze	10
3. Parciālo diferenciālvienādojumu eliptiskais tips	11
3.1. Problēmas eliptiska tipa vienādojumiem	12
4. Eliptisko robežproblēmu atrisināmībā telpā \mathcal{H}^1	13
4.1. Robežproblēmu vājās nostādnes	14
4.2. Vispārināta atrisinājuma eksistence un unitāte	18
4.3. Vispārināta atrisinājuma stabilitāte	21
5. Problēmu diskretizācija	22
5.1. Gaļorkina metode telpā \mathcal{H}^1	22
5.2. Galīgo elementu telpa	23
5.3. Lineārie Lagranža elementi	26
6. Korekti formulēta problēma	32
6.1. Robežnosacījumi uz cietām virsmām funkcijai ζ	34
6.2. Psi-Zēta sistēmas vienādojumu atrisināmība	35
7. Iterāciju process	37
8. Galīgo elementu metodes realizācija	39
8.1. Nevienmērīgais režģis	39
8.2. Afīnās transformācijas	40
8.3. Matricu elementu atrašana	43
9. Rezultātu grafiskā attēlošana	45
Secinājumi	49

Izmantotā literatūra un avoti	50
A Pielikums	51
A Izveidoto programmu kods	53

Apzīmējumi

\mathbf{V} vektoriāls lielums,

V_x, V_y, V_z vektora komponentes,

U skalārs lielums,

$\mathbf{V}_1 \cdot \mathbf{V}_2$ skalārais reizinājums,

$\mathbf{V}_1 \times \mathbf{V}_2$ vektoriālais reizinājums,

$\nabla \cdot \mathbf{V}$ vektoru lauka diverģence,

∇u skalāra lauka gradients,

$\frac{\partial u}{\partial x}$, u_x parciālais atvasinājums,

$\det A$ matricas A determinants,

b^T vektors-rindiņa (b_1, \dots, b_n) ,

$\partial\Omega$ apgabala Ω robeža,

Δu Laplasa operators,

$\bar{\Omega}$ kopas Ω slēgums,

$C^n(\Omega)$ n reizes nepārtraukti diferencējamo funkciju telpa,

$\sup_x f(x)$ funkcijas suprēms,

$\inf_x f(x)$ funkcijas infīms,

$\dim H$ telpas H dimensija,

A^{-1} matricas A inversā matrica,

$\text{int}\Omega$ kopas Ω iekšiene,

$\Omega_1 \cup \Omega_2$ kopu apvienojums,

$\Omega_1 \cap \Omega_2$ kopu šķēlums,

\emptyset tukšā kopa,

$f = O(g)$ vienādas kārtas lielumi,

δ_{ij} Kronekera simbols.

Ievads

Maģistra darbā tiek aplūkots cilindru virknes aptecēšanas uzdevums divās dimensijās viskozam nespiežamam šķidrumam magnētiskā lauka iedarbībā.

Darba mērķis ir izanalizēt gan šo konkrētu uzdevumu, gan galvenus teorijas aspektus, jo, uzdevumu risinot, rodas vajadzība apskatīt dažādu tipu robežnosacījumus, tādus kā: homogēnus un nehomogēnus Dirihlē nosacījumus, Neimana, periodiskuma nosacījumus.

1. nodaļā tiek formulēts matemātiskais modelis, kas balstās uz fizikas likumiem: Navjē-Stoksa vienādojumu sistēma. Šajā nodaļā diezgan sarežģīti uzdots modelis tiek vienkāršots: nezināmo lielumu skaits tiek samazināts, ievesti bezdimensiju mainīgie. Definēts galvenais pētāmais objekts šajā darba: Psi-Zēta sistēma (1.7). Tiek uzdots arī uzdevuma ģeometrija.

2. nodaļā notiek situācijas analīze: izejot no ģeometrijas un vienādojumu tipiem Psi-Zēta sistēmā (1.7), pamatota tuvināto metožu izvēle.

3. nodaļā ir dots ieskats klasiskajā eliptisko parciālo diferenciālvienādojumu teorijā. Ir definēti jēdzieni: Dirihlē, Neimana un periodiskā robežproblēma.

4. nodaļas mērķis ir iegūt vājos formulējumus problēmām, kas satur vispārīgo otrās kārtas eliptisko operatoru: Dirihlē, Neimana un periodiskai robežproblēmām.

Gandrīz visos izmantotos avotos:[1],[2],[3], tehnika ir apmēram līdzīga, bet bieži vien, formulējums iegūts tikai kādam speciālgadījumam. Lielā daļā no grāmatām: [1],[2],[3],[4], arī vispārināti formulētas problēmas korektība ir pamatota vienādā veidā: pierādījumi balstās uz operatoru īpašībām, kas ir līdzīgas tām, kas piemita sākotnējām diferenciāloperatoram. Tieši tāpēc tāda tipa pierādījumus ir viegli adaptēt un problēmu korektība ir pamatota apakšnodaļās 4.2. un 4.3..

5. nodaļā ir parādīts process, kā var nonākt pie lineāras vienādojumu sistēmas, apskatot vispārināti formulētu problēmu kādā galīgā telpā: Gaļorkina metode. Uzmanība tiek koncentrēta uz lineāro galīgo elementu ieviešanu bāzes funkciju lomā un to definēšanu uz vienmērīgas un nevienmērīgas triangulācijas. Vēl viens nodaļas mērķis: iegūt formulas vajadzīgo koeficientu aprēķināšanai bāzes funkcijām un sastādīt algoritmu, kas pieļauj precizitātes palielināšanu.

Sākot ar 6. nodaļu, tiek turpināta konkrētas problēmas analīze. Izmantojot teorētiskus rezultātus, no iepriekšējām nodaļām (3., 4., 5.), formāli tiek uzdoti robežnosacījumi Psi-Zēta sistēmai (1.7), ir iegūta problēmas vājā nostādne, parādīta problēmas korektība.

Problēmas jaunais formulējums ir izveidots tādā veidā, lai varētu diezgan ātri sastādīt iterāciju procesu vienādojumu sistēmas risināšanai. Par to iet runa 7. nodaļā.

8. nodaļā tiek analītiski uzdots viss, kas vajadzīgs galīgo elementu koda sastādīšanai. Uzdots nevienmērīgais režģis, definētas afīnas transformācijas uz vienības šūnu, atrastas formulas galīgo elementu matricu aizpildīšanai.

9. nodaļā rezultāti tiek demonstrēti grafiski un tabulu veidā.

Pielikumā A. definēti darbā ietvertie, neizskaidrotie termini, izmantotās teorēmas un formulas; atrodams sastādīts galīgo elementu kods.

1. Modeļa izveide

Klasiskajā variantā viskoza nesaspiežama šķidruma plūsmu magnētiskā lauka iedarbībā apraksta Navjē-Stoksa vienādojumu sistēma [5]:

$$\rho\left(\frac{\partial\mathbf{V}}{\partial t} + \mathbf{V} \cdot \nabla\mathbf{V}\right) = -\nabla\mathbf{p} + \mu\Delta\mathbf{V} + \mathbf{F}. \quad (1.1)$$

Turklāt jābūt spēkā nepārtrauktības likumam:

$$\nabla \cdot \mathbf{V} = 0. \quad (1.2)$$

Šeit \mathbf{V} ir šķidruma ātrums, \mathbf{p} - spiediens, ρ - blīvums un μ - dinamiskā viskozitāte. No 'tīras' hidrodinamikas sistēmu (1.1),(1.2) atšķir saskaitāmais $\mathbf{F} = \mathbf{j} \times \mathbf{B}$ - elektromagnētiskais spēks. Kā redzams, tas ir vektors, perpendikulārs abiem diviem: \mathbf{j} - strāvas blīvumam, un \mathbf{B} - magnētiskā lauka indukcijai.

Maģistra darbā nolemts analizēt bezindukcijas tuvinājumu šai problēmai, kā arī apskatīt stacionāro gadījumu ($\frac{\partial\mathbf{V}}{\partial t} = 0$) divās dimensijās. Līdz ar to, 1. nodaļas mērķis ir šāds: iegūt pēc iespējas vienkāršāku sistēmu (1.1),(1.2) formulējumu Dekarta koordinātēs divās dimensijās.

Formulējumu vienkāršojot, jāsamazina nezināmo funkciju skaits sistēmā (1.1),(1.2), jāieved palīgmainīgie, palīglieļumi.

Sākumā jāuzzina, kā izskatās saskaitāmais \mathbf{F} Dekarta koordinātēs.

Vispārinātais Oma likums kustības gadījumam ir uzdots šādā veidā [6]:

$$\mathbf{j} = \sigma(\mathbf{E} + \mathbf{V} \times \mathbf{B}),$$

kur σ - elektriskā vadītspēja, \mathbf{E} - elektriskais lauks. Tāpēc spēka vektora komponentes uzdotas formā:

$$F_x = -\sigma B_y(V_x B_y - V_y B_x + E_z), \quad F_y = \sigma B_x(V_x B_y - V_y B_x + E_z),$$

Magnētiskā lauka indukcijas komponentes izskatās šādi:

$$B_x = \cos(\alpha), \quad B_y = \sin(\alpha),$$

kur α leņķis starp Ox - asi un indukcijas vektoru. Maģistra darbā tiek apskatīti divi tādi: $\alpha = 0$ un $\alpha = \frac{\pi}{2}$.

Tagad var uzrakstīt sistēmu (1.1),(1.2) Dekarta koordinātēs:

$$\begin{cases} \rho \frac{\partial V_x}{\partial x} V_x + \rho \frac{\partial V_x}{\partial y} V_y = -\frac{\partial p}{\partial x} + \mu \Delta V_x + F_x \\ \rho \frac{\partial V_y}{\partial x} V_x + \rho \frac{\partial V_y}{\partial y} V_y = -\frac{\partial p}{\partial y} + \mu \Delta V_y + F_y \\ \frac{\partial(V_x)}{\partial x} + \frac{\partial(V_y)}{\partial y} = 0. \end{cases} \quad (1.3)$$

Nākamais solis ir šāds:

Ievedam bezdimensiju mainīgus (apzīmējot vecos ar tildi):

$$\mathbf{V} := \frac{\tilde{\mathbf{V}}}{V_0}, \quad \mathbf{x} := \frac{\tilde{\mathbf{x}}}{L_0}, \quad \mathbf{p} := \frac{\tilde{\mathbf{p}}}{\rho V_0^2}, \quad \mathbf{F} := \frac{\tilde{\mathbf{F}}}{B_0^2 \sigma V_0}$$

kur V_0 - raksturīgais ātrums, L_0 - raksturīgais izmērs. $Re = \frac{V_0 L_0}{\nu}$, $S = \frac{\sigma B_0^2 L_0}{\rho V_0}$ ir Reinoldsa un Stjūarta skaitļi, $\nu := \frac{\mu}{\rho}$ - kinemātiskā viskozitāte, B_0 - raksturīgā indukcija.

Ieved arī lielumus: $\zeta := \frac{\partial V_y}{\partial x} - \frac{\partial V_x}{\partial y}$ - virpuļu funkcija,

$$\bar{\mathbf{p}} := \mathbf{p} + \frac{1}{2} \begin{pmatrix} V_x^2 + V_y^2 \\ V_x^2 + V_y^2 \end{pmatrix} - \begin{pmatrix} S E_z A_z \\ S E_z A_z \end{pmatrix}. \quad (1.4)$$

Šeit $A_z := y \cos(\alpha) - x \sin(\alpha)$. Diferencējot šo funkciju (kopā ar citiem 'modificētā' spiediena (1.4) saskaitāmiem) pēc telpas mainīgajiem, iegūst vajadzīgo.

Tādā gadījumā spēka komponentes arī mazliet vienkāršojas:

$$\bar{F}_x = -S(V_x \sin^2 \alpha + V_y \sin \alpha \cos \alpha),$$

$$\bar{F}_y = S(V_x \sin \alpha \cos \alpha - V_y \cos^2 \alpha).$$

Iegūst sistēmu:

$$\begin{cases} -\zeta V_y = -\frac{\partial \bar{p}}{\partial x} + Re^{-1} \Delta V_x + S \bar{F}_x, \\ \zeta V_x = -\frac{\partial \bar{p}}{\partial y} + Re^{-1} \Delta V_y + S \bar{F}_y, \\ \frac{\partial(V_x)}{\partial x} + \frac{\partial(V_y)}{\partial y} = 0, \end{cases} \quad (1.5)$$

Lai samazinātu nezināmo funkciju skaitu sistēmā (1.5), var izdarīt vēl sekojošas lietas:

1. Pirmo vienādojumu sistēmā (1.5) diferencē pēc mainīga y . Iegūst:

$$-\zeta \frac{\partial V_y}{\partial y} - \frac{\partial \zeta}{\partial y} V_y = -\frac{\partial^2 \bar{p}}{\partial x \partial y} + Re^{-1} \left(\frac{\partial^3 V_x}{\partial x^2 y} + \frac{\partial^3 V_x}{\partial y^3} \right) + S \frac{\partial F_x}{\partial y},$$

2. No iegūtas izteiksmes atņem otro vienādojumu, atvasinātu pēc mainīga x . Ņemot vērā nepārtrauktības vienādojuma (trešā vienādojuma sistēmā (1.5)) izpildi, iegūst:

$$-\cancel{\zeta} \frac{\partial V_y}{\partial y} - \cancel{\zeta} \frac{\partial V_x}{\partial x} - \frac{\partial \zeta}{\partial y} V_y - \frac{\partial \zeta}{\partial x} V_x = -\frac{\partial^2 \bar{p}}{\partial x \partial y} + \frac{\partial^2 \bar{p}}{\partial y \partial x} + Re^{-1} \left(\frac{\partial^3 V_x}{\partial x^2 y} + \frac{\partial^3 V_x}{\partial y^3} - \frac{\partial^3 V_y}{\partial x^3} - \frac{\partial^3 V_y}{\partial y^2 x} \right) +$$

$$+S \frac{\partial F_x}{\partial y} - S \frac{\partial F_y}{\partial x}.$$

3. Izmantojot faktus, ka $(\frac{\partial^3 V_x}{\partial x^2 y} + \frac{\partial^3 V_x}{\partial y^3} - \frac{\partial^3 V_y}{\partial x^3} - \frac{\partial^3 V_y}{\partial y^2 x}) = -\Delta \zeta$ un

$$-S \frac{\partial F_x}{\partial y} + S \frac{\partial F_y}{\partial x} = \frac{\partial V_x}{\partial x} \sin \alpha \cos \alpha - \frac{\partial V_y}{\partial x} \cos^2 \alpha + \frac{\partial V_x}{\partial y} \sin^2 \alpha - \frac{\partial V_y}{\partial y} \sin \alpha \cos \alpha,$$

pēdējais solis ir potenciāla jeb strāvas funkcijas ieviešana:

$$V_x = \frac{\partial \psi}{\partial y}, \quad V_y = -\frac{\partial \psi}{\partial x}, \quad -\Delta \psi = \zeta, \quad (1.6)$$

un nezināmo funkciju skaita sistēmā (1.5) samazināšana līdz divām:

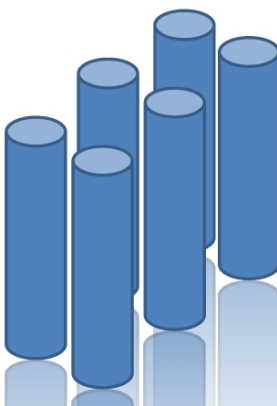
$$\begin{cases} Re^{-1} \Delta \zeta + J(\psi, \zeta) + S f = 0, \\ -\Delta \psi = \zeta, \end{cases} \quad (1.7)$$

$$\text{kur } J(\psi, \zeta) = \frac{\partial \psi}{\partial x} \frac{\partial \zeta}{\partial y} - \frac{\partial \psi}{\partial y} \frac{\partial \zeta}{\partial x}, \quad f = \nabla \cdot (\tilde{A} \nabla \psi), \quad \tilde{A} = \begin{pmatrix} \sin^2 \alpha & \sin \alpha \cos \alpha \\ \sin \alpha \cos \alpha & \cos^2 \alpha \end{pmatrix}.$$

Definīcija 1. Sistēmu (1.7) sauc par Psi-Zēta sistēmu.

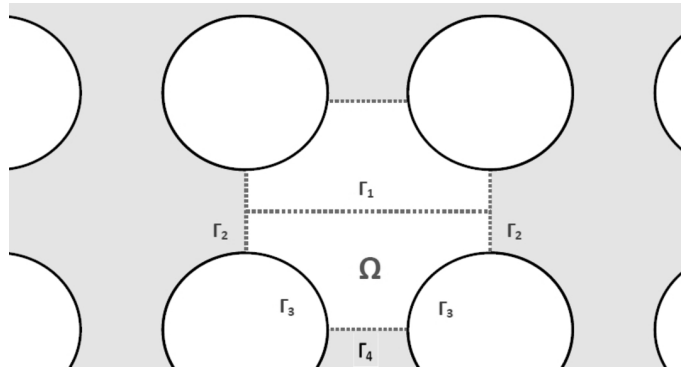
1.1. Cilindru virkne

Līdz šim nebija skaidrs, kāda veida apgabals ir uzdots telpā \mathbb{R}^2 Dekarta koordinātēs, kurā ir uzdota sistēma (1.7). Maģistra darbā ir nolemts analizēt bezgalīgo cilindru virkni divās dimensijās. Trijās dimensijās virknes fragments parādīts 1. attēlā:



1. att. Cilindru virknes fragments

Uz plaknes virkne izskatītos šādi (2. attēls, apskatīts šķērsgriezums):



2. att. Bezgalīga cilindru virkne divās dimensijās

Lai visu bezgalīgo apgabalu izanalizētu, pietiek izdalīt galīgu apakšapgabalu Ω (2. attēls) un uzdot simetrijas un periodiskuma robežnosacījumus. Pagaidām neuzdodot tādu apgabalu un atbilstošus robežnosacījumus koordinātēs, var vienkārši, skatoties uz 2. attēlu izsecināt, ka par simetrijas asīm derētu izvēlēties Γ_1 un Γ_4 . Periodiskumam jāizpildās uz apgabala robežas Γ_2 . Uz lokiem Γ_3 jāuzdod nosacījumi, piemēroti cietām virsmām. Formāli tas tiks izdarīts 6. nodaļā.

2. Psi-Zēta sistēmas analīze

Tā kā tagad ir skaidrs, kāda tipa ģeometrija ir izvēlētajam uzdevumam un kā izskatās diferenciālvienādojumi, kas apraksta fizikālo problēmu, ir pienācis laiks pieņemt lēmumu, kādas metodes tiks izmantotas sistēmas (1.7) atrisinājuma atrašanai.

Sistēma (1.7) ir divu lineāru eliptisku vienādojumu sistēma (katrs eliptisks attiecībā pret vienu no nezināmām funkcijām. Otru funkciju šajā gadījumā uzskata par fiksētu). Eliptiskums un linearitāte tiks formāli parādīti 4. nodaļā, bet pagaidām vajag saprast sekojošo:

1. Mērķis ir iegūt skalāru lauku ψ , jeb strāvas funkcijas vērtības apgabala Ω punktos. Vajadzības gadījumā no vektora potenciāla sadalījuma ir jānāk dabūt arī ātruma vektoru lauks \mathbf{V} . Tas nozīmē, ka šo problēmu skaitliski analizējot, pilnībā pietiktu ar lineāro aproksimāciju. Pirmais vienādojums sistēmā (1.7) satur otras kārtas parciālos atvasinājumus (tā skaitā arī jauktos). Tā kā šo sistēmu apskata kā eliptisko robežproblēmu virkni, visdrīzāk ir jākonstruē iterāciju process. Lai katrā solī tādu

problēmu risinātu, jāiegūst attiecīgo robežproblēmu vajās nostādnes kas ļaus atvieglot prasību par otras kārtas nepārtrauktu parciālo atvasinājumu eksistenci līdz pirmās kārtas vispārināto atvasinājumu eksistencei.

2. Otrkārt, uzdevumam ir diezgan sarežģīta ģeometrija. Analītiska atrisinājuma atrašana, pat kādam vienkāršotam sistēmas (1.7) variantam būtu ļoti grūts vai pat neiespējams uzdevums. Universāla metode, kas tiek galā ar grūtībām, kurus rada šādas ģeometrijas ir *galīgo elementu metode* (5.), kas labi saskaņojas ar robežproblēmu vājam nostādnēm. Lineāru elementu izvēle šajā gadījumā būtu pilnīgi pietiekoša.

Tagad tiek piedāvātas 3 nodaļas, kurās ir izklāstīta eliptisku robežproblēmu teorija, jo daži operatori sistēmā (1.7) ir eliptiskie operatori, uzdoti vispārīgā formā, kā arī galīgo elementu metodes, kuru var pielietot pakāpeniski katrā solī vienai eliptiskai problēmai, pamataspekti.

3. Parciālo diferenciālvienādojumu eliptiskais tips

Apskatīsim otrās kārtas lineāru parciālo diferenciālvienādojumu

$$-(a(x, y)u_{xx} + 2b(x, y)u_{xy} + c(x, y)u_{yy}) + d(x, y)u_x + e(x, y)u_y + f(x, y)u = g(x, y), \quad (3.1)$$

kas matricu-vektoru formā pārrakstāms šādi:

$$Lu = -\nabla \cdot (A\nabla u) + b^T \nabla u + f(x, y)u = g(x, y), \quad (3.2)$$

kur $(x, y) \in \Omega \subset \mathbb{R}^2$, $A = \begin{pmatrix} a(x, y) & b(x, y) \\ b(x, y) & c(x, y) \end{pmatrix}$, $b = \begin{pmatrix} d(x, y) \\ e(x, y) \end{pmatrix}$.

Ir pieņemts, ka matricas A un vektora b elementi, funkcijas f un g - apgabalā Ω ierobežotas funkcijas.

Definīcija 2. [2] Vienādojums (3.2) ir eliptiska tipa vienādojums punktā (x, y) , ja $\det A > 0$ punktā (x, y) .

Definīcija 3. [2] Vienādojums (3.2) ir eliptiska tipa vienādojums apgabalā Ω , ja $\det A > 0$ $\forall (x, y) \in \Omega$.

Definīcija 4. [2] Lineārs diferenciāloperators Lu ir eliptisks, ja matrica A ir simetriska un pozitīvi definita $\forall (x, y) \in \Omega$.

3.1. Problēmas eliptiska tipa vienādojumiem

Definīcija 5. [2] Problēmu

$$\begin{cases} Lu = -\nabla \cdot (A\nabla u) + b^T \nabla u + f(x, y)u = g(x, y), \Omega \\ u = 0, \quad \partial\Omega \end{cases} \quad (3.3)$$

sauc par Dirihlē problēmu.

Definīcija 6. [2] Problēmu

$$\begin{cases} Lu = -\nabla \cdot (A\nabla u) + b^T \nabla u + f(x, y)u = g(x, y), \Omega \\ A\nabla u \cdot \hat{n} = h(x, y), \quad \partial\Omega, \end{cases} \quad (3.4)$$

kur \hat{n} - apgabala Ω ārējā normāle, sauc par Neimana problēmu.

Dažreiz uz apgabala Ω divām paralēlām malām var būt uzdoti arī *periodiskuma nosacījumi*.

Definīcija 7. Par periodiskiem robežnosacījumiem attiecībā uz paralēlām malām Γ_i un Γ_j , x vai y ass virzienā, sauc nosacījumus formā

$$u(x, y)|_{\Gamma_i} = u(x, y)|_{\Gamma_j}, A\nabla u(x, y) \cdot \hat{n}|_{\Gamma_i} = A\nabla u(x, y) \cdot \hat{n}|_{\Gamma_j}. \quad (3.5)$$

Definīcija 8. Problēmu

$$\begin{cases} Lu = -\nabla \cdot (A\nabla u) + b^T \nabla u + f(x, y)u = g(x, y), \Omega \\ u(x, y)|_{\Gamma_i} = u(x, y)|_{\Gamma_j}, \\ A\nabla u(x, y) \cdot \hat{n}|_{\Gamma_i} = A\nabla u(x, y) \cdot \hat{n}|_{\Gamma_j}, \\ u = 0, \quad \partial\Omega \setminus \Gamma_i \setminus \Gamma_j. \end{cases} \quad (3.6)$$

kur \hat{n} - apgabala Ω ārējā normāle, sauc par periodisko problēmu.

Piezīme 1. Pieņemts, ka nehomogenitāte ir problēmas (3.3) vienādojumā un robežnosacījumi doti homogēni. Vajadzības gadījumā, Dirihlē problēmu ar nehomogēniem nosacījumiem var reducēt uz šādu formu.

Piezīme 2. Pieņemts, ka $\partial\Omega$ - Nepārtraukta pēc Lipšica apgabala Ω robeža (A, 28).

4. Eliptisko robežproblēmu atrisināmībā telpā \mathcal{H}^1

Robežproblēmu vispārinātās nostādnes dod iespēju apskatīt atrisinājumus, kuri ir meklējami funkciju telpās, kas neatbilst klasisko atrisinājumu teorijai (klasiskā gadījumā, $u \in C^2(\Omega) \cap C^0(\overline{\Omega})$ Dirihlē problēmas (3.3) gadījumā un $u \in C^2(\Omega) \cap C^1(\overline{\Omega})$ Neimana problēmas (3.4) un periodiskas problēmas (3.6) gadījumos, [2]).

Mērķis ir pārrakstīt problēmas formulējumu operatoru vienādojuma formā $a(u, v) = f(v)$, precizējot funkciju u un v telpas.

Definīcija 9. Funkciju telpu H reizinājumā $H \times H$ definētu attēlojumu $a(u, v) : H \times H \rightarrow \mathbb{R}$ sauc par *bilineāru formu*, ja ir spēkā šādas aksiomas:

$\forall \alpha \in \mathbb{R} \forall u, v, u_1, v_1 \in H :$

1. $a(\alpha u, v) = a(u, \alpha v) = \alpha a(u, v)$,
2. $a(u + u_1, v) = a(u, v) + a(u_1, v)$,
3. $a(u, v + v_1) = a(u, v) + a(u, v_1)$.

Citiem vārdiem, operatoram $a(u, v)$ jābūt lineāram pēc abiem argumentiem.

Definīcija 10. Funkciju telpā H definētu attēlojumu $f(v) : H \rightarrow \mathbb{R}$ sauc par *lineāru funkcionāli*, ja ir spēkā aksiomas:

$\forall \alpha \in \mathbb{R} \forall v, v_1 \in H :$

1. $f(\alpha v) = \alpha f(v)$,
2. $f(v + v_1) = f(v) + f(v_1)$.

Definīcija 11. Problēmu, kas skan šādi:

Atrast funkciju $u \in H$, kas apmierina vienādojumu

$$a(u, v) = f(v) \quad \forall v \in G,$$

kur $a(u, v)$ - bilineārā forma, $f(v)$ - lineārs funkcionālis, sauc par matemātiskās fizikas problēmas *vājo* jeb *vispārināto nostādni*.

Tāda tipa problēmai literatūrā var atrast dažādus rezultātus, kas garantē tās atrisināmību un atrisinājuma unitāti pie zināmiem nosacījumiem uz problēmas vienādojuma labo un kreiso pusi noteiktās funkciju telpās. Diezgan reti tas tiek pierādīts eliptiskam operatoram, uzdotam vispārīgā formā (kā Psi-Zētas sistēmas gadījumā) - parasti tas tiek izdarīts tikai speciālgadījumam - Laplasa operatoram. Tāpēc nākamajās apakšnodaļās uzmanība tiek veltīta tieši vispārīgam eliptiskam operatoram.

4.1. Robežproblēmu vājās nostādnes

Vājš formulējums tiks iegūts robežproblēmām (3.3), (3.4) un (3.6). Šo problēmu vienādojuma

$$-\nabla \cdot (A\nabla u) + b^T \nabla u + f(x, y)u = g(x, y)$$

abas puses reizina ar funkciju $v(x, y)$. Funkciju $v : \Omega \rightarrow \mathbb{R}$ sauksim par testa funkciju, ja tā ir pietiekami gluda, jeb ja $v \in C^\infty(\Omega)$.

Iegūto izteiksmi

$$-\nabla \cdot (A\nabla u)v + b^T \nabla uv + f(x, y)uv = vg$$

integrē pa apgabalu Ω : ([1], [7])

$$-\int_{\Omega} \nabla \cdot (A\nabla u)v dx + \int_{\Omega} b^T \nabla uv dx + \int_{\Omega} f(x, y)uv dx = \int_{\Omega} vg dx. \quad (4.1)$$

Turklāt, vienādojuma (4.1) "galvenā daļa" $-\int_{\Omega} \nabla \cdot (A\nabla u)v dx$ tiek integrēta parciāli. Parciālas integrēšanas formula iegūta pārveidojot:

$$\begin{aligned} \nabla \cdot (vA\nabla u) &= \nabla \cdot \begin{pmatrix} v(a_{11}u_x + a_{12}u_y) \\ v(a_{21}u_x + a_{22}u_y) \end{pmatrix} = (v(a_{11}u_x + a_{12}u_y))_x + (v(a_{21}u_x + a_{22}u_y))_y = \\ &= v_x(a_{11}u_x + a_{12}u_y) + v(a_{11}u_{xx} + a_{12}u_{yx}) + v_y(a_{21}u_x + a_{22}u_y) + v(a_{21}u_{xy} + a_{22}u_{yy}) = \\ &= \nabla v^T A\nabla u + \nabla \cdot (A\nabla u)v. \end{aligned}$$

Integrējot pa apgabalu Ω , iegūst:

$$\int_{\Omega} \nabla \cdot (A\nabla u)v dx = \int_{\Omega} \nabla \cdot (vA\nabla u) dx - \int_{\Omega} \nabla v^T A\nabla u dx$$

Turklāt, diverģences teorēma (A, 10) dod:

$$\int_{\Omega} \nabla \cdot (vA\nabla u) dx = \int_{\partial\Omega} v(A\nabla u) n d\sigma,$$

kur n - apgabala Ω ārējā normāle. Līdz ar to, vienādojums (4.1) ir uzrakstāms šādi:

$$-\int_{\partial\Omega} v(A\nabla u)nd\sigma + \int_{\Omega} \nabla v^T A\nabla u dx + \int_{\Omega} b^T \nabla u v dx + \int_{\Omega} f(x, y) u v dx = \int_{\Omega} g v dx. \quad (4.2)$$

Tālāk svarīgu lomu spēlē robežnosacījumu tips. Dirihlē nosacījumu (3.3) gadījumā testa funkcijai ir prasība $v(x, y) \in C_0^1(\Omega)$. Šī testa funkciju klases apakšklase ietver sevī funkcijas, kas pieņem nulles vērtības ārpus vaļēja apgabala Ω (A, 29). Tādā gadījumā, pirmais saskaitāmais vienādojuma (4.2) kreisajā pusē ir nulle. Neimana nosacījumu (3.4) gadījumā, šim saskaitāmam ir svarīga loma un tāpēc netiek prasīts lai šī funkcija ir nulle uz apgabala robežas, proti $v(x, y) \in C^1(\Omega)$ un $-\int_{\partial\Omega} v(A\nabla u)nd\sigma = -\int_{\partial\Omega} h v d\sigma$.

Iegūts formulējums (4.2) atbilst formai $a(u, v) = f(v)$, kur:

1. Dirihlē problēmai (3.3), periodiskai problēmai (3.6):

$$\begin{aligned} a(u, v) &= \int_{\Omega} \nabla v^T A\nabla u dx + \int_{\Omega} b^T \nabla u v dx + \int_{\Omega} f(x, y) u v dx, \\ f(v) &= \int_{\Omega} v g dx, \end{aligned} \quad (4.3)$$

2. Neimana problēmai (3.4):

$$\begin{aligned} a(u, v) &= \int_{\Omega} \nabla v^T A\nabla u dx + \int_{\Omega} b^T \nabla u v dx + \int_{\Omega} f(x, y) u v dx, \\ f(v) &= \int_{\Omega} g v dx + \int_{\partial\Omega} h v d\sigma, \end{aligned} \quad (4.4)$$

Piezīme 3. Spēkā linearitātes un bilinearitātes aksiomas objektiem $f(v)$ un $a(u, v)$, kas seko uzreiz no integrāļa īpašībām.

Funkciju telpas

Matemātiskās fizikas problēmu, kas tika apskatītas 3. nodaļā, vienādojumi tagad ir pārrakstīti vajadzīgā formā. Otrais solis vispārināta formulējuma iegūšanai - norādīt funkciju telpas, kurās problēmas atrisinājums u un testa funkcija v ir meklējamas. Abos gadījumos (4.3),(4.4) lielumiem $a(u, v)$ un $f(v)$ jābūt galīgiem. Tāpēc tiek prasīts lai funkcijas u un v , kā arī to parciālie atvasinājumi būtu funkcijas no kvadrātiski integrējamu funkciju klases.

Definīcija 12. [3] Pie $1 \leq p < \infty$, telpu $L^p(\Omega)$ definē šādi:

$$L^p(\Omega) = \{u : \int_{\Omega} |u(x)|^p dx < \infty\}.$$

Telpā $L^p(\Omega)$ definē normu

$$\|u\|_{L^p} := \|u\|_{0,p} := \left(\int_{\Omega} |u(x)|^p dx \right)^{\frac{1}{p}}$$

Telpa $L^2(\Omega)$ ir Hilberta telpa, ja tajā ir definēts skalārs reizinājums

$$(u, v)_{L^2} := (u, v)_0 := \int_{\Omega} uv dx.$$

Definīcija 13. [3] Funkciju $w \in L^2(\Omega)$ sauc par funkcijas $u \in L^2(\Omega)$ vispārināto atvasinājumu, ja

$$(u, \partial^\alpha v)_0 = (-1)^{|\alpha|} (w, v)_0 \text{ katrai testa funkcijai } v \in C_0^\infty(\Omega),$$

kur $\alpha = (\alpha_1, \dots, \alpha_n)$ - multiindekss, $|\alpha| := \sum_{i=1}^n \alpha_i$, $\partial^\alpha v := \frac{\partial^{\alpha_1}}{\partial x_1^{\alpha_1}} \dots \frac{\partial^{\alpha_n}}{\partial x_n^{\alpha_n}}$.

Tas, savukārt, nozīmē, ka funkcijas u atvasinājumi $\partial^\alpha u$ var arī neeksistēt tiešā veidā, bet tikai zem integrāļa un attiecībā pret testa funkcijām. Funkcijas atvasinājuma $\partial^\alpha u$ vispārinājumu apzīmē ar $D^\alpha u$.

Definīcija 14. [3] Pie $m > 0$ un $p \geq 1$, Soboļeva telpu $W^{m,p}(\Omega)$ definē šādi:

$$W^{m,p}(\Omega) = \{u : u \in L^p(\Omega), D^\alpha u \in L^p(\Omega), |\alpha| \leq m\}.$$

Telpā $W^{m,p}(\Omega)$ definē normu:

$$\|u\|_{m,p} := \left(\sum_{|\alpha| \leq m} \|D^\alpha u\|_{0,p}^p \right)^{\frac{1}{p}}$$

Telpa $W^{m,2}(\Omega)$ ir Hilberta telpa, ja tajā ir definēts skalārs reizinājums

$$(u, v)_m := \sum_{|\alpha| \leq m} (D^\alpha u, D^\alpha v)_0,$$

un to apzīmē ar $H^m(\Omega) := W^{m,2}(\Omega)$.

Definīcija 15. [1] Telpu $H_0^1(\Omega)$ definē šādi:

$$H_0^1(\Omega) = \{u : u \in H^1(\Omega); u = 0, \partial\Omega\}.$$

Telpai H^1 pieder kvadrātiski integrējamas funkcijas kopā ar visiem saviem vispārinātiem daļējiem atvasinājumiem, kuriem arī jābūt kvadrātiski integrējamām funkcijām, un pieņēmums, ka problēmu (4.3) un (4.4) atrisinājumi pieder šai telpai garantē lielumu $a(u, v)$ un $f(v)$ eksistenci. Dirihlē problēmas vispārināta formulējuma (4.3) atrisinājumu ir vērts meklēt šīs telpas apakštelpā H_0^1 , jo šajā gadījumā robežnosacījumi $u = 0, \partial\Omega$ ir iekļauti telpas definīcijā. Neimana problēmas vispārināta formulējuma (4.4) atrisinājumam jāpieņem telpai H^1 un robežnosacījumi šajā gadījumā ir iekļauti funkcionāla $f(v)$ izteiksmē. Parasti pieņem, ka testa funkciju telpa sakrīt ar atrisinājuma telpu ([1])

Vienādojumam $a(u, v) = f(v)$ tagad var precizēt atrisinājuma un testa funkciju telpas. Faktiski tika iegūtas divas lietas:

1. Dirihlē problēmas (3.3) vājā nostādne:

Atrast funkciju $u \in H_0^1(\Omega)$, kas apmierina vienādojumu

$$a(u, v) = f(v) \quad \forall v \in H_0^1, \tag{4.5}$$

kur $a(u, v), f(v)$ ir formā (4.3).

2. Neimana problēmas (3.4) vājā nostādne:

Atrast funkciju $u \in H^1(\Omega)$, kas apmierina vienādojumu

$$a(u, v) = f(v) \quad \forall v \in H^1, \tag{4.6}$$

kur $a(u, v), f(v)$ ir formā (4.4).

Pēc Dirihlē un Neimana robežproblēmu parauga var izvest arī formulējumus periodiskām robežproblēmām. Kā jau iepriekš bija definēts, (3.6), periodiskie nosacījumi ir funkcijas vērtību un funkcijas atvasinājuma periodiskums attiecībā uz kādām divām paralēlām malām:

$$u(x, y)|_{\Gamma_i} = u(x, y)|_{\Gamma_j}, A\nabla u(x, y) \cdot \hat{n}|_{\Gamma_i} = A\nabla u(x, y) \cdot \hat{n}|_{\Gamma_j}. \quad (4.7)$$

Nav grūti ieraudzīt, ka pirmajā gadījumā situācija ir līdzīga Dirihlē nosacījumiem, otrajā - Neimana nosacījumiem, vienīga atšķirība: tie nav atklāti uzdoti. Pieņemsim tagad, ka attālums starp malām Γ_j un Γ_i ir l un tiek apskatīts periodiskums x ass virzienā. Pieņemsim arī, ka robežas atlikušajā daļā ir uzdoti homogēni Dirihlē nosacījumi. Tad eliptiska tipa vienādojumam, uzdotam vispārīgā formā var rakstīt:

$$-\int_{\partial\Omega} v(A\nabla u)nd\sigma + \int_{\Omega} \nabla v^T A\nabla u dx + \int_{\Omega} b^T \nabla u v dx + \int_{\Omega} f(x, y)u v dx = \int_{\Omega} g v dx. \quad (4.8)$$

Integrējot pa kontūru $\partial\Omega$, orientācija mainās, sasniedzot malu Γ_i , līdz ar to:

$$-\int_{\partial\Omega} v(A\nabla u)nd\sigma + \int_{\partial\Omega} v(A\nabla u)nd\sigma + \int_{\Omega} \nabla v^T A\nabla u dx + \int_{\Omega} b^T \nabla u v dx + \int_{\Omega} f(x, y)u v dx = \int_{\Omega} g v dx. \quad (4.9)$$

Var uzreiz saprast, ka atvasinājuma periodiskums vienmēr izpildīsies, vāji formulējot periodisku problēmu. Testa funkciju un atrisinājuma telpām tad paliek uzdot ierobežojumu uz funkciju vērtībām uz attiecīgām malām. Definēsim šādu telpu:

$$H_{0,px}^1(\Omega) = \{u : u \in H^1(\Omega); u = 0, \partial\Omega \setminus \Gamma_j \setminus \Gamma_i, u(x, y) = u(x + l, y), \Gamma_i \cup \Gamma_j\}.$$

Līdz ar to, periodiskai robežproblēmai ar homogēniem Dirihlē nosacījumiem uz $\partial\Omega \setminus \Gamma_j \setminus \Gamma_i$, vājš formulējums izskatīsies šādi:

Atrast funkciju $u \in H_{0,px}^1(\Omega)$, kas apmierina vienādojumu

$$a(u, v) = f(v) \quad \forall v \in H_{0,px}^1, \quad (4.10)$$

kur $a(u, v)$, $f(v)$ nosaka (4.9).

4.2. Vispārināta atrisinājuma eksistence un unitāte

Definīcija 16. Funkcijas, kas apmierina prasības (4.5), (4.6) un (4.10) sauc, attiecīgi, par problēmu (3.3), (3.4) un (3.6) vispārinātiem atrisinājumiem.

Vispārināta atrisinājuma eksistenci un unitāti garantē vairākas literatūrā atrodamās lemmas. Tās, savukārt, prasa lai dažas bilineāras formas $a(u, v)$ un lineāra funkcionāļa $f(v)$ īpašības izpildītos.

Definīcija 17. [1] Hilberta telpu reizinājumā $H \times H$ definētu bilineāru formu $a(u, v) : H \times H \rightarrow \mathbb{R}$ sauc par H -eliptisku, ja

$$\exists \gamma > 0, \forall u \in H \quad \gamma \|u\|_H^2 \leq a(u, u).$$

Definīcija 18. [3] Hilberta telpu reizinājumā $H \times H$ definētu bilineāru formu $a(u, v) : H \times H \rightarrow \mathbb{R}$ sauc par nepārtrauktu, ja

$$\exists \lambda > 0, \forall u, v \in H \quad |a(u, v)| \leq \lambda \|u\|_H \|v\|_H.$$

Definīcija 19. Normētā telpā H definētu lineāru funkcionāli $f(v) : H \rightarrow \mathbb{R}$ sauc par nepārtrauktu, ja

$$\exists M > 0, \forall v \in H \quad |f(v)| \leq M \|v\|_H.$$

Nākamais rezultāts garantē atrisinājuma eksistenci un unitāti:

Laksa-Milgrama lemma. [1]

Ja Hilberta telpu H reizinājumā $H \times H$ definēta bilineāra forma $a(u, v) : H \times H \rightarrow \mathbb{R}$ ir nepārtraukta un H -eliptiska, lineārs funkcionālis $f(v) : H \rightarrow \mathbb{R}$ - nepārtraukts, tad uzdevumam, kurš skan šādi:

Atrast funkciju $u \in H$, kas apmierina vienādojumu $a(u, v) = f(v) \quad \forall v \in H$,

ir viens vienīgs atrisinājums.

Dirihlē, Neimana un periodiskai problēmām atbilstošiem uzdevumiem (4.5), (4.6), (4.10) nav grūti parādīt, ka Laksa-Milgrama lemmas nosacījumi izpildās.

1. **Bilineāras formas $\mathbf{a}(\mathbf{u}, \mathbf{v})$ nepārtrauktība.** Problēmās (4.5), (4.6) un (4.10),

$$a(u, v) = \int_{\Omega} \nabla v^T A \nabla u dx + \int_{\Omega} b^T \nabla u v dx + \int_{\Omega} f(x, y) u v dx.$$

Lai parādītu, ka $|a(u, v)| \leq \lambda \|u\|_{H^1} \|v\|_{H^1}$, izteiksmi $a(u, v)$ pārraksta formā:

$$a(u, v) = \sum_{0 \leq i, j \leq 2} (a_{ij}(x, y) \cdot \partial^i u, \partial^j v)_{L_2},$$

kur funkcijas $a_{ij}(x, y)$ ir:

$$\begin{pmatrix} a_{00} & a_{01} & a_{02} \\ a_{10} & a_{11} & a_{12} \\ a_{20} & a_{21} & a_{22} \end{pmatrix} = \begin{pmatrix} f(x, y) & 0 & 0 \\ d(x, y) & a(x, y) & b(x, y) \\ e(x, y) & b(x, y) & c(x, y) \end{pmatrix}$$

- vienādojuma (3.1) koeficienti - apgalabā Ω ierobežotas funkcijas. Tad var rakstīt (izmantojot Kosī nevienādību (A, 11)):

$$\begin{aligned} |a(u, v)| &= \left| \sum_{0 \leq i, j \leq 2} (a_{ij}(x, y) \cdot \partial^i u, \partial^j v)_{L_2} \right| \leq \lambda \sum_{0 \leq i, j \leq 2} \|\partial^i u\|_{L^2} \|\partial^j v\|_{L^2} = \\ &= \lambda \sqrt{\int_{\Omega} (|u|^2 + |u_x|^2 + |u_y|^2) dx} \sqrt{\int_{\Omega} (|v|^2 + |v_x|^2 + |v_y|^2) dx} = \lambda \|u\|_{H^1} \|v\|_{H^1}, \end{aligned} \quad (4.11)$$

kur $\lambda = \max_{0 \leq i, j \leq 2} \{C : a_{ij}(x, y) \leq C\}$.

2. **Bilineāras formas $\mathbf{a}(\mathbf{u}, \mathbf{v})$ H^1 -eliptiskums.**

Šeit ir jāparāda, ka $a(u, u) \geq \gamma \|u\|_{H^1}^2$. Atkal pārraksta:

$$\begin{aligned} a(u, u) &= \sum_{0 \leq i, j \leq 2} (a_{ij}(x, y) \cdot \partial^i u, \partial^j u)_{L_2} \geq \gamma_1 \int_{\Omega} (|u|^2 + |u_x|^2 + |u_y|^2) dx + \\ &+ \sum_{\substack{0 \leq i, j \leq 2 \\ i \neq j}} (a_{ij}(x, y) \cdot \partial^i u, \partial^j u)_{L_2} \geq \gamma \int_{\Omega} (|u|^2 + |u_x|^2 + |u_y|^2) dx = \gamma \|u\|_{H^1}^2, \end{aligned}$$

kur $\gamma_1 = \min_{0 \leq i \leq 2} \{C : a_{ii}(x, y) \geq C\}$.

3. Lineāra funkcionāļa $f(v)$ nepārtrauktība.

- (a) Problēmās (4.5), (4.10), $f(v) = \int_{\Omega} gv dx$. Šeit jāparāda: $|f(v)| \leq M\|v\|_{H^1}$. To var parādīt, atkal izmantojot Koši nevienādību (A, 11):

$$|f(v)| = |(g, v)_{L^2}| \leq \|f\|_{L^2} \|g\|_{L^2} \leq M\|v\|_{H^1}.$$

- (b) Problēmā (4.6), $f(v) = \int_{\Omega} gv dx + \int_{\partial\Omega} hvd x$. Nepārtrauktību šajā gadījumā, piemēram, grāmatā [1], pierāda izmantojot rezultātu:

Ja $\partial\Omega$ - Nepārtraukta pēc Lipšica apgabala Ω robeža, tad

$$\exists K > 0 : \forall v \in H^1(\bar{\Omega}) \quad \|v\|_{L^2(\partial\Omega)} \leq K\|v\|_{H^1(\Omega)}.$$

Tad, ņemot vērā iepriekšējo novērtējumu (3a) un atkal izmantojot Koši nevienādību (A, 11), var rakstīt:

$$\begin{aligned} |f(v)| &\leq |(g, v)_{L^2(\Omega)}| + |(h, v)_{L^2(\partial\Omega)}| \leq \\ &\leq M\|v\|_{H^1(\Omega)} + \|h\|_{L^2(\partial\Omega)}\|v\|_{L^2(\partial\Omega)} = M\|v\|_{H^1(\Omega)} + K_1\|v\|_{L^2(\partial\Omega)} \leq \\ &\leq M\|v\|_{H^1(\Omega)} + K_1 \cdot K\|v\|_{H^1(\Omega)} \leq \max\{M, K_1 \cdot K\}\|v\|_{H^1(\Omega)}. \end{aligned}$$

4.3. Vispārināta atrisinājuma stabilitāte

Stabilitāti rāda atkal izmantojot bilineāras formas $a(u, v)$ H -eliptiskumu. Vispārinātam atrisinājumam u jābūt nepārtraukti atkarīgam no datiem $f(v)$, tātad ir jābūt spēkā: ([3])

$$\|u\|_{H^1} \leq \|f(v)\|. \quad (4.12)$$

Funkcionāļa $f(v)$ normu izvēlās formā $\sup_{v \in H^1} \frac{|f(v)|}{\|v\|_{H^1}}$. Bilineāras formas $a(u, v)$ H -eliptiskums dod:

$$\gamma\|u\|_{H^1}^2 \leq a(u, u) = f(u),$$

un, savukārt,

$$f(u) \leq \|f\| \cdot \|u\|_{H^1} = \sup_{u \in H^1} \frac{|f(u)|}{\|u\|_{H^1}} \cdot \|u\|_{H^1},$$

no kurienes seko prasības (4.12) izpilde.

5. Problēmu diskretizācija

Šajā nodaļā jāiegūst problēmas

Atrast funkciju $u \in H(\Omega)$, kas apmierina vienādojumu

$$a(u, v) = f(v) \quad \forall v \in H, \quad (5.1)$$

diskrētais analogs, balstoties uz telpas diskretizācijas pieeju [1]. Šajā gadījumā tiek apskatīta telpas H apakštelpa, kurai ir galīga dimensija. Problēmai (5.1) atbilstošā diskrēta problēma izskatās šādi:

Atrast funkciju $u \in H_h(\Omega)$, kas apmierina vienādojumu

$$a(u_h, v_h) = f(v_h) \quad \forall v_h \in H_h. \quad (5.2)$$

Uzdevumam (5.2) arī eksistē viens vienīgs atrisinājums, jo ir spēkā telpu iekļaušanas [1]:

$$H_h \subset H^1, \quad H_h \subset H_0^1 \quad \text{un} \quad H_h \subset H_{0,p}^1.$$

Telpas H_h dimensija jeb bāzes elementu skaits ir $N \sim \frac{1}{h}$ un pie $h \rightarrow 0$, $\dim H_h \rightarrow \infty$.

5.1. Gaļorkina metode telpā \mathcal{H}^1

Definīcija 20. [4] Kopu no N telpas H_h lineāri neatkarīgiem elementiem

$$S := \{\psi_1, \dots, \psi_N\} \quad (5.3)$$

sauc par bāzi telpā H_h ja

$$\forall i \in \{1 \dots N\} \quad \forall \psi \in H \quad \exists \beta_i \in \mathbb{R} : \quad \psi = \sum_{i=1}^N \beta_i \psi_i,$$

jeb ja katru telpas H_h elementu var izteikt kā bāzes elementu lineāro kombināciju. [8]

N -dimensiju telpā H_h izvēlas kādu bāzi S un apskata N vienādojumus, katrā i -tā gadījumā, $i \in \{1 \dots N\}$, par testa funkciju $v_h \in H_h$ izvēloties bāzes funkciju $\psi_i \in H_h$:

$$a(u_h, \psi_i) = f(\psi_i) \quad \forall i \in \{1 \dots N\}.$$

Elementu u_h meklē bāzes funkciju lineāras kombinācijas formā $\sum_{j=1}^N \xi_j \psi_j$:

$$a\left(\sum_{j=1}^N \xi_j \psi_j, \psi_i\right) = f(\psi_i) \quad \forall i \in \{1 \dots N\}.$$

Objektu $a(u_h, v_h)$ un $f(v_h)$ bilinearitātes un linearitātes īpašību dēļ:

$$\sum_{j=1}^N a(\psi_j, \psi_i) \cdot \xi_j = f(\psi_i) \quad \forall i \in \{1 \dots N\}.$$

Šī N vienādojumu sistēma attiecībā pret nezināmiem ξ_j , $j \in \{1 \dots N\}$, pārrakstāma arī matricas formā:

$$A\xi = f, \tag{5.4}$$

kur $a_{ij} = a(\psi_j, \psi_i)$, $f_i = f(\psi_i)$.

Šeit ir vērts atgādināt, ka bilineārā forma $a(u, v)$ ir H^1 -eliptiska, un šī īpašība tai piemīt arī telpā H_h , jo ir spēkā šo telpu iekļaušana. No bilineāras formas $a(u_h, v_h)$ H_h -eliptiskuma seko ka matrica A ir pozitīvi definita, un līdz ar to, nesingulāra: $\xi^T A \xi \geq 0 \quad \forall \xi \in \mathbb{R}^N$, kur $A = (a(\phi_j, \phi_i))$, $1 \leq i, j \leq N$, jo

$$\forall \xi_i, \xi_j \in \mathbb{R} \quad \xi^T A \xi = \sum_{i,j=1}^N a(\phi_j, \phi_i) \xi_j \xi_i,$$

un bilinearitātes, H^1 -eliptiskuma dēļ:

$$\sum_{i,j=1}^N a(\phi_j, \phi_i) \xi_j \xi_i = a\left(\sum_{j=1}^N \xi_j \phi_j, \sum_{i=1}^N \xi_i \phi_i\right) = a(u_h, u_h) \geq \gamma \|u_h\|_{H^1}^2 \geq 0.$$

Vektora $\xi = A^{-1}f$ noteikšana ir pēdējais solis tuvināta atrisinājuma $u_h = \sum_{j=1}^N \xi_j \psi_j$ izteiksmes noteikšanā. Bāzes funkciju ψ_j , $j \in \{1 \dots N\}$ izvēle Gaļorkina pieejā netiek precizēta; taisot lineāru aproksimāciju, tas parasti tiek izdarīts definējot galīgo elementu telpu.

5.2. Galīgo elementu telpa

Lineāras algebras uzdevuma (5.4) atvieglošanai prasa lai matrica A būtu vāji aizpildīta. Tas savukārt ir atkarīgs no bāzes funkciju (5.3) izvēles. Tieši tāpēc, bāzes funkcijas galīgo elementu metodē ir definētas lokāli, uz apgabala Ω atsevišķiem apakšapgabaliem jeb *elementiem*. Apgabalu Ω sadala pilnībā tādos elementos. Elementu skaits tad ir ciešā sakarā ar bāzes funkciju skaitu.

Prasa arī lai bāzes funkcijas būtu gabaliem lineāras vai gabaliem polinomiālas funkcijas, kas garantē lielumu $a_{ij} = a(\psi_j, \psi_i)$, $f_i = f(\xi_i)$ eksistenci, jo tāda tipa funkcijām eksistē to vispārinātie atvasinājumi, nepieciešami šo lielumu aprēķināšanai.

Galvenais metodes aspekts tomēr ir apgabala sadales apakšgabalos jeb *triangulācijas* realizēšana apgabalam $\bar{\Omega}$.

Definīcija 21. [1] Ierobežota apgabala $\Omega \subset \mathbb{R}^n$ sadali \mathcal{T}_h uz elementiem $T \subset \mathbb{R}^n$, $T \in \mathcal{T}_h$, sauc par apgabala Ω triangulāciju, ja ir spēkā

1. $\bar{\Omega} = \bigcup_{T \in \mathcal{T}_h} T$.
2. $\forall T \in \mathcal{T}_h$: T - slēgta, $\text{int}(T) \neq \emptyset$, ∂T - Nepārtrauktā pēc Lipsīca robeža.
3. $\forall T_i, T_j \in \mathcal{T}_h$, $i \neq j$: $\text{int}(T_i) \cap \text{int}(T_j) = \emptyset$.

Turklāt gadījumā kad apgabalu sadala daudzskaldņos, pastāv vēl ceturrtā prasība:

4. $\forall T_i \in \mathcal{T}_h$: katra T_i skaldne ir vai nu cita elementa T_j skaldne, vai arī tā pieder apgabala Ω robežai $\partial\Omega$.

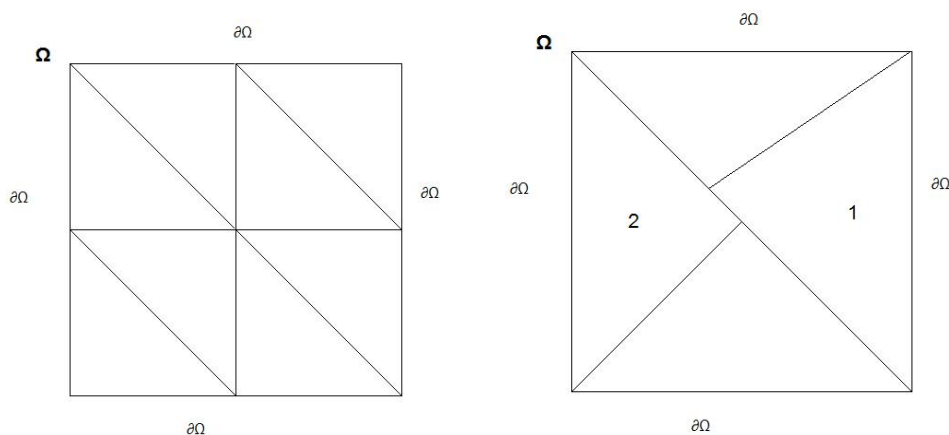
Visbiežāk tiek izmantoti polinomiālie elementi, tāpēc definējot *galīgo elementu telpu*, tiek izmantota tieši šī pieeja.

Definīcija 22. Telpu

$$S_h^{(m)} := \{u \in C^0(\bar{\Omega}) : u|_{T_i} \in \Pi_m \forall T_i \in \mathcal{T}_h\}$$

sauc par m -tās kārtas polinomiālo galīgo elementu telpu attiecībā uz triangulāciju \mathcal{T}_h .

1. nodaļā apskatītas problēmas ir divdimensiju problēmas, un vispopulārākais divdimensiju apgabala sadales veids ir sadale trijstūros. Tādā gadījumā jebkuriem diviem triangulācijai \mathcal{T}_h piederošiem trijstūriem jābūt vienai kopīgai malai. Prasības 4. izpilde un neizpilde parādīta 3. attēlā:



3. att. Trijstūriem 1 un 2 nav spēkā īpašība 4.

Par galīgu elementu parasti sauc jebkuru kopas \mathcal{T}_h elementu, bet pastāv arī vispārīgāka galīga elementa definīcija, kas aptver uzreiz vairākus metodes aspektus.

Definīcija 23. [1] Par *galīgu elementu* telpā \mathbb{R}^n sauc trijnieku (T, P, Σ) , kur:

1. T - slēgta, $\text{int}(T) \neq \emptyset$, ∂T - Nepārtrauktā pēc Lipšica robeža.
2. P - definētu uz T funkciju $T \rightarrow \mathbb{R}$ telpa.
3. Σ - lineāri neatkarīgu lineāru funkcionāļu φ_i galīgā kopa ar īpašību:
 $\forall \alpha_i \in \mathbb{R} \exists p \in P : \varphi_i(p) = \alpha_i$.

Tātad, galīgo elementu raksturo bāzes funkcijas $p \in P$, definētas šajā elementā un brīvības pakāpes φ_i . Kopas Σ īpašība ļauj viennozīmīgi noteikt lineāras vai polinomiālas funkcijas, definētas uz elementa T . Gadījumā kad brīvības pakāpes ir veidā $p \rightarrow p(a_i)$, kur a_i - mezgla punkti, bāzes funkciju (uzdotās pakāpes polinomu) atrašana ir interpolācijas uzdevums.

Aproksimācijas kvalitāte

Skaidrs, ka aproksimācijas kvalitāte ir atkarīga no gabaliem polinomiālo bāzes funkciju kārtas izvēles. Bet pašam par sevi Gaļorkina atrisinājumam, bāzes funkcijas neprecizējot, literatūrā ir ļoti bieži sastopama lemma, kas garantē, ka ar precizitāti līdz konstantei, diskrēts atrisinājums u_h ir tikpat tuvs nepārtrauktam atrisinājumam, cik tuva jebkura cita funkcija no H^1 :

Ceas lemma [3], [1] *Pie nosacījumiem, ka bilineāra forma a ir nepārtraukta un H^1 -eliptiska, problēmu (5.1) un (5.2) atrisinājumi u un u_h pie kādas $C > 0$, apmierina:*

$$\|u - u_h\|_{H^1} \leq C \inf_{v_h \in H_h} \|u - v_h\|_{H^1}.$$

Tas arī nozīmē, ka aproksimācija u_h ir vislabākā. Pierādījums, protams, balstās uz bilineāras formas H^1 -eliptiskumu.[1, 109 lpp.]

Ceas lemmas apgalvojums ir atslēga visiem turpmākiem konverģences rezultātiem: pie nosacījuma, ka nepārtraukts atrisinājums $u \in H^{k+1}$, var pierādīt, ka

$$\|u - u_h\|_{H^1(\Omega)} = O(h^k)$$

[1, 136. lpp]

5.3. Lineārie Lagranža elementi

Definīcija 24. Lineārie Lagranža elementi ir telpas

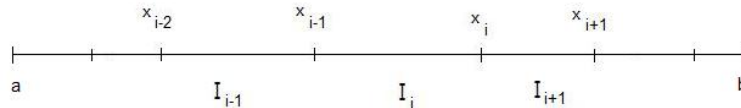
$$S_h^{(1)} := \{u \in C^0(\bar{\Omega}) : u|_{T_i} \in \Pi_1 \forall T_i \in \mathcal{T}_h\}$$

elementi.

Šajā apakšnodaļā tiks atrastas analītiskas izteiksmes bāzes funkcijām vienā un divās dimensijās standarta tipa apgabaliem. Iegūtos rezultātus var likt pamatā nestandarta apgabala Ω analīzei Psi-Zēta sistēmai (1.7).

Viendimensijas gadījums

Viendimensijas gadījumā bāzes funkcijas konstruē uz intervāla (a, b) . Intervālu sadala nogriežņos $I_i = (x_{i-1}, x_i)$, $a = x_0 < x_1 < \dots < x_n = b$ (4. attēls:)



4. att. Nevienmērīgais režģis

Tādā gadījumā, katrs i -tais elements I_i ir asociēts ar mezgla punktiem x_{i-1} un x_i . Elementa brīvības pakāpes – vērtības $\phi_i(x_j)$ – ļauj viennozīmīgi noteikt lineāru funkciju, definētu uz elementa I_i ; parasti tiek izvēlēts $\phi_i(x_j) = \delta_{ij}$. Katras bāzes funkcijas ϕ_i nesējs (A, 29) tad ir elementu I_i un I_{i+1} apvienojums.

Taisnes vienādojums katrā no elementiem ir $ax + b$ un diviem elementiem I_i un I_{i+1} ar kopīgo režģa punktu x_i , ievērojot prasību $\phi_i(x_j) = \delta_{ij}$, var sastādīt divas lineāras vienādojumu sistēmas koeficientu a un b katram elementam noteikšanai:

$$\begin{cases} ax_i + b = 1 \\ ax_{i-1} + b = 0, \end{cases} \quad (5.5)$$

un

$$\begin{cases} ax_i + b = 1 \\ ax_{i+1} + b = 0. \end{cases} \quad (5.6)$$

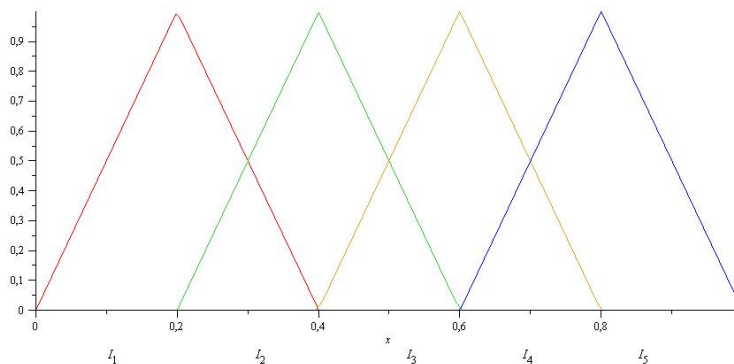
Sistēmas (5.5) atrisinājums ir $\begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} \frac{1}{x_i - x_{i-1}} \\ -\frac{x_{i-1}}{x_i - x_{i-1}} \end{pmatrix}$,

Sistēmas (5.6) atrisinājums ir $\begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} -\frac{1}{x_{i+1}-x_i} \\ \frac{x_{i+1}}{x_{i+1}-x_i} \end{pmatrix}$.

Bāzes funkcijas $\phi_i(x)$ analītiska izteiksme tad ir formā:

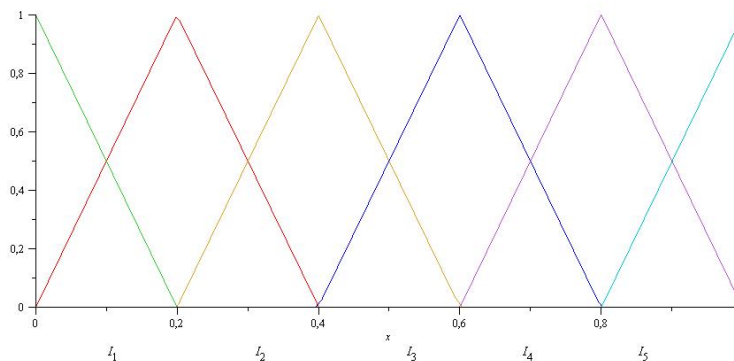
$$\phi_i(x) = \begin{cases} \frac{x-x_{i-1}}{x_i-x_{i-1}}, & x \in I_i \\ \frac{x_{i+1}-x}{x_{i+1}-x_i}, & x \in I_{i+1} \\ 0, & \text{citur,} \end{cases} \quad (5.7)$$

kur $i = 0..n$. Izvēloties $i \in \{1, ..n - 1\}$, iegūst bāzes funkcijas Dirihlē problēmas (3.3) viendimensijas variantam. Tādā gadījumā, uz elementa I_1 ir definēta tikai augošā funkcija, uz elementa I_n - tikai dilstošā, kas garantē homogēna Dirihlē nosacījuma izpildi. (5. attēls:)



5. att. Bāzes funkcijas $\phi_1 - \phi_4$, $n=5$

Izvēloties $i \in \{0, ..n\}$, iegūst bāzes funkcijas Neimana problēmas (3.4) viendimensijas variantam. Ir pievienotas divas nepilnas bāzes funkcijas, kas nepieņem nulles vērtības intervāla galapunktos (6. attēls:)



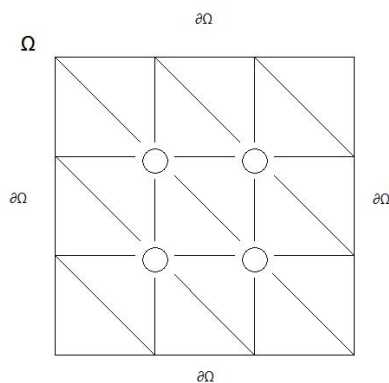
6. att. Bāzes funkcijas $\phi_0 - \phi_5$, $n=5$

Periodisko robežnosacījumu gadījumā ir jāizvēlas $i \in \{0, \dots, n-1\}$, jo vienai bāzes funkcijai obligāti jābūt periodiskai. Viendimensionāls gadījums palīdz saprast, kādā veidā ir jāuzdod bāze divās dimensijās.

Divdimensiju gadījums

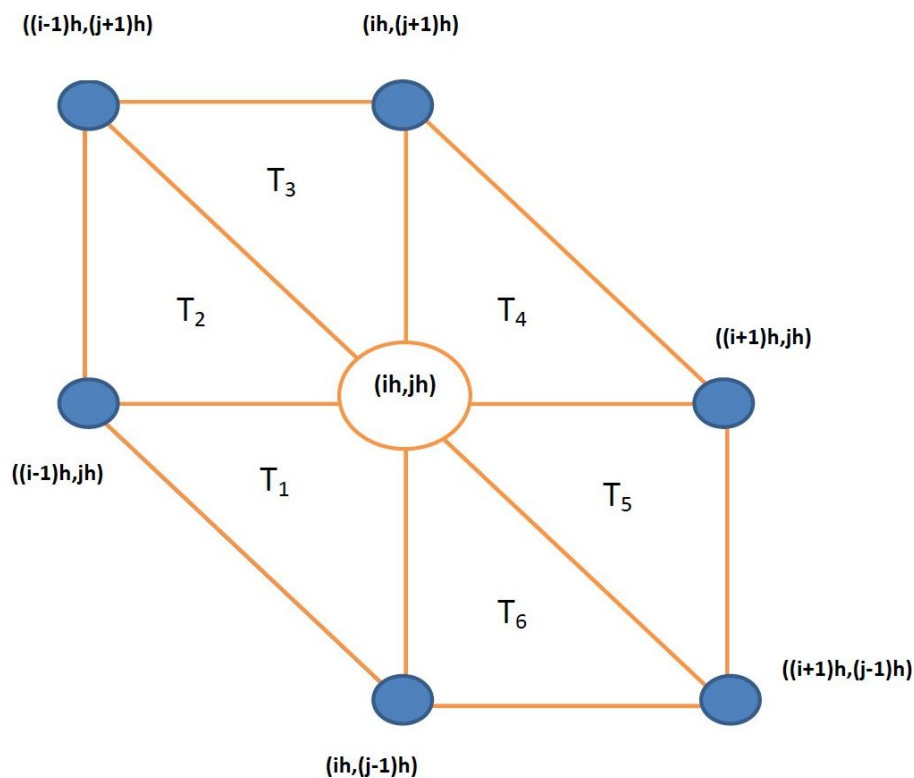
Divās dimensijās taisnstūra apgabalu Ω vienmērīgi sadala trijstūros:

Definīcija 25. Par taisnstūrveida apgabala *vienmērīgu triangulāciju* sauksim apgabala sadali ar līnijām $x = ih$, $y = jh$, $y = ih - x$, $i, j = 1..N + 1$ (7. attēls). Par *virsoņnēm* tad sauc iekšējos režģa punktus un katru gabaliem lineāru bāzes funkciju asociē ar kādu virsotni.



7. att. Vienmērīga triangulācija ar četrām virsoņnēm

Funkcijai $\phi_{ij}(x, y)$ ir prasība: tā pieņem vērtību 1 uz virsotnes (ih, jh) un lineāri dilst katrā no sešiem blakus esošiem trijstūriem. Katras bāzes funkcijas $\phi_{ij}(x, y)$ nesējs ir šo sešu trijstūru apvienojums (8. attēls).



8. att. Asociētas ar virsotni (ih, jh) , bāzes funkcijas nesējs

8. attēlā shematiski parādīts virsotņu izvietojums. Plaknes vienādojums katrā no trīsstūriem T_k , $k = 1..6$ ir formā $a_k + b_k x + c_k y$, $(x, y) \in T_k$.

Zinot šo sešu virsotņu koordinātes, var atrast koeficientu a_k , b_k un c_k , $k = 1..6$ vērtības, atrisinot sešas lineāras trīs vienādojumu sistēmas:

$$\begin{cases} a_k + b_k x_1 + c_k y_1 = 1 \\ a_k + b_k x_2 + c_k y_2 = 0 \\ a_k + b_k x_3 + c_k y_3 = 0, \end{cases} \quad (5.8)$$

kur $(x_1, y_1) = (ih, jh)$ - virsotne, kurā $\phi_{ij} = 1$, un (x_2, y_2) , (x_3, y_3) - pārējas divas k -tā trijstūra virsotnes (8. attēls).

Sistēmas (5.8) matrica ir $\begin{pmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{pmatrix}$, kuras determinanta puse ir patvaļīga trijstūra ar virsotnēm (x_i, y_i) , $i = 1..3$, laukums. Tādā gadījumā, formula 3×3 matricas inversas

matricas aprēķināšanai (A, 12) dod:

$$\begin{pmatrix} a_k \\ b_k \\ c_k \end{pmatrix} = \frac{1}{\det(A)} \begin{pmatrix} x_2y_3 - x_3y_2 \\ y_2 - y_3 \\ x_3 - x_2 \end{pmatrix} = \frac{1}{2S} \begin{pmatrix} x_2y_3 - x_3y_2 \\ y_2 - y_3 \\ x_3 - x_2 \end{pmatrix} = \frac{1}{h^2} \begin{pmatrix} x_2y_3 - x_3y_2 \\ y_2 - y_3 \\ x_3 - x_2 \end{pmatrix}, \quad (5.9)$$

kur S - trīsstūra laukums, vienmērīga režģa gadījumā, $S = \frac{1}{2}h^2$.

Bāzes funkcijas ϕ_{ij} tad ir formā:

$$\left\{ \begin{array}{l} a_1 + b_1x + c_1y, \quad (x, y) \in T_1 \\ a_2 + b_2x + c_2y, \quad (x, y) \in T_2 \\ a_3 + b_3x + c_3y, \quad (x, y) \in T_3 \\ a_4 + b_4x + c_4y, \quad (x, y) \in T_4 \\ a_5 + b_5x + c_5y, \quad (x, y) \in T_5 \\ a_6 + b_6x + c_6y, \quad (x, y) \in T_6 \\ 0, \quad \text{citur,} \end{array} \right. \quad (5.10)$$

kur $T_1 - T_6$ - trīsstūri, asociēti ar virsotni (ih, jh) . Trīsstūri T_i , $i = 1..6$ uzdod ierobežojumi:

$$\begin{array}{ll} T_1 : \left\{ \begin{array}{l} (i-1)h \leq x \leq ih \\ (i+j-1)h - x \leq y \leq jh, \end{array} \right. & T_2 : \left\{ \begin{array}{l} (i-1)h \leq x \leq ih \\ jh \leq y \leq (i+j)h - x, \end{array} \right. \\ T_3 : \left\{ \begin{array}{l} (i-1)h \leq x \leq ih \\ (i+j)h - x \leq y \leq (j+1)h, \end{array} \right. & T_4 : \left\{ \begin{array}{l} ih \leq x \leq (i+1)h \\ jh \leq y \leq (i+j+1)h - x, \end{array} \right. \\ T_5 : \left\{ \begin{array}{l} ih \leq x \leq (i+1)h \\ (i+j)h - x \leq y \leq jh, \end{array} \right. & T_6 : \left\{ \begin{array}{l} ih \leq x \leq (i+1)h \\ (j-1)h \leq y \leq (i+j)h - x. \end{array} \right. \end{array}$$

Zinot šos ierobežojumus, bāzes funkciju ϕ_{ij} izteiksmes var iegūt ar datora palīdzību. Koeficientu aprēķināšanai var realizēt šādu algoritmu:

$$1. \text{ Definē nobīdes matricas: } nob1 := h \cdot \begin{pmatrix} 0 & -1 \\ -1 & 0 \\ -1 & 1 \\ 0 & 1 \\ 1 & 0 \\ 1 & -1 \end{pmatrix}, \quad nob2 := h \cdot \begin{pmatrix} -1 & 0 \\ -1 & 1 \\ 0 & 1 \\ 1 & 0 \\ 1 & -1 \\ 0 & -1 \end{pmatrix}.$$

2. katram $i=1..N$

katram $j=1..N$

katram $k=1..6$

uzdod: $x_1 := ih, y_1 := jh,$

$x_2 := x_1 + nob1(k, 1), y_2 := y_1 + nob1(k, 2),$

$x_3 := x_1 + nob2(k, 1), y_3 := y_1 + nob2(k, 2).$

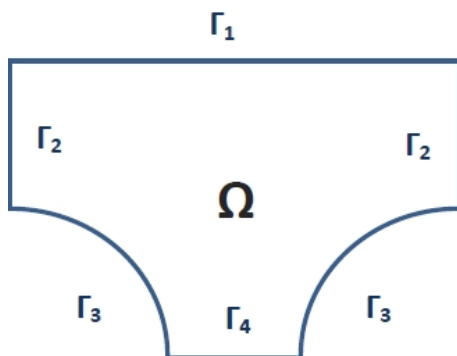
Koeficientus a_k, b_k, c_k atrod, izmantojot sakarību (5.9) un ievieto bāzes funkcijas ϕ_{ij} izteiksmē (5.10).

Šis algoritms ļauj konstruēt bāzes funkcijas Dirihlē problēmām ar homogēniem robežnosacījumiem, bet pieņemot: $i, j = 0..N + 1$, virsotnes var apskatīt arī uz apgabala robežas gadījumā, kad jārisina Neimana problēma. Gadījums $i, j = 0..N$ tad atbilst periodiskuma nosacījumiem.

Piezīme 4. *Zināšanas par definētas divdimensionālā taisnstūrveida apgabalā vienmērīgas triangulācijas konstruēšanas tehnikām ļaus uzkonstruēt arī nevienmērīgo Psi-Zēta sistēmas (1.7) apgabalam Ω 8. nodaļā.*

6. Korekti formulēta problēma

Tagad rokās ir visi instrumenti korektai problēmas formulēšanai. Tas nozīmē: robežnosacījumus, kas tika apskatīti 1. nodaļā tagad jāuzdod formāli. Jāatgādina, ka visa bezgalīga apgabala apakšapgabals Ω izskatās šādi (9. attēls):



9. att. Apgabals Ω

Vispirms jāuzdod kopa Ω Dekarta koordinātēs:

$$\Omega := \left\{ (x, y) \mid 0 < x < L_1 \wedge y < L_2 \wedge y > \tilde{f}(x) \right\},$$

Nav nozīmēs kurā vietā ir izvēlēta nulle (koordinātu sākumpunkts), tāpēc par punktu $\begin{pmatrix} 0 \\ 0 \end{pmatrix}$ izvēlas kreisās riņķa līnijas centru. L_1 ir apvilka ap figūru Ω taisnstūra garums, L_2 - platums.

Uzdosim arī apgabala robežu (6.1):

$$\begin{aligned} \Gamma_1 &:= [0, L_1] \times \{L_2\}, \\ \Gamma_2 &:= \{0\} \times [R, L_2 - R] \cup \{L_1\} \times [R, L_2 - R], \\ \Gamma_3 &:= [0, R] \times f_1([0, R]) \cup [L_1 - R, L_1] \times f_2([L_1 - R, L_1]), \\ \Gamma_4 &:= [R, L_1 - R] \times \{0\}, \\ \partial\Omega &= \Gamma_1 \cup \Gamma_2 \cup \Gamma_3 \cup \Gamma_4. \end{aligned} \tag{6.1}$$

Ar R tika apzīmēts riņķa līnijas rādiuss, kuru var arī izmantot kā raksturīgo izmēru 1. nodaļā. Definējot kopu Ω tika izmantota gabaliem nepārtraukta funkcija, uzdota formā

$$\tilde{f}(x) := \begin{cases} f_1(x), & 0 < x \leq R, \\ 0, & R < x < L_1 - R, \\ f_2(x), & L_1 - R \leq x < L_1, \end{cases}$$

$$f_1(x) := \sqrt{R^2 - x^2}, \quad f_2(x) := \sqrt{R^2 - (x - L_1)^2},$$

Piezīme 5. Var izvēlēties arī citas funkcijas f_1 un f_2 vietā. Piemēram, izvēloties $f_i(x) \equiv 1$, $i = 1, 2$, iegūst četrkantīgus cilindrus.

Tagad vajag formalizēt to, kas vizuāli tika konstatēts 1. nodaļā, 1.1. apakšnodaļā: uzdot robežnosacījumus koordinātēs.

1. Simetrijas nosacījumi. Ātruma komponentēm tas nozīmē $V_y = 0$, $\frac{\partial V_x}{\partial x} = 0$, no kurienes $\zeta = 0$, $\Gamma_1 \cup \Gamma_4$.
2. Periodiskie robežnosacījumi. Pieņemot $\hat{n} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$, uzdod periodiskuma nosacījumus (3.5) uz apgabala robežas segmenta Γ_2 .
3. Nosacījumi uz cietām virsmām $V_x = V_y = 0$, Γ_3 .
4. Nosacījumi uz augšēja un apakšēja robežas segmentiem. Šeit ļoti bieži tiek izmantota caurteces normēšana.

Definīcija 26. Par caurteci sauc lielumu

$$Q := \int_0^{L_2} V_x dy.$$

Par normu izvēlēsimies lielumu 1. Tātad

$$Q := \int_0^{L_2} V_x dy = \int_0^{L_2} \frac{\partial \psi}{\partial y} dy = \psi|_{L_2} - \psi|_0 = 1. \quad (6.2)$$

Līdz ar to, uzdod: $\psi = 1$, Γ_1 , $\psi = 0$, Γ_4 .

Piezīme 6. Izvēloties šādu normu, iegūst šķidrums plūsmu no kreisās uz labo pusi. Pretējam virzienam būtu jāizvēlas $Q = -1$.

Psi-Dzeta sistēma, kopa ar uzdotiem robežnosacījumiem izskatās šādi:

$$\left\{ \begin{array}{l} \frac{1}{Re} \Delta \zeta + J(\psi, \zeta) + Sf = 0, \Omega, \\ -\Delta \psi = \zeta, \Omega, \\ \psi = 1, \Gamma_1, \psi = 0, \Gamma_4, \\ \zeta = 0, \Gamma_1 \cup \Gamma_4, \\ \psi(x, y) = \psi(x + L_1, y), \Gamma_2, \\ \zeta(x, y) = \zeta(x + L_1, y), \Gamma_2, \\ \nabla \psi(x, y) \cdot \hat{n} = \nabla \psi(x + L_1, y) \cdot \hat{n}, \Gamma_2, \\ \nabla \zeta(x, y) \cdot \hat{n} = \nabla \zeta(x + L_1, y) \cdot \hat{n}, \Gamma_2, \\ \psi = 0, \Gamma_3. \end{array} \right. \quad (6.3)$$

Robežnosacījumi uz segmenta Γ_3 funkcijai ζ literatūrā tiek uzdoti ļoti reti. Šim nolūkam parasti aplūko diferenciālvienādojumu uz apgabala robežas [5]. Par to tiks sīkāk pastāstīts nākamajā apakšnodaļā.

6.1. Robežnosacījumi uz cietām virsmām funkcijai ζ

Izmantojot Teilora izvīzījumus, literatūrā ļoti bieži konstruē iterāciju procesus, kuri diskrētā formā konverģē uz diskrētām funkcijas ζ vērtībām uz cietām virsmām [9]. Šajā apakšnodaļā tiek uzkonstruēts šāds process, kuru var pielietot jebkuram divdimensionālam nevienmērīgām režģim.

Ievedīsim šādus apzīmējumus:

$$\psi_0 := \psi|_{\Gamma_3}; \quad \zeta_0 := \zeta|_{\Gamma_3},$$

$$\psi_1(x, y) := \psi_0(x \pm h, y); \quad \psi_2(x, y) := \psi_0(x, y \pm h_1).$$

un tagad otrais vienādojums Psi-Zēta sistēmā (6.3) tiek definēts uz lokiem Γ_3 :

$$-\Delta \psi_0 = \zeta_0. \quad (6.4)$$

Katram režģa punktam šādā viedā tiek atrasti arī četri kaimiņpunkti.

Tagad apskata Teilora izvīzījumus abos virzienos:

$$\psi_1 = \psi_0 \pm h \frac{\partial \psi_0}{\partial x} + \frac{h^2}{2} \frac{\partial^2 \psi_0}{\partial x^2} \pm \frac{h^3}{6} \frac{\partial^3 \psi_0}{\partial x^3} + O(h^4), \quad (6.5)$$

$$\psi_2 = \psi_0 \pm h_1 \frac{\partial \psi_0}{\partial y} + \frac{h_1^2}{2} \frac{\partial^2 \psi_0}{\partial y^2} \pm \frac{h_1^3}{6} \frac{\partial^3 \psi_0}{\partial y^3} + O(h_1^4). \quad (6.6)$$

Iterāciju procesu izveido, saskaitot $h_1^2 \psi_1$ ar $h^2 \psi_2$ un atmetot locekļus kārtā, augstākā par otru:

$$\begin{aligned} h_1^2 \psi_1 + h^2 \psi_2 - \cancel{h_1^2 \psi_0} - \cancel{h^2 \psi_0} \mp \cancel{h h_1^2 \frac{\partial \psi_0}{\partial x}} \mp \cancel{h^2 h_1 \frac{\partial \psi_0}{\partial y}} &= \\ = \frac{h_1^2 h^2}{2} \frac{\partial^2 \psi_0}{\partial x^2} + \frac{h_1^2 h^2}{2} \frac{\partial^2 \psi_0}{\partial y^2} &= -\frac{h_1^2 h^2}{2} \zeta_0, \end{aligned}$$

jo ir spēkā $\psi|_{\Gamma_3} \equiv 0$, $\frac{\partial \psi_0}{\partial x} = 0$, $\frac{\partial \psi_0}{\partial y} = 0$ un vienādojums (6.4).

Līdz ar to, vajadzīga robežnosacījuma iegūšanai izmanto sakarību

$$\zeta_0 = -2\left(\frac{\psi_1}{h^2} + \frac{\psi_2}{h_1^2}\right), \quad (6.7)$$

kuras pamatā var izveidot iterāciju procesu formā (6.8), vajadzības gadījumā pielietojot apakšējo relaksāciju ar koeficientu β :

$$\zeta_0^{(k+1)} = -2\beta\left(\frac{\psi_1}{h^2} + \frac{\psi_2}{h_1^2}\right) + (1 - \beta)\zeta_0^{(k)}, \quad k = 0, 1, \dots, \beta \in [0, 1]. \quad (6.8)$$

6.2. Psi-Zēta sistēmas vienādojumu atrisināmība

2. nodaļā tika nolemts izveidot iterāciju procesu sistēmas (6.3) risināšanai. Tāpēc tā ir jāsadala divās problēmās:

$$\left\{ \begin{array}{l} -\Delta \psi = \zeta, \quad \Omega, \\ \psi = 0, \quad \Gamma_3 \cup \Gamma_4 \\ \psi = 1, \quad \Gamma_1 \\ \psi(x, y) = \psi(x + L_1, y), \quad \Gamma_2 \\ \nabla \psi(x, y) \cdot \hat{n} = \nabla \psi(x + L_1, y) \cdot \hat{n}, \quad \Gamma_2. \end{array} \right. \quad (6.9)$$

$$\left\{ \begin{array}{l} \frac{1}{Re} \Delta \zeta + J(\psi, \zeta) + Sf = 0, \quad \Omega, \\ \zeta = 0, \quad \Gamma_1 \cup \Gamma_4, \\ \zeta(x, y) = \zeta(x + L_1, y), \quad \Gamma_2, \\ \nabla \zeta(x, y) \cdot \hat{n} = \nabla \zeta(x + L_1, y) \cdot \hat{n}, \quad \Gamma_2, \\ \zeta = \zeta_0^{(k)}, \quad \Gamma_3. \end{array} \right. \quad (6.10)$$

Problēmas (6.9) atrisinājums ir funkcija ψ un tā tika sastādīta fiksējot funkciju ζ . Otrādi, problēmas (6.10) atrisinājums ir funkcija ζ un tā tika sastādīta fiksējot funkciju ψ . $\zeta_0^{(k)}$ ir funkcija, kuru iegūst pielietojot iterācijas (6.8).

Tagad ir laiks izmantot teoriju, izklāstītu 4. nodaļā.

Sākuma uzrakstīsim vājo nostādni problēmai (6.9). Var ieraudzīt, ka tā ir periodiska robežproblēma (3.6) ar nehomogenitāti Dirihlē robežnosacījumos. Kā jau bija teikts 1. piezīmē, tāda tipa problēmas ir viegli reducējamas uz (3.6). Šim nolūkam ievēsim funkciju ψ_1 ar īpašību $\psi_1 \in \mathcal{H}_{E_1}^1$, kur

$$\mathcal{H}_{E_1}^1 := \{\psi \in \mathcal{H}^1(\Omega) \mid \psi = 1, \Gamma_1; \psi = 0, \Gamma_3 \cup \Gamma_4; \psi(x, y) = \psi(x + L_1, y), \Gamma_2\}.$$

Atrisinājums tagad ir formā $\psi + \psi_1 \in \mathcal{H}_{E_1}^1$ un palīgproblēmas vājā nostādne izskatīsies šādi:

$$\int_{\Omega} \nabla \psi \cdot \nabla v dx = \int_{\Omega} v \zeta - \nabla \psi_1 \nabla v dx \quad \forall v \in \mathcal{H}_{E_0}^1, \quad \psi \in \mathcal{H}_{E_0}^1, \quad (6.11)$$

kur

$$\mathcal{H}_{E_0}^1 := \{v \in \mathcal{H}^1(\Omega) \mid v = 0, \Gamma_3 \cup \Gamma_4 \cup \Gamma_1; v(x, y) = v(x + L_1, y), \Gamma_2\}.$$

Analoģiski, problēmai (6.10): tā arī ir periodiska problēma formā (3.6), bet nehomogēns Dirihlē nosacījums šoreiz ir uzdots uz Γ_3 .

Apzīmējot $J(\psi, \zeta) =: \nabla^\perp \psi \cdot \nabla \zeta$, un ievēdot lielumu $\zeta_1 \in \mathcal{H}_{E_2}^1$, $\zeta + \zeta_1 \in \mathcal{H}_{E_2}^1$,

iegūst:

$$-\int_{\Omega} \nabla \zeta \cdot \nabla v dx + \int_{\Omega} \nabla^\perp \psi \cdot \nabla \zeta v dx = \int_{\Omega} (\tilde{A} \nabla \psi) \cdot \nabla v + \nabla \zeta_1 \nabla v dx \quad \forall v \in \mathcal{H}_{E_0}^1, \quad \zeta \in \mathcal{H}_{E_0}^1, \quad (6.12)$$

$$\mathcal{H}_{E_2}^1 := \{\zeta \in \mathcal{H}^1(\Omega) \mid \zeta = \zeta_0^{(k)}, \Gamma_3; \zeta = 0, \Gamma_1 \cup \Gamma_4; \zeta(x, y) = \zeta(x + L_1, y), \Gamma_2\},$$

kur $\zeta_0^{(k)}$ — atkal ir funkcija, kuru iegūst pielietojot iterācijas (6.8).

Palīgproblēmu (6.11) un (6.12) korektība ir parādīta 4. nodaļā.

Lai Laksa-Milgrama lemmas (4.2.) nosacījumi pilnībā izpildītos, vēl būtu jāparāda lineāro funkcionāļu nepārtrauktība problēmu (6.11), (6.12) labajās pusēs, bet tas izriet no fakta, ka labās puses apraksta eliptiskie operatori un nepārtrauktība seko uzreiz no novērtējuma (4.11), iegūta eliptiskam operatoram vispārīgā formā 4. nodaļā.

Palīgproblēmu (6.11) un (6.12) formulējumi ir saskaņoti ar 4. nodaļas teoriju, bet diskretizācijai, procesu formalizējot, izmanto sākotnējās problēmas:

$$\int_{\Omega} \nabla \psi \cdot \nabla v dx = \int_{\Omega} v \zeta dx \quad \forall v \in \mathcal{H}_{E_0}^1, \quad \psi \in \mathcal{H}_{E_1}^1, \quad (6.13)$$

$$-\int_{\Omega} \nabla \zeta \cdot \nabla v dx + \int_{\Omega} \nabla^\perp \psi \cdot \nabla \zeta v dx = \int_{\Omega} (\tilde{A} \nabla \psi) \cdot \nabla v dx \quad \forall v \in \mathcal{H}_{E_0}^1, \quad \zeta \in \mathcal{H}_{E_2}^1. \quad (6.14)$$

Tagad var izveidot iterāciju procesu, diskretizējot tieši telpas $\mathcal{H}_{E_1}^1$, $\mathcal{H}_{E_2}^1$. Gadījumā, ja process konverģēs, varēs konstatēt, ka problēma (6.3) ir atrisināta.

7. Iterāciju process

Izmantojot teoriju no 5. nodaļas, problēmu (6.11), (6.12) atrisinājumus meklē telpās ar galīgo dimensiju:

$$\psi \in S_1^h := \text{span}\{\phi_1, \dots, \phi_{\tilde{N}}, \phi_{\tilde{N}+1}, \dots, \phi_N, \phi_{N+1}, \dots, \phi_{N+N_{\partial_1}}\},$$

$$\zeta \in S_2^h := \text{span}\{\phi_1, \dots, \phi_{\tilde{N}}, \phi_{\tilde{N}+1}, \dots, \phi_N, \phi_{N+1}, \dots, \phi_{N+N_{\partial_2}}\},$$

$$v \in S_0^h := \text{span}\{\phi_1, \dots, \phi_{\tilde{N}}, \phi_{\tilde{N}+1}, \dots, \phi_N\}.$$

kur $\phi_1, \dots, \phi_{\tilde{N}}$ — bāzes funkcijas, kas atbilst iekšējiem režģa punktiem, $\phi_{\tilde{N}+1}, \dots, \phi_N$ — periodiskās bāzes funkcijas, $\phi_{N+1}, \dots, \phi_{N+N_{\partial_1}}$ — bāzes funkcijas, kas interpolē nehomogēnu Dirihlē robežnosacījumu funkcijai ψ apgabala robežas segmentā Γ_1 , $\phi_{N+1}, \dots, \phi_{N+N_{\partial_2}}$ — bāzes funkcijas, kas interpolē nehomogēnu Dirihlē robežnosacījumu funkcijai ζ apgabala robežas segmentā Γ_3 . Abus divus atrisinājumus meklē formā

$$\psi = \sum_{j=1}^N \widehat{\psi}_j \phi_j + \sum_{j=N+1}^{N+N_{\partial_1}} \phi_j, \quad (7.1)$$

$$\zeta = \sum_{j=1}^N \widehat{\zeta}_j \phi_j + \sum_{j=N+1}^{N+N_{\partial_2}} \widehat{\zeta}_{0,j}^{(k)} \phi_j. \quad (7.2)$$

$\widehat{\psi}$, $\widehat{\zeta} \in \mathbb{R}^n$ - nezināmas vērtības.

1. Rerezentāciju (7.1) ievieto vienādojumā (6.13). Saskaņā ar Gaļorkina metodi (5.1.), iegūst lineāru sistēmu:

$$A\widehat{\psi} = g_1(\widehat{\zeta}),$$

kur

$$a(\phi_i, \phi_j) = \int_{\Omega} \nabla \phi_j \cdot \nabla \phi_i dx \quad - \quad (7.3)$$

— matricas A elementi.

Labās puses vektora elementi ir

$$g_1(i) = D\widehat{\zeta} - \sum_{j=N+1}^{N+N_{\partial_1}} 1 \cdot \int_{\Omega} \nabla \phi_j \cdot \nabla \phi_i dx,$$

kur matricas D elementi ir šādi:

$$d(\phi_i, \phi_j) = \int_{\Omega} \phi_j \phi_i dx. \quad (7.4)$$

2. Reprezentāciju (7.2) ievieto vienādojumā (6.14). Sastādot nedaudz sarežģītāku lineāro sistēmu, jāizmanto arī šāda reprezentācija funkcijai ψ :

$$\psi = \sum_{l=1}^N \widehat{\psi}_l \psi_l,$$

jo var redzēt, ka vienādojuma (6.14) abas puses ir atkarīgas no fiksētas funkcijas ψ . Atkal, saskaņā ar Gaļorkina metodi (5.1.), iegūst lineāru sistēmu:

$$B(\widehat{\psi})\widehat{\zeta} = g_2(\widehat{\psi}, \widehat{\zeta}_0^{(k)}),$$

kur

$$b(\phi_i, \phi_j) = -a(\phi_i, \phi_j) + \sum_{l=1}^N \widehat{\psi}_l \int_{\Omega} \nabla^{\perp} \phi_l \cdot \nabla \phi_j \phi_i dx, \quad (7.5)$$

un

$$g_2(i) = C\widehat{\psi} + \sum_{j=N+1}^{N+N_{\partial_2}} \widehat{\zeta}_{0,j}^{(k)} \int_{\Omega} \nabla \phi_j \cdot \nabla \phi_i dx - \sum_{j=N+1}^{N+N_{\partial_2}} \sum_{l=1}^N \widehat{\psi}_l \widehat{\zeta}_{0,j}^{(k)} \int_{\Omega} \nabla^{\perp} \phi_l \cdot \nabla \phi_j \phi_i dx.$$

Matricas C elementi ir formā

$$c(\phi_i, \phi_j) = \int_{\Omega} \widetilde{A} \nabla \phi_j \cdot \nabla \phi_i dx. \quad (7.6)$$

Līdz ar to, veidojas iterāciju process formā (7.7):

$$\begin{cases} A\widehat{\psi}^{(k)} = g_1(\widehat{\zeta}^{(k)}), \\ B(\widehat{\psi}^{(k)})\widehat{\zeta}^{(k+1)} = g_2(\widehat{\psi}^{(k)}, \widehat{\zeta}_0^{(k)}), \\ k = 0, 1, \dots, \end{cases} \quad (7.7)$$

kur:

$$\widehat{\psi}^{(k)}, \widehat{\zeta}^{(k)} \in \mathbb{R}^N, \quad \widehat{\zeta}_0^{(k)} \in \mathbb{R}^{N_{\partial_2}}.$$

Piezīme 7. Var redzēt, ka iterāciju process (6.8) robežnosacījuma uz lokiem Γ_3 iegūšanai tagad ir iekļauts galvenajā iterāciju procesā (7.7) lielumu $\widehat{\psi}$ un $\widehat{\zeta}$ noteikšanai. Faktiski, katrā iterāciju procesa (7.7) soļa ietvaros notiek arī viens procesa (6.8) solis.

8. Galīgo elementu metodes realizācija

7. nodaļā tika iegūtas matricas, kuras invertējot, dabū Gaļorkina atrisinājumu. Tagad mērķis ir iegūt galīgo elementu atrisinājumu. Tas nozīmē: definēt triangulāciju uz kopas Ω un izejot no pieņēmuma, ka bāzes funkcijas ir lineāras, aizpildīt matricas A, B, C un D . Literatūra diezgan bieži var sastapt matricu A un D veidošanas procesu Lagranža galīgo elementus izmantojot, angļu valodā tām pat ir nosaukumi - *stiffness matrix* un *mass matrix* [10].

Lai atrastu šo un arī pārējo matricu elementus, definētajā triangulācijā katru trīsstūri transformē uz vienības šūnu (vienības trīsstūri), un attiecīgās funkcijas integrē tikai pa šo vienības šūnu, izmantojot mainīgo maiņu. Sāksim ar nevienmērīgas triangulācijas, jeb nevienmērīga divdimensionāla režģa ieviešanu.

8.1. Nevienmērīgais režģis

Parasti praksē pieņemts izmantotas gatavas paketes (Matlab, utml). Bet šajā konkrētajā uzdevumā var iztikt ar analītisko izteiksmju izmantošanu un režģa uzdošanu ar saviem spēkiem. Tāda pieeja nav tik universāla, bet konkrētajā gadījumā tā dod pat priekšrocību: ģeometrijai ar riņķveida cilindriem tāds režģis dod palielināto precizitāti stūra punktos funkcijai ζ (kas ir svarīga informācija, jo funkcijai ζ ir lielas izmaiņas šajās vietās ([5])).

Uzkonstruēsim šādu režģi (10. attēls): Apzīmējot

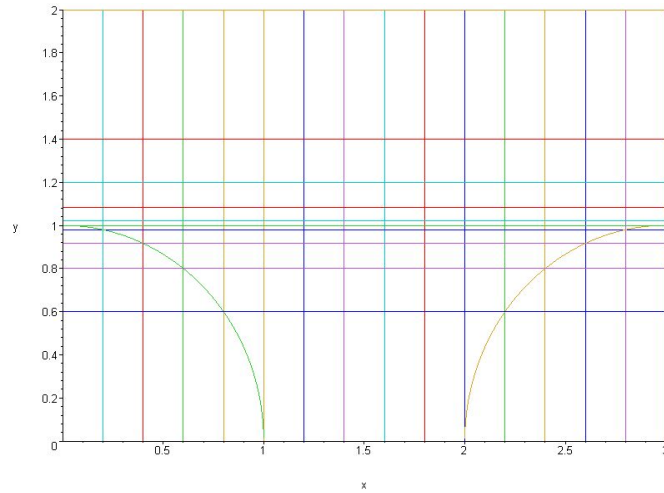
$$N_1 := \frac{R}{h}, \quad N := \frac{L_1}{h},$$

ievedam šādas funkcijas:

$$\begin{aligned} f_1(x) &:= \sqrt{1-x^2}, \\ f_2(x) &:= -\sqrt{1-x^2} + 2, \\ f_3(x) &:= \sqrt{1-(x-3)^2}, \end{aligned}$$

Režģi var uzdot ar kopu (8.1):

$$\begin{aligned} \Omega_h &:= \{(x, y) : x = ih, y = f_1(jh), y = f_2(jh), \\ & y \geq f_{1,3}(x), i = 0..N, j = 0..N_1\}. \end{aligned} \tag{8.1}$$



10. att. Nevienmērīgais režģis

Piezīme 8. *Kā jau bija teikts piezīmē 5, funkciju f_1 , f_3 (kas atbilst tām pašām funkcijām no 6. nodaļas) vietā var izvēlēties arī citas un pieeja režģa definēšanai paliks nemainīga. Šajā gadījumā būtu jāmaina arī funkcija f_2 .*

Būtu diezgan grūti tādā pašā veidā (analītiski) uzdot arī atbilstošu režģim (8.1) triangulāciju, var tikai pateikt, ka tā veidojas konstruējot taisnstūru diagonāles, kas sakrītīs ar trīsstūru hipotenūzām.

8.2. Afīnās transformācijas

Dotajā triangulācijā, kas atbilst režģim (8.1), var ieraudzīt tikai triju tipu trīsstūrus (11. attēls, ar h_1 apzīmēts nevienmērīgais solis):

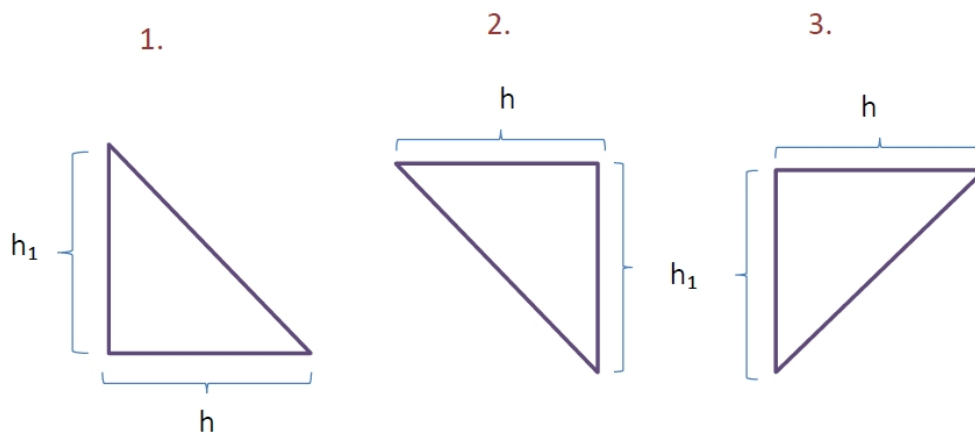
Var viegli pārlicināties, ka attiecīgo katešu garumi sakrīt ar nevienmērīgā režģa (8.1) soliem. Katrs no tiem jātransformē uz *vienības šūnu* (12.. attēls).

Definīcija 27. Par vienības šūnu sauc kopu

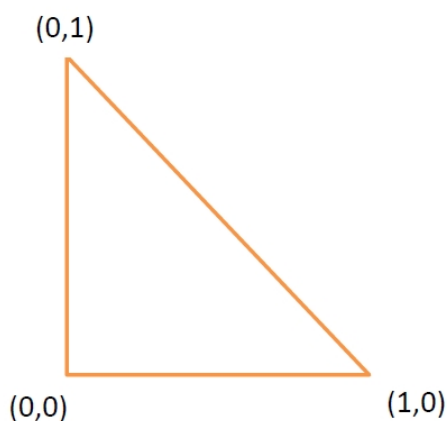
$$T_0 = \left\{ \begin{pmatrix} \xi \\ \eta \end{pmatrix} : \xi \geq 0, \eta \geq 0, \eta \geq 1 - \xi \right\}. \quad (8.2)$$

Jāatrod nesingulāra matrica $B_T \in \mathbb{R}^{2 \times 2}$ un vektors $b_T \in \mathbb{R}^2$ bijektīvam attēlojumam:

$$\begin{pmatrix} x \\ y \end{pmatrix} = B_T \begin{pmatrix} \xi \\ \eta \end{pmatrix} + b_T, \quad (8.3)$$



11. att. Triju veidu trīsstūri



12. att. Vienības šūna

kur $\begin{pmatrix} x \\ y \end{pmatrix}$ - koordinātes, kurās uzdots režģis (8.1), $\begin{pmatrix} \xi \\ \eta \end{pmatrix}$ - koordinātes, kurās uzdots vienības šūna (8.2).

Apskatīsim patvaļīgu trīsstūru, uzdotu koordinātēs $\begin{pmatrix} x \\ y \end{pmatrix}$, ar virsotnēm punktos $\begin{pmatrix} x_1 \\ y_1 \end{pmatrix}$, $\begin{pmatrix} x_2 \\ y_2 \end{pmatrix}$ un $\begin{pmatrix} x_3 \\ y_3 \end{pmatrix}$. Katra punkta transformācija uz attiecīgo vienības šūnas (8.2) virsotni noved pie lineārās sistēmas ar sešiem nezināmiem (matricas B_T , vektora b_T elementu

noteikšanai (8.4)):

$$\begin{cases} \begin{pmatrix} x_1 \\ y_1 \end{pmatrix} = B_T \begin{pmatrix} 0 \\ 0 \end{pmatrix} + b_T, \\ \begin{pmatrix} x_2 \\ y_2 \end{pmatrix} = B_T \begin{pmatrix} 1 \\ 0 \end{pmatrix} + b_T, \\ \begin{pmatrix} x_3 \\ y_3 \end{pmatrix} = B_T \begin{pmatrix} 0 \\ 1 \end{pmatrix} + b_T. \end{cases} \quad (8.4)$$

no kurienes viennozīmīgi seko, ka $B_T = \begin{pmatrix} x_2 - x_1 & x_3 - x_1 \\ y_2 - y_1 & y_3 - y_1 \end{pmatrix}$, $b_T = \begin{pmatrix} x_1 \\ y_1 \end{pmatrix}$.

Var viegli saprast, ka pirmajā situācijā 11. attēlā, $B_T = \begin{pmatrix} h & 0 \\ 0 & h_1 \end{pmatrix}$, otrajā - $B_T = \begin{pmatrix} -h & 0 \\ 0 & -h_1 \end{pmatrix}$, trešajā - $B_T = \begin{pmatrix} h & 0 \\ 0 & -h_1 \end{pmatrix}$.

Transformācijas jakobiāns pirmajā un otrajā situācijās vienāds ar hh_1 , trešajā - $-hh_1$, jeb

$$|\det B_T| = hh_1. \quad (8.5)$$

Acīmredzot, $|\det B_T| \neq 0$, režģa konstrukcijas dēļ.

7. nodaļā tika uzkonstruēts iterāciju process (7.7) un atrastas formulas tajā ietilpstošu matricu A , B , C , un D elementu atrašanai. Skaidrs, ka katram uzdevuma risināšanai izvēlētajam režģim šie elementi būs dažādi. Izmantojot izveidotās šajā nodaļā afinās transformācijas, var sarēķināt šos skaitļus konkrētajām režģim (8.1).

Uz vienas vienības šūnas izvēloties gabaliem lineārās bāzes funkcijas attiecīgajā telpā, var būt definētas tikai trīs lineāras funkcijas, proti:

$$\begin{aligned} \tilde{\phi}_1(\xi, \eta) &= 1 - \xi - \eta, \\ \tilde{\phi}_2(\xi, \eta) &= \xi, \\ \tilde{\phi}_3(\xi, \eta) &= \eta, \end{aligned}$$

un līdz ar to,

$$\nabla \tilde{\phi}_1 = \begin{pmatrix} -1 & -1 \end{pmatrix}, \quad \nabla \tilde{\phi}_2 = \begin{pmatrix} 1 & 0 \end{pmatrix}, \quad \nabla \tilde{\phi}_3 = \begin{pmatrix} 0 & 1 \end{pmatrix}.$$

Piezīme 9. Fiksējot kādu trīsstūri triangulācijā (8.1), visus vajadzīgos operatorus aplūkojot galīgajās telpās, izmantota viena un tā paša lineāra transformācija. Bet katrai matricai no saraksta vajadzīga sava pieceja.

8.3. Matricu elementu atrašana

Aizpildot milzīgas matricas, kas rodas galīgo elementu metodi pielietojot, parasti izmanto palīdzinstrumentu: *lokālas matricas*. Triangulācijas gadījumā tās ir 3×3 matricas $A_T := (a(\phi_i, \phi_j))_{i,j}$, $i, j = 1..3$. Galīgo elementu kodu (A) sastādot, globālās matricas A , B , C , D tika aizpildītas izmantojot lokālās A_T , B_T , C_T , D_T un vizuālo režģa (8.1) analīzi.

1. Atmetot zināmas vērtības uz robežas (6.1) segmenta Γ_1 , formula funkcijas ψ noteikšanai k -tajā iterāciju procesa (7.7) solī ir

$$\psi = \sum_{j=1}^N \widehat{\psi}_j \phi_j, \quad (8.6)$$

kur $\widehat{\psi} \in \mathbb{R}^N$ - vektors ar nezināmām vērtībām režģa (8.1) punktos.

Matrica A bija uzdots formā (7.3):

$$a(\phi_i, \phi_j) = \int_{\Omega} \nabla \phi_j \cdot \nabla \phi_i dx.$$

Tagad var uzrakstīt:

$$\int_{\Omega} \nabla \phi_j \cdot \nabla \phi_i dx = \sum_T \int_T \nabla \phi_j \cdot \nabla \phi_i dx =: \sum_T a_T(\phi_i, \phi_j).$$

Līdzīgi bija uzdots arī matrica C (7.6):

$$c(\phi_i, \phi_j) = \int_{\Omega} \tilde{A} \nabla \phi_j \cdot \nabla \phi_i dx,$$

un var rakstīt:

$$\int_{\Omega} \tilde{A} \nabla \phi_j \cdot \nabla \phi_i dx = \sum_T \int_T \tilde{A} \nabla \phi_j \cdot \nabla \phi_i dx =: \sum_T c_T(\phi_i, \phi_j).$$

Transformējot uz vienības šūnu T_0 , (8.2), jāsarēķina gradienti jaunajās koordinātēs.

Apzīmējot gradientu koordinātēs $\begin{pmatrix} x \\ y \end{pmatrix}$ ar ∇_x , un gradientu koordinātēs $\begin{pmatrix} \xi \\ \eta \end{pmatrix}$ ar ∇_{ξ} , pārveido:

$$\bar{A} \nabla_x \phi = \bar{A} \begin{pmatrix} \phi_x \\ \phi_y \end{pmatrix} = \bar{A} \begin{pmatrix} \phi_{\xi} \xi_x + \phi_{\eta} \eta_x \\ \phi_{\xi} \xi_y + \phi_{\eta} \eta_y \end{pmatrix} = \bar{A} \cdot (\nabla_{\xi} \phi \cdot B_T^{-1})^T. \quad (8.7)$$

Izvēloties $\bar{A} = E$, iegūst vajadzīgos operatorus izteiksmē $a(\phi_i, \phi_j)$, bet pieņemot $\bar{A} = \tilde{A}$, var arī iegūt operatoru izteiksmē $c(\phi_i, \phi_j)$ jaunajās koordinātēs. Iegūst sekojošo. Matricai A_T :

$$a_T(\phi_i, \phi_j) = |\det B_T| \int_{T_0} \nabla \tilde{\phi}_j B_T^{-1} \cdot \nabla \tilde{\phi}_i B_T^{-1} d\xi d\eta, \quad (8.8)$$

matricai C_T :

$$c_T(\phi_i, \phi_j) = |\det B_T| \int_{T_0} \tilde{A} \nabla \tilde{\phi}_j B_T^{-1} \cdot \nabla \tilde{\phi}_i B_T^{-1} d\xi d\eta, \quad (8.9)$$

2. Atmetot zināmas vērtības uz robežas (6.1) segmenta Γ_3 , formula funkcijas ζ noteikšanai k -tajā iterāciju procesa (7.7) solī ir

$$\zeta = \sum_{j=1}^N \hat{\zeta}_j \phi_j, \quad (8.10)$$

kur $\hat{\zeta} \in \mathbb{R}^N$ - vektors ar nezināmām vērtībām režģa (8.1) punktos. Matricas B elementu atrašana ir vissarežģītākā no programmēšanas viedokļa procedūra, jo formulā (7.5) var redzēt, ka tā vienmēr ir atkarīga no kādām iepriekš iegūtām vērtībām. Uzrakstīsim šo formulu vēlreiz:

$$b(\phi_i, \phi_j) = -a(\phi_i, \phi_j) + \sum_{l=1}^N \hat{\psi}_l^{(k)} \int_{\Omega} \nabla^\perp \phi_l \cdot \nabla \phi_j \phi_i dx$$

Nezināms tagad paliek tikai otrais saskaitāmais, kuru apzīmē ar $J(\phi_i, \phi_j)$:

$$J(\phi_i, \phi_j) = \sum_T \sum_{l=1}^N \hat{\psi}_l^{(k)} \int_T \nabla^\perp \phi_l \cdot \nabla \phi_j \phi_i dx,$$

Zinot, ka $\nabla^\perp = \bar{A} \nabla$, kur $\bar{A} = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$, var izmantot jau iepriekš iegūto rezultātu (8.7) un rakstīt:

$$j_T(\phi_i, \phi_j) = |\det B_T| \int_{T_0} \bar{A} \nabla (a_1 \tilde{\phi}_1 + a_2 \tilde{\phi}_2 + a_3 \tilde{\phi}_3) B_T^{-1} \cdot \nabla \tilde{\phi}_j B_T^{-1} \tilde{\phi}_i d\xi d\eta, \quad (8.11)$$

kur a_1 , a_2 , un a_3 - iepriekšējā iterācijā iegūtas funkcijas ψ vērtības trīsstūra T virsotnēs. To atrašana ir tehnisks darbs ar masīviem $\hat{\psi}^{(k)}$: (pielikums A).

Matricai D_T vienkārši var rakstīt

$$d_T(\phi_i, \phi_j) = |\det B_T| \int_{T_0} \tilde{\phi}_j \tilde{\phi}_i d\xi d\eta. \quad (8.12)$$

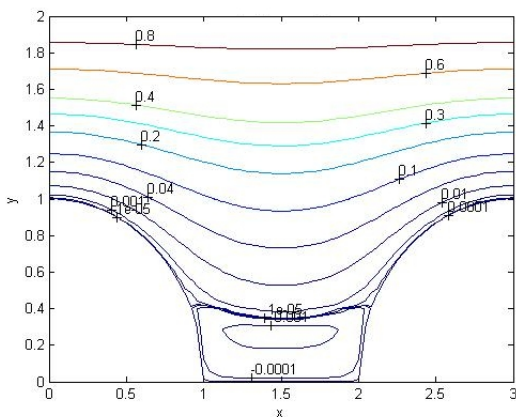
9. Rezultātu grafiskā attēlošana

Tagad ar datora palīdzību, izmantojot lokālās matricas (8.8, 8.9, 8.11, 8.12), tika aizpildītas globālās (7.3, 7.4, 7.5, 7.4) un realizēts iterāciju process (7.7). Kods atrodams pielikumā (A).

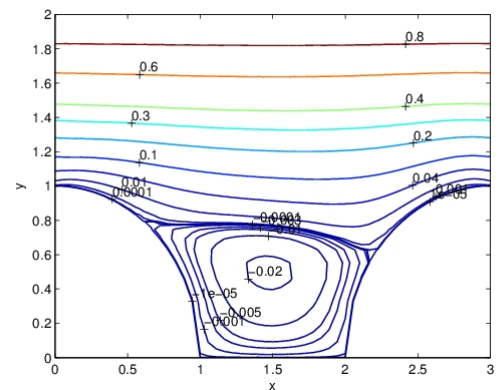
Izmantojot kodu (A), var iegūt strāvas un virpuļu funkciju sadalījumus - vērtības režģa (8.1) punktos. Grafiski mūs interesēs tikai strāvas funkcijas sadalījumi. Katra izveidota bilde nozīme ka process ir nokonverģējis, t.i. ir spēkā $\max_i |\hat{\zeta}^{(k)} - \hat{\zeta}^{(k-1)}| \leq \varepsilon$. Visos gadījumos ε nepārsniedz 10^{-2} . Palielinot parametra Re vērtību (jau sākot ar $Re = 3$), ir nepieciešama apakšēja relaksācija formulai (6.8). Pa visiem apskatītiem gadījumiem, $\min \beta = 10^{-2}$.

Datorpaketē MATLAB ir izveidotas attiecīgas bildes. Attēlojot funkcijas ψ līmeņlīnijas uz plaknes tiek izmantots nevienmērīgais režģis. Attēlojot ātruma vektoru laukus uz plaknes, tiek izmantots vienmērīgais režģis, kuru iegūst režģi (8.1) analītiski uzdodot izmantojot lineārās funkcijas. Kā jau bija teikts piezīmēs 5,8, tāda iespēja pastāv un uzdevums šajā gadījumā tiek reducēts uz rombu aptecēšanas uzdevumu divās dimensijās (prizmu aptecēšana trijās).

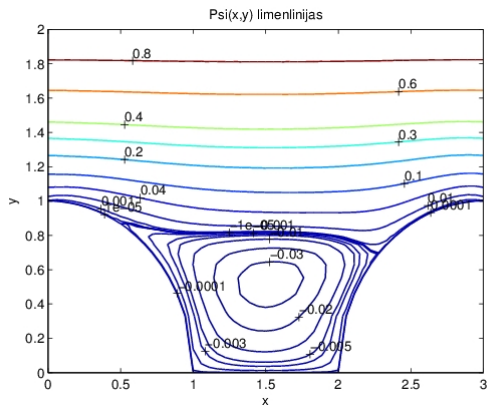
Sākumā attēlosim uz plaknes strāvas funkcijas līmeņlīnijas. Mainot parametrus Re un S - Reynoldsa un Stjūarta skaitļus, var iegūt dažādus rezultātus. Pielietotājus parasti interesē virpuļu sadalījumi, t.i. funkcijas ψ negatīvas vērtības. Līmeņlīniju pieeja ($\psi = const$) dod iespēju ātri ieraudzīt ģeometriskas vietas, kurās virpuļi veidojas.



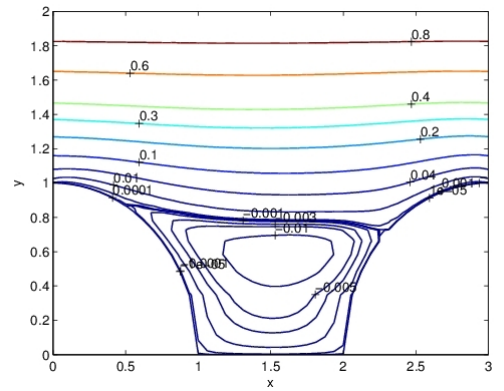
13. att.: Funkcijas ψ līmeņlīnijas.
 $Re = 0, S = 0$.



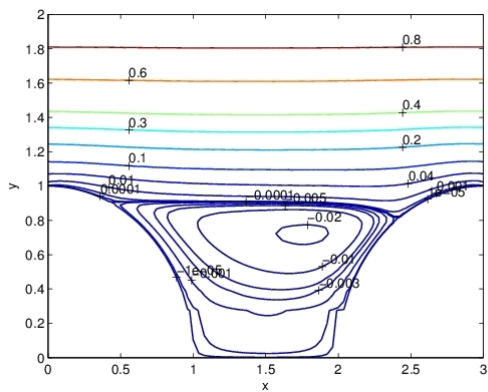
14. att.: Funkcijas ψ līmeņlīnijas.
 $Re = 40, S = 0$.



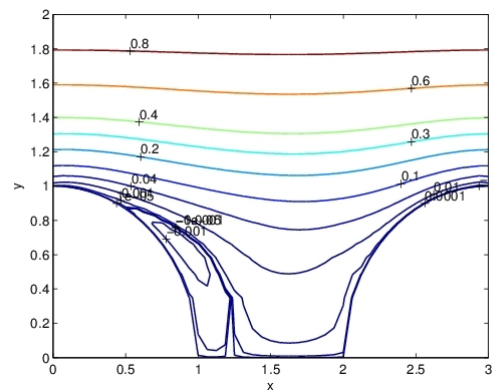
15. att.: Funkcijas ψ līmeņlīnijas.
 $Re = 100, S = 0.$



16. att.: Funkcijas ψ līmeņlīnijas.
 $Re = 100, S = 2, \alpha = 0.$

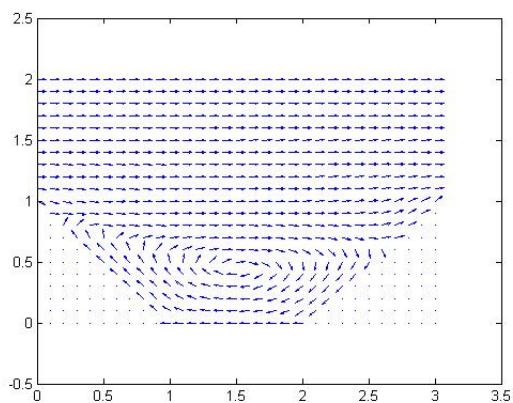


17. att.: Funkcijas ψ līmeņlīnijas.
 $Re = 100, S = 2, \alpha = 0.$

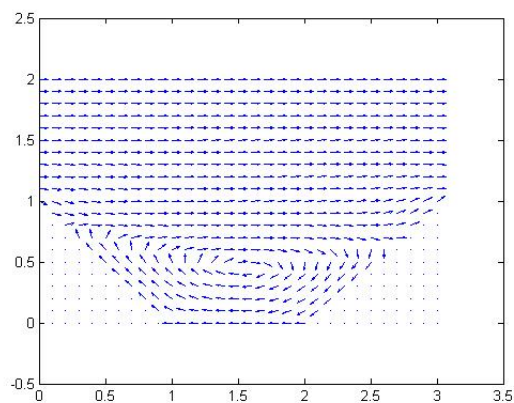


18. att.: Funkcijas ψ līmeņlīnijas.
 $Re = 100, S = 2, \alpha = \frac{\pi}{2}.$

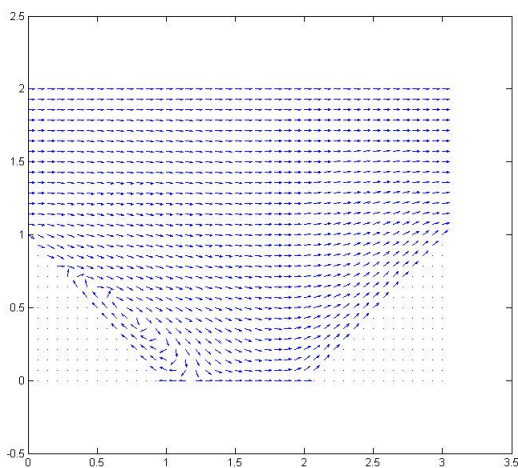
Tagad attēlosim arī vektoru laukus:



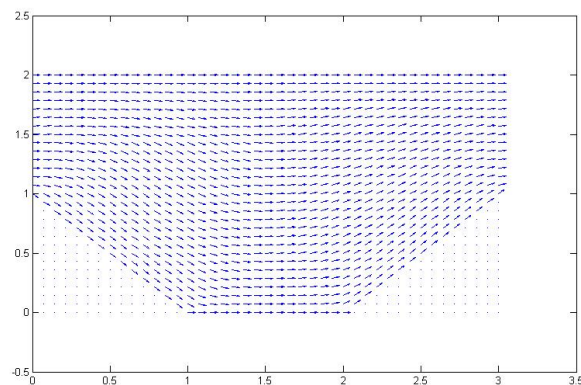
23. att.: \mathbf{V} vektoru lauks. $Re = 100, S = 0$.



24. att.: \mathbf{V} vektoru lauks. $Re = 100, S = 2, \alpha = 0$.



25. att.: \mathbf{V} vektoru lauks. $Re = 100, S = 1, \alpha = \frac{\pi}{2}$.



26. att.: \mathbf{V} vektoru lauks. $Re = 100, S = 20, \alpha = \frac{\pi}{2}$.

26. attēlā atkal var redzēt, ka virpuļi tika pilnībā 'iznīcināti' pie lielām parametra S vērtībām leņķim $\alpha = \frac{\pi}{2}$, kas ir saskaņā ar teoriju ([6]).

Secinājumi

Maģistra darbā diezgan sarežģīts Navjē-Stoksa formulējums tika reducēts uz vienkāršāko, kuram ir izveidots risināšanas algoritms. Izmantojot datorpaketi MATLAB, realizēts kods, sastādīts no izveidota algoritma. Problēmas diskretizācija veikta patstāvīgi, analītiski uzdodot režģi, līdz ar to, koda veidošanas procesā nebija izmantotas gatavas MATLAB paketes diferenciālvienādojumu risināšanai.

Datorpakete MATLAB ir izvēlēta jo iegūtais algoritms pieļauj plašu darbošanos ar masīviem (daudzkārtīgas transponēšanas, matricu invertēšanas, utml.)

Pētījumu varētu attīstīt vēl šādā virzienā: mainot ģeometriju, (piemēram ievēšot dziļākas bedres) var iegūt jaunus strāvas, virpuļu funkciju, spiediena sadalījumus.

Izmantotā literatūra un avoti

- [1] Ф. Сьярле. *Метод Конечных Элементов для Эллиптических Задач*. Мир, 1980.
- [2] Robert C.Rogers Michael Renardy. *An Introduction to Partial Differential Equations*. Springer, 1993.
- [3] Wolfgang Hackbusch. *Theorie und Numerik elliptischer Differentialgleichungen*. Teubner Verlag, 1996.
- [4] Р. Уэйт Э. Митчелл. *Метод Конечных Элементов для Уравнений с Частными Производными*. Мир, 1981.
- [5] П.Роуч. *Вычислительная Гидродинамика*. Мир, 1980.
- [6] Э.В.Щербинин Ю.М.Гельфгат, О.А.Лиелаусис. *Жидкий металл под действием электромагнитных сил*. Zinātne, 1976.
- [7] Х.Э.Калис. *Обобщенные решения уравнений математической физики*. Latvijas Universitāte, 1984.
- [8] Х.Э.Калис. *Вариационно-разностные методы для решения краевых задач*. Latvijas Universitāte, 1989.
- [9] К.Эйшлг А.Том. *Числовые расчеты полей в технике и физике*. Энергия, 1964.
- [10] P.Yardley G.Evans, K.Blackledge. *Numerical Methods for Partial Differential Equations*. Springer, 2000.
- [11] <http://mathworld.wolfram.com/DivergenceTheorem.html>.

A Pielikums

Definīcija 28. [3] Ierobežotam apgabalam $\Omega \subset \mathbb{R}^n$ ir nepārtraukta pēc Lipsīca robeža, ja

$$\forall x \in \partial\Omega \quad \exists B(x) \subset O_i :$$

$$O_i \cap \Omega = O_i \cap \Omega_i,$$

kur $B(x)$ - sfēra, O_i , $i = 1..M$ - vaļējās kopas,

$$\Omega_i = \{(x_1, x_2) \in \mathbb{R}^n : x_1 \in \mathbb{R}^{n-1}, x_2 \in \mathbb{R}, x_2 < \phi_i(x_1)\},$$

ϕ_i , $i = 1..M$ - funkcijas, kas apmierina Lipsīca nosacījumu:

$$\exists L > 0 \quad \forall x, y \in \partial\Omega \quad |\phi_i(x) - \phi_i(y)| \leq L|x - y|.$$

Teorēma 10. [11] Apgabalam $\Omega \subset \mathbb{R}^2$ ar robežu $\partial\Omega$, ir spēkā:

$$\int_{\Omega} \nabla \cdot u \, dx = \int_{\partial\Omega} (u^T n) d\sigma,$$

kur $u := \begin{pmatrix} u_1 \\ u_2 \end{pmatrix}$ - vektoru lauks, n - apgabala Ω ārējā normāle.

Definīcija 29. [3] Par klasi C_0^n sauc funkciju klasi, kas sevī ietver n reizes nepārtraukti diferencējamas funkcijas ar kompaktu nesēju, proti:

$$C_0^n(\Omega) := \{u \in C^n(\Omega), \text{supp } u - \text{kompakta kopa}\},$$

kur $\text{supp } u := \overline{\{x \in \Omega : u(x) \neq 0\}}$ - funkcijas nesējs.

Apgalvojums 11. [7] (Koši nevienādība telpā $L^2(\Omega)$)

$$\forall u, v \in L^2(\Omega) : (u, v)_{L^2} \leq \|u\| \cdot \|v\|.$$

Apgalvojums 12. (*Formula 3×3 matricas \bar{A} inversas matricas aprēķināšanai*)

$$\bar{A}^{-1} = \frac{1}{\det \bar{A}} \begin{pmatrix} A & D & G \\ B & E & H \\ C & F & K \end{pmatrix},$$

kur $\bar{A} := \begin{pmatrix} a & b & c \\ d & e & f \\ g & h & k \end{pmatrix}$, $A := ek - fh$, $B := fg - dk$, $C := dh - eg$, $D := ch - bk$, $E := ak - cg$, $F := gb - ah$, $G := bf - ce$, $H := cd - af$, $K := ae - bd$.

A Izveidoto programmu kods

```
function fem_psi_dzeta2

%galvenie parametri:
%R-radiusss; N1-punktu skaits uz rinka linijas;

h=1/14; R=1; N1=R/h+1;
L=3*R; L1=2*R; N2=(L-2*R)/h+1;
N=L/h+1;
M=2*N1-1;

fun1=@(x)sqrt(R^2-x.^2); fun2=@(x)-sqrt(R^2-x.^2)+2*R;
%fun1=@(x)R-x; fun2=@(x)x+R;
w1=[-0.00001,-0.0001,-0.003,-0.005,-0.05,-0.08,-0.07,-0.06,-0.1,-0.2,
-0.12,-0.15,-0.001,-0.01,-0.03,-0.02,-0.04,-0.05,0.00001, 0.0001,
0.001,0.01,0.04,0.1,0.2,0.3,0.4,0.6,0.8,1.1,1.2];
divAj=[0 -1;1 0];

%Fizikalie parametri
Re=100; S=0; alpha=0;
divA=[cos(alpha)^2 sin(alpha)*cos(alpha);sin(alpha)*cos(alpha) sin(alpha)^2];

%number of interior mesh points in the 1st subarea
num1=0;
```

```

for i=1:N1-1
    num1=num1+N2+2*(i-1);
end
%number of interior mesh points in the 2nd subarea
num2=(N-2)*floor((M-2)/2);

%dimensions
dim=num1+num2+floor((M-2)/2);
dim2=dim+N-2;
dim3=N1-1;
dimper=floor((M-2)/2);
totdim=dim2+2+2*dim3;

%meshgrid:
for i=1:M

    if i<=N1 Y(i,:)= fun1((N1-i)*h) *ones(N,1); end
    if i>N1 Y(i,:)= fun2((i-N1)*h) *ones(N,1); end

    X(i,:)=L:-h:0;
end

%afiinas transformacijas

function [h1_1,h1_2] = soli(row,fun)
    h1_1=abs(fun((N1-row)*h)-fun((N1-row-1)*h));
    h1_2=abs(fun((N1-row-1)*h)-fun((N1-row-2)*h));

end

%gradientiem

```

```

function btt1=bt1(h1)
    btt1=(h1*h)^(-1)*[h1 0;0 h];
end
function btt2=bt2(h1)
    btt2=(h1*h)^(-1)*[-h1 0;0 -h];
end
function btt3=bt3(h1)
    btt3=(-h1*h)^(-1)*[-h1 0;0 h];
end
% der1=sym('der1');
% der2=sym('der2');

%local stiffness matrices
function ls = lsm(aa,tpe,bt,h1,ii,jj)

    gradientsj=[-1 -1;1 0;0 1]; gradientsi=[-1 -1;1 0;0 1];
    jacobian=h*h1;

    if tpe==1
        for kk=1:3
            for ll=1:3
                lsmx(kk,ll)=(jacobian/2)*(gradientsj(kk,:)*bt)
                    *(gradientsi(ll,:)*bt)';
            end
        end
    end

    if tpe==3

```

```

for kk=1:3
    for ll=1:3
        lsmx(kk,ll)=(jacobian/2)*
        (divA*bt*gradientsj(kk,:))'*(gradientsi(ll, :)*bt)';
    end
end
end

if tpe==4

lsmx=jacobian*[1/12,1/24,1/24;1/24,1/12,1/24;1/24,1/24,1/12];

end

if tpe==5
    for kk=1:3
        for ll=1:3
            lsmx(kk,ll)=(jacobian/6)*(divAj*bt*(aa(1)
            *gradientsj(1,:)+
            aa(2)*gradientsj(2,:)+aa(3)*
            gradientsj(3,:))'*(gradientsi(kk, :)*bt)';
        end
    end
end

end

```

```

%heat and magnetic_field
if tpe==6
    for kk=1:3
        for ll=1:3
            lsmx(kk,ll)=(jacobian/6)*
            ([cos(alpha),sin(alpha)]*(aa(1)*gradientsj(1,:)
            +aa(2)*gradientsj(2,)+aa(3)
            *gradientsj(3,))')^2;
        end
    end
end

if tpe==7
    for kk=1:3
        for ll=1:3
            lsmx(kk,ll)=(jacobian/2)*(divA*bt*(aa(1)
            *gradientsj(1,)+
            +aa(2)*gradientsj(2,)+aa(3)*
            gradientsj(3,))')*(gradientsi(kk,)*bt)';
        end
    end
end

ls=lsmx(ii,jj);

end

ss=zeros(num1+num2+floor((M-2)/2),1);
mas=zeros(3,6,dim-dimper+2*dim3);
function A=stiffness(tpe)

```

```

%galvenais cikls; global stiffness matrix and a load vector;
row=1; rowdim=N2; barrier=rowdim; k=0; tom=[1;
rowdim;abs(fun1((N1-3)*h)-fun1((N1-2)*h));
fun1((N1-2)*h)];k1=1;
A(1,dim2+2+k1)=lsm(mas(:,4,1),tpe,bt2(tom(4,1)),tom(4,1),1,2)
+lsm(mas(:,3,1),tpe,bt1(tom(3,1)),tom(3,1),2,1);
A(dim2+2+k1,1)=lsm(mas(:,4,1),tpe,bt2(tom(4,1)),tom(4,1),2,1)
+lsm(mas(:,3,1),tpe,bt1(tom(3,1)),tom(3,1),1,2);

    for i=1:num1

        if i>barrier

row=row+1; rowdim=N2+2*(row-1); barrier=barrier+rowdim;
[h1_1,h1_2] = soli(row,fun1);
A(i,dim2+1+k1)=lsm(mas(:,4,i),tpe,bt2(h1_1),h1_1,1,3)
+lsm(mas(:,5,i),tpe,bt1(h1_1),h1_1,3,1);
A(dim2+1+k1,i)=lsm(mas(:,4,i),tpe,bt2(h1_1),h1_1,3,1)
+lsm(mas(:,5,i),tpe,bt1(h1_1),h1_1,1,3);
k1=k1+1; A(i,dim2+2+k1)=lsm(mas(:,4,i),tpe,bt2(h1_1),h1_1,1,2)
+lsm(mas(:,3,i),tpe,bt1(h1_2),h1_2,2,1);
    A(dim2+2+k1,i)=lsm(mas(:,4,i),tpe,bt2(h1_1),h1_1,2,1)
+lsm(mas(:,3,i),tpe,bt1(h1_2),h1_2,1,2);
tom=[tom,[i;rowdim;h1_2;h1_1]];

        end

```

```

[h1_1,h1_2] = soli(row,fun1);

if i<barrier A(i,i)=lsm(mas(:,1,i),tpe,bt1(h1_2),h1_2,1,1)
+lsm(mas(:,3,i),tpe,bt1(h1_2),h1_2,2,2)
+lsm(mas(:,2,i),tpe,bt2(h1_2),h1_2,3,3)
+lsm(mas(:,4,i),tpe,bt2(h1_1),h1_1,1,1)
+lsm(mas(:,6,i),tpe,bt2(h1_1),h1_1,2,2)
+lsm(mas(:,5,i),tpe,bt1(h1_1),h1_1,3,3);
    if tpe==5 hsc(i)=lsm(mas(:,1,i),6,bt1(h1_2),h1_2,1,1)
+lsm(mas(:,3,i),6,bt1(h1_2),h1_2,2,2)
+lsm(mas(:,2,i),6,bt2(h1_2),h1_2,3,3)
+lsm(mas(:,4,i),6,bt2(h1_1),h1_1,1,1)
+lsm(mas(:,6,i),6,bt2(h1_1),h1_1,2,2)
+lsm(mas(:,5,i),6,bt1(h1_1),h1_1,3,3);
    cc(i)=lsm(mas(:,1,i),7,bt1(h1_2),h1_2,1,1)
+lsm(mas(:,3,i),7,bt1(h1_2),h1_2,2,2)
+lsm(mas(:,2,i),7,bt2(h1_2),h1_2,3,3)
+lsm(mas(:,4,i),7,bt2(h1_1),h1_1,1,1)
+lsm(mas(:,6,i),7,bt2(h1_1),h1_1,2,2)
+lsm(mas(:,5,i),7,bt1(h1_1),h1_1,3,3);
    end
    A(i,i+1)=lsm(mas(:,1,i),tpe,bt1(h1_2),h1_2,1,2)
+lsm(mas(:,6,i),tpe,bt2(h1_1),h1_1,2,1);
A(i+1,i)=lsm(mas(:,1,i),tpe,bt1(h1_2),h1_2,2,1)
+lsm(mas(:,6,i),tpe,bt2(h1_1),h1_1,1,2);
end

if i==barrier
    if row>1 A(i,dim2+1+k1)=lsm(mas(:,5,i),
tpe,bt3(h1_1),h1_1,1,3)
+lsm(mas(:,4,i),tpe,bt2(h1_1),h1_1,1,3);

```

```

A(dim2+1+k1,i)=lsm(mas(:,5,i),tpe,bt3(h1_1),h1_1,3,1)
+lsm(mas(:,4,i),tpe,bt2(h1_1),h1_1,3,1);
A(i-1,dim2+1+k1)=lsm(mas(:,5,i-1),tpe,bt1(h1_1),h1_1,3,2)
+lsm(mas(:,6,i-1),tpe,bt2(h1_1),h1_1,2,3);

A(dim2+1+k1,i-1)=lsm(mas(:,5,i-1),tpe,bt1(h1_1),h1_1,2,3)
+lsm(mas(:,6,i-1),tpe,bt2(h1_1),h1_1,3,2);
end
k1=k1+1; A(i,dim2+2+k1)=lsm(mas(:,5,i),tpe,bt3(h1_1),h1_1,1,2)
+lsm(mas(:,1,i),tpe,bt1(h1_2),h1_2,1,2);
A(dim2+2+k1,i)=lsm(mas(:,5,i),tpe,bt3(h1_1),h1_1,2,1)
+lsm(mas(:,1,i),tpe,bt1(h1_2),h1_2,2,1);

A(i,i)=lsm(mas(:,1,i),tpe,bt1(h1_2),h1_2,1,1)
+lsm(mas(:,3,i),tpe,bt1(h1_2),h1_2,2,2)
+lsm(mas(:,2,i),tpe,bt2(h1_2),h1_2,3,3)
+lsm(mas(:,4,i),tpe,bt2(h1_1),h1_1,1,1)
+lsm(mas(:,5,i),tpe,bt3(h1_1),h1_1,1,1);
if tpe==5 hsc(i)=lsm(mas(:,1,i),6,bt1(h1_2),h1_2,1,1)
+lsm(mas(:,3,i),6,bt1(h1_2),h1_2,2,2)
+lsm(mas(:,2,i),6,bt2(h1_2),h1_2,3,3)
+lsm(mas(:,4,i),6,bt2(h1_1),h1_1,1,1)
+lsm(mas(:,5,i),6,bt3(h1_1),h1_1,1,1);
cc(i)=lsm(mas(:,1,i),7,bt1(h1_2),h1_2,1,1)
+lsm(mas(:,3,i),7,bt1(h1_2),h1_2,2,2)
+lsm(mas(:,2,i),7,bt2(h1_2),h1_2,3,3)
+lsm(mas(:,4,i),7,bt2(h1_1),h1_1,1,1)
+lsm(mas(:,5,i),7,bt3(h1_1),h1_1,1,1);
end

```

```

        if i==num1 tom=[tom,[i;rowdim;h1_2;h1_1]]; else
        tom=[tom,[i;rowdim+2;h1_2;h1_1]]; end
    end

    if (rowdim+i+1)<=num1
    A(i,rowdim+i+1)=lsm(mas(:,1,i),tpe,bt1(h1_2),h1_2,1,3)
    +lsm(mas(:,2,i),tpe,bt2(h1_2),h1_2,3,1);
    A(rowdim+i+1,i)=lsm(mas(:,2,i),tpe,bt2(h1_2),h1_2,1,3)
    +lsm(mas(:,1,i),tpe,bt1(h1_2),h1_2,3,1);
    A(i,rowdim+i)=lsm(mas(:,2,i),tpe,bt2(h1_2),h1_2,3,2)
    +lsm(mas(:,3,i),tpe,bt1(h1_2),h1_2,2,3);
    A(rowdim+i,i)=lsm(mas(:,2,i),tpe,bt2(h1_2),h1_2,2,3)
    +lsm(mas(:,3,i),tpe,bt1(h1_2),h1_2,3,2);

    end

    if row==round((M-2)/2)
    A(i,rowdim+i)=lsm(mas(:,1,i),tpe,bt1(h1_2),h1_2,1,3)
    +lsm(mas(:,2,i),tpe,bt2(h1_2),h1_2,3,1);
    A(rowdim+i,i)=lsm(mas(:,2,i),tpe,bt2(h1_2),h1_2,1,3)
    +lsm(mas(:,1,i),tpe,bt1(h1_2),h1_2,3,1);
    A(i,rowdim+i-1)=lsm(mas(:,2,i),tpe,bt2(h1_2),h1_2,3,2)
    +lsm(mas(:,3,i),tpe,bt1(h1_2),h1_2,2,3);
    A(rowdim+i-1,i)=lsm(mas(:,2,i),tpe,bt2(h1_2),h1_2,2,3)
    +lsm(mas(:,3,i),tpe,bt1(h1_2),h1_2,3,2);
    end

    ff(:,i)=[row;h1_1;h1_2;rowdim;barrier];

end

```

```

for i=num1+1:num1+num2
    if i>barrier row=row+1; barrier=barrier+rowdim; end
[h1_1,h1_2] = soli(row,fun2);

%%%%%dirichlet on top
    if row==M-2
A(i,i+rowdim+dimper)=lsm(mas(:,1,i),tpe,bt1(h1_2),h1_2,1,3)
+lsm(mas(:,2,i),tpe,bt2(h1_2),h1_2,3,1);
A(i+rowdim+dimper,i)=lsm(mas(:,1,i),tpe,bt1(h1_2),h1_2,3,1)
+lsm(mas(:,2,i),tpe,bt2(h1_2),h1_2,1,3);
if mod(i-num1,N-2)>1|mod(i-num1,N-2)<1
A(i,i+rowdim+dimper-1)=lsm(mas(:,2,i),tpe,bt2(h1_2),h1_2,3,2)
+lsm(mas(:,3,i),tpe,bt1(h1_2),h1_2,2,3);
    A(i+rowdim+dimper-1,i)=lsm(mas(:,2,i),tpe,bt2(h1_2),h1_2,2,3)
+lsm(mas(:,3,i),tpe,bt1(h1_2),h1_2,3,2);
end
if mod(i-num1,N-2)==1
    A(i,dim2+1)=lsm(mas(:,3,i),tpe,bt1(h1_2),h1_2,2,3);A(dim2+1,i)=
    lsm(mas(:,3,i),tpe,bt1(h1_2),h1_2,3,2);
end
    end
%%%%%

A(i,i)=lsm(mas(:,1,i),tpe,bt1(h1_2),h1_2,1,1)
+lsm(mas(:,3,i),tpe,bt1(h1_2),h1_2,2,2)
+lsm(mas(:,2,i),tpe,bt2(h1_2),h1_2,3,3)

```

```

+lsm(mas(:,4,i),tpe,bt2(h1_1),h1_1,1,1)
+lsm(mas(:,6,i),tpe,bt2(h1_1),h1_1,2,2)
+lsm(mas(:,5,i),tpe,bt1(h1_1),h1_1,3,3);
if tpe==5 hsc(i)=lsm(mas(:,1,i),6,bt1(h1_2),h1_2,1,1)
+lsm(mas(:,3,i),6,bt1(h1_2),h1_2,2,2)
+lsm(mas(:,2,i),6,bt2(h1_2),h1_2,3,3)
+lsm(mas(:,4,i),6,bt2(h1_1),h1_1,1,1)+lsm(mas(:,6,i),6,bt2(h1_1),h1_1,2,2)
+lsm(mas(:,5,i),6,bt1(h1_1),h1_1,3,3);
cc(i)=lsm(mas(:,1,i),7,bt1(h1_2),h1_2,1,1)
+lsm(mas(:,3,i),7,bt1(h1_2),h1_2,2,2)
+lsm(mas(:,2,i),7,bt2(h1_2),h1_2,3,3)
+lsm(mas(:,4,i),7,bt2(h1_1),h1_1,1,1)
+lsm(mas(:,6,i),7,bt2(h1_1),h1_1,2,2)
+lsm(mas(:,5,i),7,bt1(h1_1),h1_1,3,3);
end

    if i<barrier
A(i,i+1)=lsm(mas(:,1,i),tpe,bt1(h1_2),h1_2,1,2)
+lsm(mas(:,6,i),tpe,bt2(h1_1),h1_1,2,1);
        A(i+1,i)=lsm(mas(:,1,i),tpe,bt1(h1_2),h1_2,2,1)
        +lsm(mas(:,6,i),tpe,bt2(h1_1),h1_1,1,2);
    end

    if (rowdim+i)<=num1+num2

        A(i,rowdim+i)=lsm(mas(:,1,i),tpe,bt1(h1_2),h1_2,1,3)
        +lsm(mas(:,2,i),tpe,bt2(h1_2),h1_2,3,1);
A(rowdim+i,i)=lsm(mas(:,2,i),tpe,bt2(h1_2),h1_2,1,3)
+lsm(mas(:,1,i),tpe,bt1(h1_2),h1_2,3,1);
A(i,rowdim+i-1)=lsm(mas(:,2,i),tpe,bt2(h1_2),h1_2,3,2)
+lsm(mas(:,3,i),tpe,bt1(h1_2),h1_2,2,3);

```

```

A(rowdim+i-1,i)=lsm(mas(:,2,i),tpe,bt2(h1_2),h1_2,2,3)
+lsm(mas(:,3,i),tpe,bt1(h1_2),h1_2,3,2);
end
ff(:,i)=[row;h1_1;h1_2;rowdim;barrier];
%periodiskas bazes funkcijas:

if mod(i-num1,N-2)==1
    k=k+1;

if k==1

A(num1+num2+k,dim2+2+k1)=lsm(mas(:,6,i+N-3),tpe,bt2(h1_1),h1_1,1,3);
A(num1+num2+k,dim2+1+k1)=lsm(mas(:,3,i-(N-2)),tpe,bt1(h1_1),h1_1,3,1);
A(dim2+2+k1,num1+num2+k)=lsm(mas(:,6,i+N-3),tpe,bt2(h1_1),h1_1,3,1);
A(dim2+1+k1,num1+num2+k)=lsm(mas(:,3,i-(N-2)),tpe,bt1(h1_1),h1_1,1,3);
end

A(num1+num2+k,num1+num2+k)=
lsm(mas(:,3,i),tpe,bt1(h1_2),h1_2,1,1)
+lsm(mas(:,4,i),tpe,bt2(h1_1),h1_1,2,2)
+lsm(mas(:,3,i-N+2),tpe,bt1(h1_1),h1_1,3,3)+lsm([1;mas(3,1,i+N-3);
mas(2,1,i+N-3)],tpe,bt2(h1_2),h1_2,3,3)
+lsm(mas(:,1,i+N-3),tpe,bt1(h1_2),h1_2,2,2)
+lsm(mas(:,6,i+N-3),tpe,bt2(h1_1),h1_1,1,1);
if tpe==5 hsc(num1+num2+k)=lsm(mas(:,3,i),6,bt1(h1_2),h1_2,1,1)
+lsm(mas(:,4,i),6,bt2(h1_1),h1_1,2,2)
+lsm(mas(:,3,i-N+2),6,bt1(h1_1),h1_1,3,3)+lsm([1;mas(3,1,i+N-3);
mas(2,1,i+N-3)],6,bt2(h1_2),h1_2,3,3)
+lsm(mas(:,1,i+N-3),6,bt1(h1_2),h1_2,2,2)

```

```

+lsm(mas(:,6,i+N-3),6,bt2(h1_1),h1_1,1,1);
cc(num1+num2+k)=lsm(mas(:,3,i),7,bt1(h1_2),h1_2,1,1)
+lsm(mas(:,4,i),7,bt2(h1_1),h1_1,2,2)
+lsm(mas(:,3,i-N+2),7,bt1(h1_1),h1_1,3,3)+lsm([1;mas(3,1,i+N-3);
mas(2,1,i+N-3)],7,bt2(h1_2),h1_2,3,3)
+lsm(mas(:,1,i+N-3),7,bt1(h1_2),h1_2,2,2)
+lsm(mas(:,6,i+N-3),7,bt2(h1_1),h1_1,1,1);
end
A(i,num1+num2+k)=lsm(mas(:,3,i),tpe,bt1(h1_2),h1_2,2,1)
+lsm(mas(:,4,i),tpe,bt2(h1_1),h1_1,1,2);
A(num1+num2+k,i)=lsm(mas(:,3,i),tpe,bt1(h1_2),h1_2,1,2)
+lsm(mas(:,4,i),tpe,bt2(h1_1),h1_1,2,1);
A(i+N-3,num1+num2+k)=lsm(mas(:,1,i+N-3),tpe,bt1(h1_2),h1_2,1,2)
+lsm(mas(:,6,i+N-3),tpe,bt2(h1_1),h1_1,2,1);
A(num1+num2+k,i+N-3)=lsm(mas(:,1,i+N-3),tpe,bt1(h1_2),h1_2,2,1)
+lsm(mas(:,6,i+N-3),tpe,bt2(h1_1),h1_1,1,2);

if row==round((M-2)/2)+1
    A(i+N-3,totdim)=lsm(mas(:,5,i+N-3),tpe,bt1(h1_1),h1_1,3,2)
    +lsm(mas(:,6,i+N-3),tpe,bt2(h1_1),h1_1,2,3);
A(totdim,i+N-3)=lsm(mas(:,5,i+N-3),tpe,bt1(h1_1),h1_1,2,3)
+lsm(mas(:,6,i+N-3),tpe,bt2(h1_1),h1_1,3,2); end

if k>1&k<=floor((M-2)/2)
    A(num1+num2+k,num1+num2+k-1)=lsm(mas(:,3,i),tpe,bt1(h1_1),h1_1,3,1)
    +lsm([1;mas(3,1,i+N-3);mas(2,1,i+N-3)],tpe,bt2(h1_1),h1_1,1,3);
    A(num1+num2+k-1,num1+num2+k)=lsm([1;mas(3,1,i+N-3);
    mas(2,1,i+N-3)],tpe,bt2(h1_1),
    h1_1,3,1)+lsm(mas(:,3,i),tpe,bt1(h1_1),h1_1,1,3);
end
if k==floor((M-2)/2) A(num1+num2+k,dim2+1)
=lsm(mas(:,3,i),tpe,bt1(h1_2),h1_2,1,3);

```

```

        A(num1+num2+k,dim2+2)=lsm([1;mas(3,1,i+N-3);
        mas(2,1,i+N-3)],tpe,bt2(h1_2),h1_2,3,1);

A(dim2+1,num1+num2+k)=lsm(mas(:,3,i),tpe,bt1(h1_2),h1_2,3,1);

A(dim2+2,num1+num2+k)=lsm([1;mas(3,1,i+N-3);
mas(2,1,i+N-3)],tpe,bt2(h1_2),h1_2,1,3);

A(num1+num2+k,dim2)=lsm(mas(:,1,i+N-3),tpe,bt1(h1_2),h1_2,2,3);

A(dim2,num1+num2+k)=lsm(mas(:,1,i+N-3),tpe,bt1(h1_2),h1_2,3,2);
        end

if k<floor((M-2)/2)
    A(i,num1+num2+k+1)=lsm(mas(:,3,i),tpe,bt1(h1_2),h1_2,2,3);
        A(num1+num2+k+1,i)=lsm(mas(:,3,i),tpe,bt1(h1_2),h1_2,3,2);
end

if k>1
    A(i+N-3,num1+num2+k-1)=lsm(mas(:,5,i+N-3),tpe,bt1(h1_1),h1_1,3,2);
        A(num1+num2+k-1,i+N-3)= lsm(mas(:,5,i+N-3),tpe,bt1(h1_1),h1_1,2,3);
end

        ff(:,num1+num2+k)=[row;h1_1;h1_2;rowdim;barrier];
end

end

```

```

%visas pareejaas mijedarbiibas:
A(dim+1,dim2+1)=lsm(mas(:,2,dim-dimper-N+3),tpe,bt2(h1_2),h1_2,1,2);
A(dim2+1,dim+1)
=lsm(mas(:,2,dim-dimper-N+3),tpe,bt2(h1_2),h1_2,2,1);
A(dim2+1,dim2+1)=lsm(mas(:,3,dim-dimper-N+3),
tpe,bt1(h1_2),h1_2,3,3)
+lsm(mas(:,2,dim-dimper-N+3),tpe,bt2(h1_2),h1_2,2,2);
if tpe==5 hsc(dim2+1)=lsm(mas(:,3,dim-dimper-N+3),
6,bt1(h1_2),h1_2,3,3)
+lsm(mas(:,2,dim-dimper-N+3),6,bt2(h1_2),h1_2,2,2);
cc(dim2+1)=lsm(mas(:,3,dim-dimper-N+3),7,
bt1(h1_2),h1_2,3,3)
+lsm(mas(:,2,dim-dimper-N+3),7,bt2(h1_2),h1_2,2,2);
end
A(dim2,dim2+2)=lsm([1;mas(3,1,dim-dimper-N+3)
;mas(2,1,dim-dimper-N+3)],tpe,bt2(h1_2),h1_2,2,1);
A(dim2+2,dim2)
=lsm([1;mas(3,1,dim-dimper-N+3);
mas(2,1,dim-dimper-N+3)],tpe,bt2(h1_2),h1_2,1,2);
A(dim2+2,dim2+2)
=lsm([1;mas(3,1,dim-dimper-N+3);
mas(2,1,dim-dimper-N+3)],tpe,bt2(h1_2),h1_2,1,1);
if tpe==5 hsc(dim2+2)=
lsm([1;mas(3,1,dim-dimper-N+3);mas(2,1,dim-dimper-N+3)],6,
bt2(h1_2),h1_2,1,1);
cc(dim2+2)=lsm([1;mas(3,1,dim-dimper-N+3);
mas(2,1,dim-dimper-N+3)],7,bt2(h1_2),h1_2,1,1);
end

for i=dim+1:dim2
    if i==dim2
        A(i,i)=lsm(mas(:,2,i-dimper-N+2),tpe,bt2(h1_2),h1_2,1,1)

```

```

+lsm([1;1;mas(2,1,i-dimper-N+2)],tpe,bt2(h1_2),h1_2,2,2)
+lsm(mas(:,1,i-dimper-N+2),tpe,bt1(h1_2),h1_2,3,3);
if tpe==5 hsc(i)=lsm(mas(:,2,i-dimper-N+2),6,
bt2(h1_2),h1_2,1,1)
+lsm([1;1;mas(2,1,i-dimper-N+2)],6,bt2(h1_2),h1_2,2,2)
+lsm(mas(:,1,i-dimper-N+2),6,bt1(h1_2),h1_2,3,3);
cc(i)=lsm(mas(:,2,i-dimper-N+2),7,bt2(h1_2),h1_2,1,1)
+lsm([1;1;mas(2,1,i-dimper-N+2)],7,bt2(h1_2),h1_2,2,2)
+lsm(mas(:,1,i-dimper-N+2),7,bt1(h1_2),h1_2,3,3);
end
else
A(i,i)=lsm(mas(:,2,i-dimper-N+2),tpe,bt2(h1_2),h1_2,1,1)
+lsm(mas(:,2,i-dimper-N+3),tpe,bt2(h1_2),h1_2,2,2)
+lsm(mas(:,1,i-dimper-N+2),tpe,bt1(h1_2),h1_2,3,3);
if tpe==5 hsc(i)=lsm(mas(:,2,i-dimper-N+2),6,
bt2(h1_2),h1_2,1,1)
+lsm(mas(:,2,i-dimper-N+3),6,bt2(h1_2),h1_2,2,2)
+lsm(mas(:,1,i-dimper-N+2),6,bt1(h1_2),h1_2,3,3);
cc(i)=lsm(mas(:,2,i-dimper-N+2),7,bt2(h1_2),h1_2,1,1)
+lsm(mas(:,2,i-dimper-N+3),7,bt2(h1_2),h1_2,2,2)
+lsm(mas(:,1,i-dimper-N+2),7,bt1(h1_2),h1_2,3,3);
end
end
if i<dim2 A(i,i+1)=lsm(mas(:,2,i-dimper-N+3),tpe,
bt2(h1_2),h1_2,2,1); A(i+1,i)=lsm(mas(:,2,i-dimper-N+3),tpe,
bt2(h1_2),h1_2,1,2); end
end

for i=dim2+3:dim2+2+2*dim3
if mod((i-dim2-2),2)==1
if i==dim2+1+2*dim3 A(i,i)=lsm(mas(:,1,i-N-dimper),tpe,
bt2(tom(4,i-dim2-2)),tom(4,i-dim2-2),2,2)

```

```

+lsm(mas(:,2,i-N-dimper),tpe,bt1(tom(3,i-dim2-2)),
tom(3,i-dim2-2),1,1);
if tpe==5 hsc(i)=
lsm(mas(:,1,i-N-dimper),6,bt2(tom(4,i-dim2-2)),
tom(4,i-dim2-2),2,2)+lsm(mas(:,2,i-N-dimper),6,bt1(tom(3,i-dim2-2))),
tom(3,i-dim2-2),1,1);
cc(i)=lsm(mas(:,1,i-N-dimper),7,
bt2(tom(4,i-dim2-2)),tom(4,i-dim2-2),2,2)
+lsm(mas(:,2,i-N-dimper),7,bt1(tom(3,i-dim2-2)),
tom(3,i-dim2-2),1,1);
end
else
A(i,i+2)=lsm(mas(:,3,i-N-dimper),tpe,
bt2(tom(3,i-dim2-2)),tom(3,i-dim2-2),3,2);
A(i+2,i)=lsm(mas(:,3,i-N-dimper),tpe,
bt2(tom(3,i-dim2-2)),tom(3,i-dim2-2),2,3);
A(i,i)=lsm(mas(:,1,i-N-dimper),tpe,
bt2(tom(4,i-dim2-2)),tom(4,i-dim2-2),2,2)
+lsm(mas(:,2,i-N-dimper),tpe,bt1(tom(3,i-dim2-2)),
tom(3,i-dim2-2),1,1)+lsm(mas(:,3,i-N-dimper),
tpe,bt2(tom(3,i-dim2-2)),tom(3,i-dim2-2),3,3);
    if tpe==5 hsc(i)=lsm(mas(:,1,i-N-dimper),
6,bt2(tom(4,i-dim2-2)),tom(4,i-dim2-2),2,2)
+lsm(mas(:,2,i-N-dimper),6,bt1(tom(3,i-dim2-2)),
tom(3,i-dim2-2),1,1)+lsm(mas(:,3,i-N-dimper),6,
bt2(tom(3,i-dim2-2)),
tom(3,i-dim2-2),3,3);
cc(i)=lsm(mas(:,1,i-N-dimper),7,bt2(tom(4,i-dim2-2)),
tom(4,i-dim2-2),2,2)+lsm(mas(:,2,i-N-dimper),7,
bt1(tom(3,i-dim2-2)),
tom(3,i-dim2-2),1,1)
+lsm(mas(:,3,i-N-dimper),7,bt2(tom(3,i-dim2-2)),

```

```

        tom(3,i-dim2-2),3,3);
        end
        end
    end
if mod((i-dim2-2),2)==0
    if i==dim2+2+2*dim3 A(i,i)=lsm(mas(:,1,i-N-dimper),
    tpe,bt3(tom(4,i-dim2-2)),
    tom(4,i-dim2-2),2,2)+lsm(mas(:,2,i-N-dimper),
    tpe,bt1(tom(3,i-dim2-2)),tom(3,i-dim2-2),1,1);
    if tpe==5 hsc(i)=lsm(mas(:,1,i-N-dimper),6,
    bt3(tom(4,i-dim2-2)),tom(4,i-dim2-2),2,2)+
    lsm(mas(:,2,i-N-dimper),6,bt1(tom(3,i-dim2-2)),
    tom(3,i-dim2-2),1,1);
    cc(i)=lsm(mas(:,1,i-N-dimper),7,bt3(tom(4,i-dim2-2)),
    tom(4,i-dim2-2),2,2)+lsm(mas(:,2,i-N-dimper),7,
    bt1(tom(3,i-dim2-2)),tom(3,i-dim2-2),1,1);
    end
else
    A(i,i+2)=lsm(mas(:,4,i-N-dimper),tpe,
    bt3(tom(3,i-dim2-2)),tom(3,i-dim2-2),3,2);
    A(i+2,i)=lsm(mas(:,4,i-N-dimper),tpe,
    bt3(tom(3,i-dim2-2)),tom(3,i-dim2-2),2,3);

A(i,i)=lsm(mas(:,1,i-N-dimper),tpe,bt3(tom(4,i-dim2-2)),
tom(4,i-dim2-2),2,2)+lsm(mas(:,2,i-N-dimper),tpe,bt1(tom(3,i-dim2-2)),
tom(3,i-dim2-2),2,2)+lsm(mas(:,3,i-N-dimper),tpe,bt2(tom(3,i-dim2-2)),
tom(3,i-dim2-2),3,3)+lsm(mas(:,4,i-N-dimper),tpe,bt3(tom(3,i-dim2-2)),
tom(3,i-dim2-2),3,3);
    if tpe==5 hsc(i)=lsm(mas(:,1,i-N-dimper),6,bt3(tom(4,i-dim2-2)),
    tom(4,i-dim2-2),2,2)+lsm(mas(:,2,i-N-dimper),6,bt1(tom(3,i-dim2-2)),
    tom(3,i-dim2-2),2,2)
    +lsm(mas(:,3,i-N-dimper),6,bt2(tom(3,i-dim2-2)),tom(3,i-dim2-2),3,3)

```

```

+lsm(mas(:,4,i-N-dimper),6,bt3(tom(3,i-dim2-2)),tom(3,i-dim2-2),3,3);
cc(i)=lsm(mas(:,1,i-N-dimper),7,bt3(tom(4,i-dim2-2)),
tom(4,i-dim2-2),2,2)+lsm(mas(:,2,i-N-dimper),7,
bt1(tom(3,i-dim2-2)),
tom(3,i-dim2-2),2,2)
+lsm(mas(:,3,i-N-dimper),7,bt2(tom(3,i-dim2-2))
,tom(3,i-dim2-2),3,3)+lsm(mas(:,4,i-N-dimper),7,
bt3(tom(3,i-dim2-2)),
tom(3,i-dim2-2),3,3);
end
end
end
end

end
%for pekle
for i=totdim+1:totdim+N1
    if i==totdim+1
A(i,i)=lsm(mas(:,4,i-totdim),tpe,bt2(tom(4,1)),tom(4,1),3,3)
+lsm(mas(:,5,i-totdim),tpe,bt1(tom(4,1)),tom(4,1),1,1);
    if tpe==5 hsc(i)=lsm(mas(:,4,i-totdim),6,bt2(tom(4,1)),
tom(4,1),3,3)
+lsm(mas(:,5,i-totdim),6,bt1(tom(4,1)),tom(4,1),1,1);
        cc(i)=lsm(mas(:,4,i-totdim),7,bt2(tom(4,1)),
tom(4,1),3,3)
+lsm(mas(:,5,i-totdim),7,bt1(tom(4,1)),
tom(4,1),1,1);end
    else A(i,i)=lsm(mas(:,4,i-totdim),tpe,
bt2(tom(4,1)),tom(4,1),3,3)
+lsm(mas(:,5,i-totdim),tpe,bt1(tom(4,1)),
tom(4,1),1,1)
+lsm(mas(:,5,i-totdim-1),tpe,bt1(tom(4,1)),
tom(4,1),2,2);

```

```

    if tpe==5 hsc(i)=lsm(mas(:,4,i-totdim),6,
    bt2(tom(4,1)),tom(4,1),3,3)
    +lsm(mas(:,5,i-totdim),6,bt1(tom(4,1)),
    tom(4,1),1,1)
    +lsm(mas(:,5,i-totdim-1),6,bt1(tom(4,1)),
    tom(4,1),2,2);
    cc(i)=lsm(mas(:,4,i-totdim),7,bt2(tom(4,1)),
    tom(4,1),3,3)
    +lsm(mas(:,5,i-totdim),7,bt1(tom(4,1)),tom(4,1),1,1)
    +lsm(mas(:,5,i-totdim-1),7,bt1(tom(4,1)),
    tom(4,1),2,2);end
end
if i==totdim+N1
    A(i,i)=lsm(mas(:,4,i-totdim),tpe,bt2(tom(4,1)),tom(4,1),3,3)
    +lsm(mas(:,5,i-totdim),tpe,bt3(tom(4,1)),tom(4,1),3,3);
    if tpe==5 hsc(i)=lsm(mas(:,4,i-totdim),6,bt2(tom(4,1)),
    tom(4,1),3,3)+lsm(mas(:,5,i-totdim),6,bt3(tom(4,1)),
    tom(4,1),3,3);
    cc(i)=lsm(mas(:,4,i-totdim),7,bt2(tom(4,1)),
    tom(4,1),3,3)
    +lsm(mas(:,5,i-totdim),7,bt3(tom(4,1)),
    tom(4,1),3,3);end
end
if i<totdim+N1

    A(i,i+1)=lsm(mas(:,5,i-totdim),tpe,bt1(tom(4,1)),tom(4,1),1,2)
    A(i+1,i)=
    lsm(mas(:,5,i-totdim),tpe,bt1(tom(4,1)),tom(4,1),2,1);
end
A(i,i-totdim)=lsm(mas(:,4,i-totdim),tpe,bt2(tom(4,1)),
tom(4,1),3,1)+lsm(mas(:,5,i-totdim),tpe,bt1(tom(4,1)),
tom(4,1),1,3);

```

```

A(i-totdim,i)=lsm(mas(:,4,i-totdim),tpe,bt2(tom(4,1)),
tom(4,1),1,3)+lsm(mas(:,5,i-totdim),tpe,bt1(tom(4,1)),
tom(4,1),3,1);
    if i>totdim+1
        A(i,i-totdim-1)=lsm(mas(:,6,i-totdim-1),
            tpe,bt2(tom(4,1)),
            tom(4,1),3,2);
A(i-totdim-1,i)=lsm(mas(:,6,i-totdim-1),tpe,
bt2(tom(4,1)),
tom(4,1),2,3);
end

    if i==totdim+N1
        A(i,i-totdim)=lsm(mas(:,4,i-totdim),tpe,bt2(tom(4,1))
            ,tom(4,1),3,1)+lsm(mas(:,5,i-totdim),tpe,bt3(tom(4,1)),
            tom(4,1),3,1);
A(i-totdim,i)=lsm(mas(:,4,i-totdim),tpe,bt2(tom(4,1)),
tom(4,1),1,3)+lsm(mas(:,5,i-totdim),tpe,bt3(tom(4,1)),
tom(4,1),1,3);
end

end

A(totdim+1,dim2+3)=lsm(mas(:,4,1),tpe,bt2(tom(4,1)),
tom(4,1),3,2);A(dim2+3,totdim+1)
=lsm(mas(:,4,1),tpe,bt2(tom(4,1)),tom(4,1),2,3);
A(totdim+N1,dim2+4)=lsm(mas(:,5,N1),tpe,bt3(tom(4,1)),
tom(4,1),3,2);A(dim2+4,totdim+N1)
=lsm(mas(:,5,N1),tpe,bt3(tom(4,1)),tom(4,1),2,3);
%

end

```

```

A=zeros(dim2+2+2*dim3,dim2+2+2*dim3);J=A;C=A;D=A;

%galvenas iteracijas:

B=(stiffness(3))';

    C=(stiffness(3))';

D=(stiffness(4))';

A=(stiffness(1))';

f=zeros(num1+num2+floor((M-2)/2),1);

dor=zeros(2*dim3,1);

counter=0;
bmain=0.01;

sigma=1;

sol=zeros(num1+num2+floor((M-2)/2),1);sol2=zeros(dim+2*dim3,1);sol2=sol;
tom2=[[zeros(4,1),tom];zeros(1,size(tom,2)+1)]; ab=[];or=[];no1=1;

```

```

%sol2=ones(dim+2*dim3,1);
    %GALVENO ITERAACIJU SAAKUMS
    while counter<40&no1>0.00001

        ssol=[sol,sol2(1:dim)];

sol=linsolve(A(1:dim,1:dim),f-A(1:dim,dim+1:dim2+2)*ones(N,1));

%
    val=ff(1,dim-dimper);
    for i=dim-dimper+1:dim
        ff(1,i)=val;
    end

mas=masv(dim,dim2,dimper,N1,M,ff,sol,num1,N);

mas;

J=(stiffness(5))';

```

```

%toma nosciijumi
no2=norm(dor-dor1(tom2,sigma,sol,N1,h,Y,num1,num2,L))
dor=dor1(tom2,sigma,sol,N1,h,Y,num1,num2,L);
%

As2=A;
Js2=J;Cs2=C;Cmin=C;
As2(dim+1:dim2+2,:)=[];
As2(:,dim+1:dim2+2)=[];
Js2(dim+1:dim2+2,:)=[];Js2(:,dim+1:dim2+2)=[];
Cs2(dim+1:dim2+2,:)=[];Cs2(:,dim+1:dim2+2)=[];
Cmin(dim+1:dim2+2,:)=[];Cmin(:,1:dim)=[];
Cmin(:,size(Cmin,2)-2*dim3+1:size(Cmin,2))=[];

sol2=bmain*linsolve(A(1:dim,1:dim)/Re-J(1:dim,1:dim),-
S*(C(1:dim,1:dim)*sol
+C(1:dim,dim+1:dim2+2)*ones(N,1))-A(1:dim,dim2+3:dim2+2+
2*dim3)*dor/Re
+J(1:dim,dim2+3:dim2+2+2*dim3)*dor)+(1-bmain)*sol2;

f=D(1:dim,1:dim)*sol2(1:dim)+D(1:dim,dim2+3:dim2+2+2*dim3)*dor;

```

```

counter=counter+1;

ssol;
    ssol=[ssol,sol,sol2(1:dim)];
    %ssol=[ssol,sol,sol2];

    if mod(counter,10)==0
        norm(ssol(:,size(ssol,2)-1)-ssol(:,size(ssol,2)-3))
    no1=norm(ssol(:,size(ssol,2))-ssol(:,size(ssol,2)-2))
    end

    ab=[ab,0.01*counter];or=[or,no1];

end

```

```

%GALVENO ITERAACIJU BEIGAS

```

```

    norm(ssol(:,size(ssol,2)-1)-ssol(:,size(ssol,2)-3))
    norm(ssol(:,size(ssol,2))-ssol(:,size(ssol,2)-2))

    %Z konstrukcija:

function [Z,Z2]=mgrid(sol,sol2)

    Z=zeros(M,N); Z2=zeros(M,N);Z3=Z2;
    r=round((M-2)/2);
    k=0; barrier=N2; tomk=2;

```

```

for i=2:r

    Z(i,:)= [zeros(1,N1+1-i),fliplr(sol(k+1:barrier)') ,
    zeros(1,N1+1-i)];
    Z2(i,:)= [zeros(1,N1-i),dor(tomk),
    fliplr(sol2(k+1:barrier)') ,dor(tomk-1),zeros(1,N1-i)];

    k=barrier;
    barrier=barrier+2*(i-1)+N2;
    tomk=tomk+2;

end

Z(r+1:M-1,:)= [zeros(M-1-r,1),
    fliplr(vec2mat(sol(num1+3-N:num1+num2),N-2)),zeros(M-1-r,1)];
Z2(r+1:M-1,:)= [zeros(M-1-r,1),
    fliplr(vec2mat(sol2(num1+3-N:num1+num2),N-2)),zeros(M-1-r,1)];

Z2(N1,1)=dor(tomk); Z2(N1,N)=dor(tomk-1);

%for periodic:
Z(round((M-2)/2)+2:M-1,1)=sol(num1+num2+1:num1+num2
+floor((M-2)/2));
Z(round((M-2)/2)+2:M-1,N)=Z(round((M-2)/2)+2:M-1,1);

Z2(round((M-2)/2)+2:M-1,1)=
sol2(num1+num2+1:num1+num2+floor((M-2)/2));
Z2(round((M-2)/2)+2:M-1,N)=
Z2(round((M-2)/2)+2:M-1,1);

```

```

        %for non-hom Dirichlet:
        Z(M,:)=ones(1,N);

    end
[Z,Z2]=mgrid(sol,sol2);

%aatrums

[px,py] = gradient(Z,10,10);
sz = sqrt(px.^2 + py.^2);
px = px./sz;
py = py./sz;

Z
Z2

figure
quiver(X,Y,py,px,.5)

figure
Q=contour(X,Y,Z,w1,'LineWidth',1);
clabel(Q)
title('Psi(x,y) limenlinijas')
xlabel('x'),ylabel('y')

%figure, contour(X,Y,Z)
%title('Psi(x,y) limenlinijas')

```

```

xlabel('x'),ylabel('y')

%figure, contour(X,Y,Z2)
%title('Dzeta(x,y) limenlinijas')
xlabel('x'),ylabel('y')
figure, surf(X,Y,Z)

figure
QQ=contour(X,Y,Z2,[0 -0.3 -0.5 -0.7 -1],'LineWidth',1);
clabel(QQ)
title('Dzeta(x,y) limenlinijas')
xlabel('x'),ylabel('y')

%periodiskums un simetrija
XXX=[X,X(:,2:size(X,2))+3*ones(size(X,1),size(X,2)-1)];
YYY=[Y,Y(:,2:size(X,2))];
ZZZ=[Z,Z(:,2:size(X,2))];

%figure, contour(XXX,YYY,ZZZ)
%title('Psi(x,y) limenlinijas')
xlabel('x'),ylabel('y')
XXX;
YYY;

for i=0:size(XXX,1)-2
    YY2(i+1,:)= -YYY(size(XXX,1)-i,:);
    ZZ2(i+1,:)= ZZZ(size(XXX,1)-i,:);
end
YYY2=[YY2;YYY];

```

```

ZZZ2=[ZZ2;ZZZ];
XXX2=[XXX;XXX(1:size(XXX)-1,:)];

%figure, contour(XXX2,YYY2,ZZZ2)
%title('Psi(x,y) limenlinijas')
%xlabel('x'),ylabel('y')

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
paliigfunkcijas
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function dor=dor1(tom2,sigma,sol,N1,h,Y,num1,num2,L)

beta=sigma;

%koordinates (Dorodnicina iteracijaam)
kk=2;cord=[0;0];bbb=N1;
while kk<=(2*N1-2)/2+1
    cord=[cord,[(N1-kk)*h, (N1-kk)*h+(bbb+1)*h;Y(kk,1), Y(kk,1)]];
    kk=kk+1;bbb=bbb+2;
end

    for i=2:2*(N1-1)+1

%toma
%
tom2(5,i)=(1-beta)*tom2(5,i)-2*beta*(h^(-2)*sol(tom2(1,i))
+tom2(3,i)^(-2)*sol(tom2(1,i)+tom2(2,i)));
        if i==size(tom2,2)|i==size(tom2,2)-1
            %tom2(5,i)=(1-beta)*tom2(5,i)-2*beta*(h^(-2)*

```

```

        sol(tom2(1,i))
        +tom2(3,i)^(-2)*sol(num1+num2+1));
tom2(5,i)=(1-beta)*tom2(5,i)-2*beta*(tom2(3,i)^(-2)*sol(num1+num2+1));
end

```

```

dor=tom2(size(tom2,1),2:size(tom2,2))';

```

```

end

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

function mas=masv(dim,dim2,dimper,N1,M,ff,sol,num1,N);

```

```

k=0;

```

```

for i=1:dim-dimper

```

```

    if ff(1,i)==1

```

```

        if i==1

```

```

            k=k+1;

```

```

            mas(:, :, dim-dimper+k)=[[sol(i);0;0],

```

```

            [0;sol(i);sol(i+ff(4,i))],

```

```

            [sol(i+ff(4,i));0;0],

```

```

            [0;0;0],[0;0;0],[0;0;0]];

```

```

            mas(:, :, i)=[[sol(i);sol(i+1);

```

```

            sol(i+ff(4,i)+1)], [sol(i+ff(4,i)+1);

```

```

            sol(i+ff(4,i));sol(i)], [0;sol(i);

```

```

            sol(i+ff(4,i))], [sol(i);0;0],[0;0;sol(i)],

```

```

            [sol(i+1);sol(i);0]];end

```

```

    if i==ff(5,i)

```

```

        k=k+1;

```

```

        mas(:, :, dim-dimper+k)=[[sol(i);0;0],

```

```

        [sol(i);0;sol(i+ff(4,i)+1)],

```

```

        [sol(i+ff(4,i)+2);sol(i+ff(4,i)+1);0],

```

```

        [sol(i+ff(4,i)+2);0;0],[0;0;0],[0;0;0]];

```

```

mas(:, :, i) = [[sol(i); 0; sol(i+ff(4, i)+1)],
[sol(i+ff(4, i)+1); sol(i+ff(4, i)); sol(i)],
[sol(i-1); sol(i); sol(i+ff(4, i))], [sol(i); sol(i-1); 0],
[sol(i); 0; 0], [0; 0; 0]]; end
if i > 1 & i < ff(5, i)
mas(:, :, i) = [[sol(i); sol(i+1);
sol(i+ff(4, i)+1)], [sol(i+ff(4, i)+1);
sol(i+ff(4, i)); sol(i)],
[sol(i-1); sol(i); sol(i+ff(4, i))],
[sol(i); sol(i-1); 0], [0; 0; sol(i)],
[sol(i+1); sol(i); 0]]; end
end

if ff(1, i) > 1 & ff(1, i) < N1 - 1

if (ff(1, i) - ff(1, i-1)) > 0
k = k + 1;
mas(:, :, dim-dimper+k) =
[[sol(i); 0; 0], [0; sol(i); sol(i+ff(4, i))],
[sol(i+ff(4, i)); 0; 0], [0; 0; 0], [0; 0; 0], [0; 0; 0]];

mas(:, :, i) = [[sol(i); sol(i+1);
sol(i+ff(4, i)+1)], [sol(i+ff(4, i)+1); sol(i+ff(4, i)); sol(i)],
[0; sol(i); sol(i+ff(4, i))], [sol(i); 0; 0],
[0; sol(i-ff(4, i)+2); sol(i)],
[sol(i+1); sol(i); sol(i-ff(4, i)+2)]]; end
if (ff(1, i+1) - ff(1, i)) > 0
k = k + 1;
mas(:, :, dim-dimper+k) =
[[sol(i); 0; 0],
[sol(i); 0; sol(i+ff(4, i)+1)], [sol(i+ff(4, i)+2);
sol(i+ff(4, i)+1); 0],

```

```

        [sol(i+ff(4,i)+2);0;0],[0;0;0],[0;0;0]];
mas(:, :, i)=[[sol(i);0;sol(i+ff(4,i)+1)],
[sol(i+ff(4,i)+1);sol(i+ff(4,i));sol(i)],
[sol(i-1);sol(i);sol(i+ff(4,i))],
[sol(i);sol(i-1);0],[sol(i);0;0],[0;0;0]];end
        if (ff(1,i)-ff(1,i-1))==0&
            (ff(1,i+1)-ff(1,i))==0
mas(:, :, i)=[[sol(i);sol(i+1);
sol(i+ff(4,i)+1)],
[sol(i+ff(4,i)+1);
sol(i+ff(4,i));sol(i)],
[sol(i-1);sol(i);
sol(i+ff(4,i))],
[sol(i);sol(i-1);
sol(i-ff(4,i)+1)],
[sol(i-ff(4,i)+1);
sol(i-ff(4,i)+2);sol(i)],
[sol(i+1);sol(i);
sol(i-ff(4,i)+2)]];
if (ff(1,i+2)-ff(1,i))>0
    mas(:, :, i)=[[sol(i);sol(i+1);sol(i+ff(4,i)+1)],
[sol(i+ff(4,i)+1);sol(i+ff(4,i));sol(i)],
[sol(i-1);sol(i);sol(i+ff(4,i))],
s[sol(i);sol(i-1);sol(i-ff(4,i)+1)],
[sol(i-ff(4,i)+1);0;sol(i)],
[sol(i+1);sol(i);0]];end
        end
    end

if ff(1,i)==N1-1
    if (ff(1,i)-ff(1,i-1))==0&(ff(1,i+1)-ff(1,i))==0
        mas(:, :, i)=[[sol(i);sol(i+1);

```

```

        sol(i+ff(4,i)), [sol(i+ff(4,i));
        sol(i+ff(4,i)-1);sol(i)],
        [sol(i-1);sol(i);sol(i+ff(4,i)-1)],
        [sol(i);sol(i-1);sol(i-ff(4,i)+1)],
        [sol(i-ff(4,i)+1);sol(i-ff(4,i)+2);sol(i)],
        [sol(i+1);sol(i);sol(i-ff(4,i)+2)]];
    end
    if (ff(1,i)-ff(1,i-1))>0
        k=k+1;
        mas(:, :, dim-dimper+k)=[[sol(i);0;0],
        [0;sol(i);sol(dim-dimper+1)], [0;0;0], [0;0;0], [0;0;0], [0;0;0]];
    mas(:, :, i)=[[sol(i);sol(i+1);sol(i+ff(4,i))],
    [sol(i+ff(4,i));sol(dim-dimper+1);sol(i)],
    [0;sol(i);sol(dim-dimper+1)], [sol(i);0;0],
    [0;sol(i-ff(4,i)+2);sol(i)], [sol(i+1);sol(i);sol(i-ff(4,i)+2)]];    end
        if (ff(1,i+1)-ff(1,i))>0
            k=k+1;
            mas(:, :, dim-dimper+k)=[[sol(i);0;0],
            [sol(i);0;sol(i+ff(4,i))],
            [sol(dim-dimper+1);sol(i+ff(4,i));0], [0;0;0], [0;0;0], [0;0;0]];
        mas(:, :, i)=[[sol(i);0;sol(i+ff(4,i))],
        [sol(i+ff(4,i));sol(i+ff(4,i)-1);sol(i)],
        [sol(i-1);sol(i);sol(i+ff(4,i)-1)], [sol(i);sol(i-1);0],
        [sol(i);0;0], [0;0;0]];
        mas(:, :, i-1)=[[sol(i-1);sol(i);
        sol(i-1+ff(4,i-1))], [sol(i-1+ff(4,i-1));
        sol(i-1+ff(4,i-1)-1);sol(i-1)],
        [sol(i-2);sol(i-1);sol(i-1+ff(4,i-1)-1)],
        [sol(i-1);sol(i-2);sol(i-1-ff(4,i-1)+1)],
        [sol(i-1-ff(4,i-1)+1);0;sol(i-1)],
        [sol(i);sol(i-1);0]];
    end

```

```

end

if ff(1,i)>N1&ff(1,i)<M-2
    mas(:, :, i)=[[sol(i);sol(i+1);
    sol(i+ff(4,i))],[sol(i+ff(4,i));
    sol(i+ff(4,i)-1);sol(i)],
    [sol(i-1);sol(i);sol(i+ff(4,i)-1)],
    [sol(i);sol(i-1);sol(i-ff(4,i))],
    [sol(i-ff(4,i));sol(i-ff(4,i)+1);sol(i)],
    [sol(i+1);sol(i);sol(i-ff(4,i)+1)]];
    if (ff(1,i)-ff(1,i-1))>0
mas(:, :, i)=[[sol(i);sol(i+1);
sol(i+ff(4,i))],
[sol(i+ff(4,i));sol(dim-dimper-N1+2+ff(1,i));
sol(i)],[sol(dim-dimper-N1+1+ff(1,i));
sol(i);sol(dim-dimper-N1+2+ff(1,i))],
[sol(i);sol(dim-dimper-N1+1+ff(1,i));
sol(i-ff(4,i))],[sol(i-ff(4,i));
sol(i-ff(4,i)+1);sol(i)],
[sol(i+1);sol(i);sol(i-ff(4,i)+1)]];
        end
        if (ff(1,i+1)-ff(1,i))>0
mas(:, :, i)=[[sol(i);sol(dim-dimper-N1+1+ff(1,i));
sol(i+ff(4,i))],[sol(i+ff(4,i));
sol(i+ff(4,i)-1);sol(i)],[sol(i-1);sol(i);
sol(i+ff(4,i)-1)],
[sol(i);sol(i-1);sol(i-ff(4,i))],
[sol(i-ff(4,i));
sol(dim-dimper-N1+ff(1,i));sol(i)],
[sol(dim-dimper-N1+1+ff(1,i));sol(i);

```

```

sol(dim-dimper-N1+ff(1,i))]];
    end

end

if ff(1,i)==M-2
    mas(:, :, i)=[[sol(i);sol(i+1);1],[1;1;sol(i)],
[sol(i-1);sol(i);1],[sol(i);sol(i-1);sol(i-ff(4,i))],
[sol(i-ff(4,i));sol(i-ff(4,i)+1);sol(i)],
[sol(i+1);sol(i);sol(i-ff(4,i)+1)]];
        if (ff(1,i)-ff(1,i-1))>0
mas(:, :, i)=[[sol(i);sol(i+1);1],[1;1;sol(i)],
[sol(dim-dimper-N1+1+ff(1,i));sol(i);1],
[sol(i);sol(dim-dimper-N1+1+ff(1,i));sol(i-ff(4,i))],
[sol(i-ff(4,i));sol(i-ff(4,i)+1);sol(i)],
[sol(i+1);sol(i);sol(i-ff(4,i)+1)]];    end
        if i==dim-dimper

mas(:, :, i)=[[sol(i);sol(dim-dimper-N1+1+ff(1,i));1],
[1;1;sol(i)], [sol(i-1);sol(i);1],
[sol(i);sol(i-1);sol(i-ff(4,i))],
[sol(i-ff(4,i))];
sol(dim-dimper-N1+ff(1,i));sol(i)],
[sol(dim-dimper-N1+1+ff(1,i));
sol(i);sol(dim-dimper-N1+ff(1,i))]];
        end

end

```

```

if ff(1,i)==N1&N1<M-2
    mas(:, :, i)=[[sol(i);sol(i+1);
    sol(i+ff(4,i))],[sol(i+ff(4,i));
    sol(i+ff(4,i)-1);sol(i)],
    [sol(i-1);sol(i);sol(i+ff(4,i)-1)],[sol(i);sol(i-1);
    sol(i-ff(4,i))],[sol(i-ff(4,i));
    sol(i-ff(4,i)+1);sol(i)],[sol(i+1);sol(i);
    sol(i-ff(4,i)+1)]];
    if (ff(1,i)-ff(1,i-1))>0
mas(:, :, i)=[[sol(i);sol(i+1);sol(i+ff(4,i))],
[sol(i+ff(4,i));sol(dim-dimper-N1+2+ff(1,i));sol(i)],
[sol(dim-dimper-N1+1+ff(1,i));
sol(i);sol(dim-dimper-N1+2+ff(1,i))],[sol(i);
sol(dim-dimper-N1+1+ff(1,i));
sol(i-ff(4,i))],
[sol(i-ff(4,i));sol(i-ff(4,i)+1);sol(i)],[sol(i+1);sol(i);
sol(i-ff(4,i)+1)]];end
    if i==num1+N-2
mas(:, :, i)=[[sol(i);sol(dim-dimper-N1+1+ff(1,i));
sol(i+ff(4,i))],[sol(i+ff(4,i));sol(i+ff(4,i)-1);sol(i)],
[sol(i-1);sol(i);
sol(i+ff(4,i)-1)],[sol(i);sol(i-1);sol(i-ff(4,i))],
[sol(i-ff(4,i));0;sol(i)],
[sol(dim-dimper-N1+1+ff(1,i));sol(i);0]];end

end

end

end

```

Maģistra darbs "Cilindru virknes aptecēšanas matemātiskais modelis" izstrādāts LU Fizikas un Matemātikas fakultātē.

Ar savu parakstu apliecinu, ka pētījums veikts patstāvīgi, izmantoti tikai tajā norādītie informācijas avoti un iesniegtā darba elektroniskā kopija atbilst izdrukai.

Autors: Maksims Marinaki

(paraksts)

(datums)

Rekomendēju darbu aizstāvēšanai.

Vadītājs: prof. Dr.math. Harijs Kalis

(paraksts)

(datums)

Recenzents: asoc.prof. Dr.math. Jānis Cepītis

(paraksts)

(datums)

Darbs iesniegts Matemātikas nodaļā _____

(datums)

(darbu pieņēma)

Darbs aizstāvēts maģistra gala pārbaudījuma komisijas sēdē

_____ prot. Nr. _____, vērtējums _____

(datums)

Komisijas sekretārs/-e: _____

(Vārds, Uzvārds)

(paraksts)