



**MATEMATICKO-FYZIKÁLNÍ
FAKULTA**
Univerzita Karlova

DIPLOMOVÁ PRÁCE

Vojtěch Příhoda

Numerická simulace problémů magnetismu

Katedra numerické matematiky

Vedoucí diplomové práce: prof. RNDr. Vít Dolejší, Ph.D., DSc.

Studijní program: Fyzika

Studijní obor: Matematické a počítačové modelování ve fyzice a technice

Praha 2018

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně a výhradně s použitím citovaných pramenů, literatury a dalších odborných zdrojů.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona v platném znění, zejména skutečnost, že Univerzita Karlova má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle §60 odst. 1 autorského zákona.

V dne

Podpis autora

Rád bych poděkoval rodině, přátelům a své přítelkyni, kteří mě trpělivě podporovali během psaní této práce. Také bych rád poděkoval svému vedoucímu, učitelům a spolužákům na MFF UK a všem ostatním, kteří přispěli k mému studiu.

Název práce: Numerická simulace problémů magnetismu

Autor: Vojtěch Příhoda

Katedra: Katedra numerické matematiky

Vedoucí diplomové práce:

prof. RNDr. Vít Dolejší, Ph.D., DSc., Katedra numerické matematiky

Abstrakt: Tato práce se zabývá řešením stacionárního magnetického pole v materiálu s nekonstantí reluktivitou v realistické geometrii. K tomuto účelu jsme rozšířili software *adgfem*. Software *adgfem* implementuje nespojitou Galerkinovu metodu a dosud byl využívám především pro rovnice typu konvekce-difuze a postrádal ucelený přístup k vytváření sítě. Tato práce přináší podrobný návod, jak tvořit komplexní geometrie s využitím softwaru *SALOME*. Ty jsou pak pomocí nově napsaného konvertoru *datToAdgfem* převedeny do formátu čitelného pro *adgfem*. Na takto vytvořené síti je pak proveden výpočet nelineárního statického magnetického pole.

Klíčová slova: *adgfem* DGM *SALOME* magnetismus

Title: Numerical simulation of problems of magnetism

Author: Vojtěch Příhoda

Department: Department of Numerical Mathematics

Supervisor:

prof. RNDr. Vít Dolejší, Ph.D., DSc., Department of Numerical Mathematics

Abstract: This thesis deals with solving stationary magnetic field in material with non-constant magnetic reluctivity in realistic geometry. To this end, we extended *adgfem* software. Software *adgfem* implements discontinuous Galerkin method and so far has been used mainly to solve convection-diffusion problems and lacked streamlined approach to computational mesh generation. This thesis contains step-by-step guideline to creation of complex geometry using software *SALOME*. This mesh is then converted to format suitable for *adgfem* using newly written convertor *datToAdgfem*. Mesh created in this way is then used for calculation of non-linear static magnetic field.

Keywords: *adgfem* DGM *SALOME* magnetism

Obsah

Úvod	2
1 Magnetické pole ve 2D	3
1.1 Zadání	3
1.2 Magnetický potenciál	3
2 Řešitelnost úlohy	5
2.1 Teorie monotóních operátorů	5
2.2 Slabá formulace	6
3 Diskretizace	9
3.1 Triangulace	9
3.2 Funkční prostory	9
3.3 Numerické řešení	10
3.4 Soustava algebraických rovnic	16
4 Tvorba sítě	18
4.1 Definice geometrie	18
4.2 SALOME	18
4.3 datToAdgfem	21
4.4 Vizualizace výsledné sítě	28
4.5 Omezení	28
5 Výpočty v adgfem	32
5.1 Magnetické pole	32
5.2 Nastavení výpočtů	32
Závěr	37
Seznam použité literatury	38
Seznam obrázků	39
Přílohy	40

Úvod

Původní myšlenkou práce bylo řešit úlohy magnetohydrodynamiky. K tomuto účelu jsme chtěli použít program *adgfem*. Program *adgfem* implementuje nespojitou Galerkinovu metodu, je vyvíjen na Katedře numerické matematiky MFF UK pod vedením prof. Víta Dolejšího. Je napsán v jazyce FORTRAN. Více o nespojité Galerkinově metodě např. v (Dolejší a M., 2015).

Už dnes je *adgfem* schopný počítat jak rovnice proudění, tak i Poissonovu rovnici, na kterou lze stacionární magnetické pole převést. Nicméně, místo snahy rovnou implementovat rovnice magnetohydrodynamiky byla jako první krok vybrána úloha stacionárního nelineárního magnetického pole. Pro tuto úlohu jsme vybrali realistickou geometrii alternátoru. Ukázalo se však, že *adgfem* není vybaven žádným robustním síťovacím nástrojem, ve kterém by bylo možné požadovanou geometrii osítovat. Navíc, z takového nástroje by mohli profitovat i další uživatelé *adgfemu*. Tím se těžiště práce přesunulo z magnetického pole na síťování.

Ukázalo se, že nejlepší cestou je použít již existující nástroje, konkrétně *WebPlotDigitizer* a *SALOME*. Kromě toho bylo třeba tyto nástroje nějak spojit. Výstup z *WebPlotDigitizeru* se podařilo dostat do *SALOME* pomocí jednoduchého skriptu v Pythonu. Pro převedení výstupu ze *SALOME* do *adgfem* bylo třeba napsat konvertor *datToAdgfem*, protože nešlo o pouhé přeskládání čísel v souboru, nýbrž bylo zajistit některé vlastnosti, které *adgfem* od sítě očekává, jako např. správně orientované trojúhelníky a hrany na okrajích sítě. Kromě toho se *datToAdgfem* umí vypořádat i s vnitřními hranicemi, tj. hranami, které oddělují např. jednotlivé materiály a musí být zachovány i po tom, co *adgfem* síť adaptuje.

Tím ovšem původní záměr práce ustoupil do pozadí. Z úlohy na výpočet magnetického pole se stal spíš prostředek, jak ověřit, že *datToAdgfem* je skutečně funkční. Na implementaci magnetohydrodynamiky se z časových důvodů už bohužel nedostalo.

1. Magnetické pole ve 2D

1.1 Zadání

Pro tuto práci byla vybrána úloha č. 4 v (Křížek a Segeth, 2001). Bud $\Omega \subset \mathbb{R}^2$ oblast, zabýváme se tedy řezem nějakým magnetickým polem, typicky nějakého elektromagnetického stroje. Systém Maxwellových rovnic popisující stacionární magnetické pole je

$$\operatorname{rot} H = f \quad \text{v } \Omega \quad (1.1)$$

$$\operatorname{div} B = 0 \quad \text{v } \Omega, \quad (1.2)$$

kde H je magnetická intenzita, B je magnetická indukce a f je proudová hustota (resp. její složka kolmá na rovinu, kterou uvažujeme). Vzhledem k tomu, že pracujeme v \mathbb{R}^2 , budeme chápat, že

$$\operatorname{rot} H = \frac{\partial H_2}{\partial x_1} - \frac{\partial H_1}{\partial x_2}.$$

Vztah mezi H a B budeme uvažovat ve tvaru

$$H(x) = \nu(x, |B(x)|^2)B(x) \quad \forall x \in \Omega, \quad (1.3)$$

kde ν je magnetická reluktivita splňující

$$\nu(x, \eta) = \begin{cases} \nu_1 = \frac{1}{\mu_0}, & \forall x \in \Omega_1, \\ \nu_2(\eta), & \forall x \in \Omega_2. \end{cases} \quad (1.4)$$

Konstanta μ_0 je permeabilita vakua. Množiny Ω_1 a Ω_2 jsou disjunktní části Ω , $\Omega \subset \overline{\Omega_1} \cup \overline{\Omega_2}$, představující různé materiály. V našem případě půjde o vzduch pro Ω_1 a ocel statoru a rotoru pro Ω_2 . Reluktivita této oceli je dána vztahem

$$\nu_2(\eta) = \frac{1}{\mu_0} \left(\alpha + (1 - \alpha) \frac{\eta^4}{\eta^4 + \beta} \right) \quad (1.5)$$

pro konstanty $\alpha = 0.0003$ a $\beta = 16000$, viz (Glowinski a Marrocco, 1974). Okrajovou podmínku pro náš problém budeme předpokládat ve tvaru

$$\vec{n} \cdot B = 0 \quad \text{na } \partial\Omega, \quad (1.6)$$

kde \vec{n} je jednotková vnější normála k Ω .

1.2 Magnetický potenciál

Díky (1.2) můžeme předpokládat existenci magnetického potenciálu u (daného až na konstantu) splňujícího

$$B = \operatorname{curl} u,$$

ke chápeme $\operatorname{curl} u = (\partial_2 u, -\partial_1 u)$. Detailnější zdůvodnění viz (Girault a Raviart, 1986). Vidíme, že

$$|B(x)| = |\operatorname{curl} u(x)| = |\nabla u(x)|,$$

což nám spolu s (1.3) dává

$$\nu(x, |B(x)|^2) = \nu(x, |\nabla u(x)|^2) \quad \forall x \in \Omega.$$

To spolu s (1.1) vede k

$$\begin{aligned} f = \operatorname{rot} H &= \operatorname{rot} (\nu(\cdot, |B|^2) B) = \operatorname{rot} (\nu(\cdot, |\operatorname{curl} u|^2) \operatorname{curl} u) \\ &= -\operatorname{div} (\nu(\cdot, |\operatorname{curl} u|^2) \nabla u). \end{aligned}$$

Pro vektor $\vec{t} = (-n_2, n_1)$ kolmý na jednotkovou vnější normálu \vec{n} dostaneme

$$0 = \vec{n} \cdot B = \vec{n} \cdot \operatorname{curl} u = \vec{t} \cdot \nabla u = \frac{\partial u}{\partial t},$$

neboli u je podél hranice $\partial\Omega$ konstantní. Bez újmy na obecnosti můžeme zvolit $u = 0$ na $\partial\Omega$. V řeči magnetického potenciálu tedy můžeme původní úlohu přeformulovat do

$$-\operatorname{div} (\nu(\cdot, |\nabla u|^2) \nabla u) = f \quad \text{v } \Omega, \quad (1.7)$$

$$u = 0 \quad \text{na } \partial\Omega. \quad (1.8)$$

Tímto jsme získali nelineární Dirichletův problém. Díky příznivým vlastnostem funkce $\nu(x, \eta)$ je možné dokázat, že operátor působící na levé straně na u je lipschitzovsky spojitý a silně monotóní. Tyto vlastnosti nám s pomocí teorie monotóních operátorů zajistí řešitelnost této úlohy. Provedení důkazu je v následující kapitole.

2. Řešitelnost úlohy

V této kapitole se budeme zabývat řešitelností úlohy (1.7), (1.8). Provedený postup následuje využívá teorii monotóních operátorů a je převzat z (Křížek a Segeth, 2001).

2.1 Teorie monotóních operátorů

Buď \mathbb{H} Hilbertův prostor se skalárním součinem $\langle \cdot, \cdot \rangle$ a normou $\|\cdot\|$ indukovanou tímto skalárním součinem. Uvažujme (nelineární) operátor $\mathbb{A}: \mathbb{H} \mapsto \mathbb{H}$. Budeme hledat $x \in \mathbb{H}$ řešení rovnice

$$\mathbb{A}x = b \tag{2.1}$$

pro dané $b \in \mathbb{H}$.

Definice 1. Řekneme, že operátor $\mathbb{A}: \mathbb{H} \mapsto \mathbb{H}$ se nazývá

1. *monotóní, jestliže splňuje*

$$\langle \mathbb{A}v_1 - \mathbb{A}v_2, v_1 - v_2 \rangle \geq 0 \quad \forall v_1, v_2 \in \mathbb{H}, \tag{2.2}$$

2. *ostře monotóní, jestliže splňuje*

$$\langle \mathbb{A}v_1 - \mathbb{A}v_2, v_1 - v_2 \rangle > 0 \quad \forall v_1, v_2 \in \mathbb{H}, v_1 \neq v_2, \tag{2.3}$$

3. *silně monotóní, jestliže existuje konstanta $C > 0$ taková, že*

$$\langle \mathbb{A}v_1 - \mathbb{A}v_2, v_1 - v_2 \rangle > C \|v_1 - v_2\|^2 \quad \forall v_1, v_2 \in \mathbb{H}, \tag{2.4}$$

Zřejmě platí, že silně monotóní operátor je i ostře monotóní a ostře monotóní operátor je i monotóní. S pomocí pojmů z této definice nyní můžeme zformulovat věty o řešitelnosti a jednoznačnosti řešení rovnice (2.1). Začneme větou o jednoznačnosti, která je mnohem jednodušší.

Věta 1. *Buď operátor \mathbb{A} v rovnici (2.1) ostře monotóní. Pak má rovnice (2.1) nejvýše jedno řešení.*

Důkaz. Buďte $x_1, x_2 \in \mathbb{H}$ takové, že $\mathbb{A}x_1 = \mathbb{A}x_2 = b$. Pak dostaneme

$$\langle \mathbb{A}x_1 - \mathbb{A}x_2, v_1 - v_2 \rangle = \langle b - b, v_1 - v_2 \rangle = 0.$$

srovnáním s (2.3) z definice ostré monotonie vidíme, že nutně $x_1 = x_2$. □

Věta 2. *Nechť operátor \mathbb{A} v rovnici (2.1) je silně monotóní a lipschitzovsky spojitý. Pak pro každé $b \in \mathbb{H}$ existuje právě jedno řešení rovnice (2.1).*

Důkaz. Definujeme pomocné zobrazení $T_\varepsilon: \mathbb{H} \mapsto \mathbb{H}$ vztahem

$$T_\varepsilon(v) = v - \varepsilon(\mathbb{A}v - b), \quad v \in \mathbb{H}.$$

pro nějaké $\varepsilon > 0$, které zafixujeme později. Dle (2.4) a z lipschitzovské spojitosti pro každé $v_1, v_2 \in \mathbb{H}$ dostaneme

$$\begin{aligned} \|T_\varepsilon(v_1) - T_\varepsilon(v_2)\|^2 &= \|(v_1 - v_2) - \varepsilon(\mathbb{A}v_1 - \mathbb{A}v_2)\|^2 \\ &= \|(v_1 - v_2)\|^2 - 2\varepsilon \langle \mathbb{A}v_1 - \mathbb{A}v_2, v_1 - v_2 \rangle + \varepsilon^2 \|\mathbb{A}v_1 - \mathbb{A}v_2\|^2 \\ &\leq (1 - 2\varepsilon C + \varepsilon^2 C_L^2) \|(v_1 - v_2)\|^2, \end{aligned}$$

kde C_L je konstanta lipschitzovskosti. Zvolme nyní

$$\varepsilon \in \left(0, \frac{2C}{C_L^2}\right),$$

pak dostaneme nerovnost

$$(1 - 2\varepsilon C + \varepsilon^2 C_L^2) < 1,$$

tedy zobrazení T_ε je kontraktivní. Z Banachovy věty o pevném bodě pak dostáváme, že existuje právě jedno $u \in \mathbb{H}$ takové, že $T_\varepsilon(u) = u$. Po úpravě dostaneme, že

$$\mathbb{A}u = b,$$

což jsme chtěli dokázat. □

2.2 Slabá formulace

Slabou formulaci úlohy (1.7) – (1.8) dostaneme klasicky pomocí přenásobení rovnice (1.7) testovací funkcí a integrací per partes.

Definice 2. Řekneme, že $u \in H_0^1(\Omega)$ je slabé řešení úlohy (1.7)– (1.8) pokud $\forall v \in H_0^1(\Omega)$ splňuje

$$a(u, u, v) = \langle f, v \rangle_{L^2(\Omega)}, \quad (2.5)$$

kde

$$a(u, v, w) = \left\langle \nu(\cdot, |\nabla u|^2) \nabla v, \nabla w \right\rangle_{L^2(\Omega)} \quad u, v, w \in H_0^1(\Omega). \quad (2.6)$$

Povšimněme si, že tato definice formy $a(u, v, w)$ je korektní, neboť $\nu(x, \eta)$ je omezená a $\nabla v, \nabla w \in L^2(\Omega)$, tudíž integrál přes oblast Ω je konečný. K vyšetření řešitelnosti této úlohy použijeme teorii monotóních operátorů. Definujeme operátor $\mathbb{A}: H_0^1(\Omega) \mapsto H_0^1(\Omega)$ vztahem

$$\langle \mathbb{A}w, v \rangle_{H_0^1(\Omega)} = a(w, w, v) \quad w, v \in H_0^1(\Omega). \quad (2.7)$$

Pro každé dané $w \in H_0^1(\Omega)$ je $a(w, w, \cdot)$ lineární spojitý funkcional na $H_0^1(\Omega)$, tudíž dle Rieszovy věty existuje právě jedno $\mathbb{A}w \in H_0^1(\Omega)$ a tudíž operátor \mathbb{A} je dobře definovaný. Nyní je pro aplikaci věty 2 potřeba dokázat, že operátor \mathbb{A} je lipschitzovsky spojitý a silně monotóní. K tomu si nejprve připravíme následující lemma:

Lemma 3. *Bud funkce $\nu: \Omega \times \mathbb{R} \mapsto \mathbb{R}$ definovaná jako v (1.4). Pak existují $C_a, C_b > 0$ takové, že pro každé $x \in \Omega$ platí*

$$\left| \nu(x, |\eta|^2)\eta - \nu(x, |\xi|^2)\xi \right| \leq C_a |\eta - \xi| \quad \eta, \xi \in \mathbb{R}^2, \quad (2.8)$$

$$C_b |\eta - \xi|^2 \leq \left(\nu(x, |\eta|^2)\eta - \nu(x, |\xi|^2)\xi \right) \cdot (\eta - \xi) \quad (2.9)$$

Důkaz. Nejprve učiníme pozorování o funkci $\nu(x, \eta)$. Srovnáním s definicí (1.4) snadno ověříme, že existují $C_1, C_2, C_3 > 0$ takové, že

$$C_1 < \nu(x, \eta) < C_2 \quad \forall \eta \geq 0, \quad (2.10)$$

$$\left| \theta \frac{\partial \nu}{\partial \eta}(x, \eta) \right| < C_3 \quad \forall \eta \geq \theta \geq 0 \quad (2.11)$$

Zdůrazněme, že tyto odhady jsou nezávislé na x . Zvolme nyní libovolné $x \in \Omega$ pevné. Definujeme nyní pomocnou funkci $G: \Omega \times \mathbb{R}^2 \mapsto \mathbb{R}^2$ předpisem

$$G(x, \xi) = \nu(x, |\xi|^2)\xi \quad x \in \Omega, \xi \in \mathbb{R}^2. \quad (2.12)$$

Zderivujeme funkci G podle druhé proměnné a dostaneme

$$\frac{dG(x, \xi)}{d\xi} = \begin{pmatrix} 2\xi_1^2 \nu'(x, |\xi|^2) + \nu(x, |\xi|^2) & 2\xi_1 \xi_2 \nu'(x, |\xi|^2) \\ 2\xi_1 \xi_2 \nu'(x, |\xi|^2) & 2\xi_2^2 \nu'(x, |\xi|^2) + \nu(x, |\xi|^2) \end{pmatrix},$$

kde ν' je označena derivace podle druhé proměnné. Nyní pomocí (2.10) a (2.11) můžeme derivace odhadnout

$$\begin{aligned} \left| \frac{dG_1(x, \xi)}{d\xi_1} \right| &\leq \left| 2\xi_1^2 \nu'(x, |\xi|^2) \right| + \left| \nu(x, |\xi|^2) \right| \leq 2C_3 + C_2, \\ \left| \frac{dG_1(x, \xi)}{d\xi_2} \right| &= \left| 2\xi_1 \xi_2 \nu'(x, |\xi|^2) \right| \leq \left| \xi_1^2 \nu'(x, |\xi|^2) \right| + \left| \xi_2^2 \nu'(x, |\xi|^2) \right| \leq 2C_3, \end{aligned}$$

atd. Je tedy vidět, že parciální derivace G jsou spojité a omezené, a tedy G je lipschitzovská ve své druhé proměnné s konstantou lipschitzovskosti nezávislou na x . Vztah (2.8) tedy platí s C_a rovnou konstantě lipschitzovskosti funkce G .

Nyní dokážeme (2.9). Buďte $\eta, \xi \in \mathbb{R}^2$ libovolné, $\theta = \eta - \xi$, $x \in \Omega$ libovolné pevné. Definujeme

$$H(x, t) = \nu(x, |\xi + t\theta|^2)(\xi + t\theta) \cdot \theta \quad t \in [0; 1].$$

Derivací podle t dostaneme

$$\frac{\partial H}{\partial t}(x, t) = \nu(x, |\xi + t\theta|^2) |\theta|^2 + 2\nu'(x, |\xi + t\theta|^2) ((\xi + t\theta) \cdot \theta)^2 \geq C_1 |\eta - \xi|^2,$$

kde jsme využili (2.10) a faktu, že funkce ν je ve své druhé proměnné neklesající, tudíž člen obsahující ν' je nezáporný. Nerovnost (2.9) dostaneme výpočtem

$$\begin{aligned} \left(\nu(x, |\eta|^2)\eta - \nu(x, |\xi|^2)\xi \right) \cdot (\eta - \xi) &= \nu(x, |\eta|^2)\eta \cdot (\eta - \xi) - \nu(x, |\xi|^2)\xi \cdot (\eta - \xi) \\ &= H(x, 1) - H(x, 0) = \frac{\partial H}{\partial t}(x, t_0) \\ &\geq C_1 |\eta - \xi|^2, \end{aligned}$$

kde jsme využili větu o střední hodnotě. Nerovnost (2.9) tedy platí s $C_b = C_1$. \square

Nyní využijeme lemma 3 k důkazu lipschitzovské spojivosti a silné monotonie operátoru \mathbb{A} . Začneme s lipschitzovskou spojivostí. Budte $v_1, v_2 \in H_0^1(\Omega)$ libovolné. Pak z (2.8) dostaneme

$$\left| \nu(x, |\nabla v_1|^2) \nabla v_1 - \nu(x, |\nabla v_2|^2) \nabla v_2 \right| \leq C_a |\nabla v_1 - \nabla v_2| \quad \forall x \in \Omega.$$

Umocněním tohoto vztahu na druhou, integrací přes oblast Ω a opětovným odmocněním dostaneme

$$\left\| \nu(\cdot, |\nabla v_1|^2) \nabla v_1 - \nu(\cdot, |\nabla v_2|^2) \nabla v_2 \right\|_{L^2(\Omega)} \leq C_a \|\nabla(v_1 - v_2)\|_{L^2(\Omega)}.$$

Z tohoto vztahu, (2.6), (2.7) a Cauchy-Schwarzovy nerovnosti dostaneme

$$\begin{aligned} \|\mathbb{A}v_1 - \mathbb{A}v_2\|_{H_0^1(\Omega)} &= \sup_{\substack{w \neq 0 \\ w \in H_0^1(\Omega)}} \frac{\langle \mathbb{A}v_1 - \mathbb{A}v_2, w \rangle_{H_0^1(\Omega)}}{\|w\|_{H_0^1(\Omega)}} \\ &= \sup_{\substack{w \neq 0 \\ w \in H_0^1(\Omega)}} \frac{a(v_1, v_1, w) - a(v_2, v_2, w)}{\|w\|_{H_0^1(\Omega)}} \\ &\leq C_a \sup_{\substack{w \neq 0 \\ w \in H_0^1(\Omega)}} \frac{\|\nabla(v_1 - v_2)\|_{L^2(\Omega)} \|\nabla w\|_{L^2(\Omega)}}{\|w\|_{H_0^1(\Omega)}} \\ &\leq C_a \|v_1 - v_2\|_{H_0^1(\Omega)}. \end{aligned}$$

Vidíme tedy, že operátor \mathbb{A} je skutečně lipschitzovský. Dále ověříme, že operátor \mathbb{A} je silně monotóní. Necht $v_1, v_2 \in H_0^1(\Omega)$ jsou libovolné funkce. Nyní pomocí vztahů (2.6), (2.7) a (2.9) dostaneme

$$\begin{aligned} &\langle \mathbb{A}v_1 - \mathbb{A}v_2, v_1 - v_2 \rangle_{H_0^1(\Omega)} \\ &= \left\langle \nu(\cdot, |\nabla v_1|^2) \nabla v_1 - \nu(\cdot, |\nabla v_2|^2) \nabla v_2, \nabla(v_1 - v_2) \right\rangle_{L^2(\Omega)} \\ &\geq C_b \|\nabla(v_1 - v_2)\|_{L^2(\Omega)}^2 \\ &\geq \widehat{C} \|v_1 - v_2\|_{H_0^1(\Omega)}^2, \end{aligned}$$

kde \widehat{C} je kladná konstanta, jejíž existence plyne z Fridrichsovy nerovnosti. Vidíme tedy, že operátor \mathbb{A} je silně monotóní a můžeme aplikovat větu 2. Dokázali jsme tedy, že úloha (1.7), (1.8) má právě jedno řešení.

3. Diskretizace

V této kapitole si stručně představíme nespojitou Galerkinovu metodu (DGM, discontinuous Galerkin method), konkrétněji její variantu s neúplnou vnitřní penalizací (IIPG, incomplete interior penalty Galerkin). Do větší hloubky i širě rozebírá tuto metodu (Dolejší a M., 2015).

3.1 Triangulace

V této práci budeme uvažovat oblast Ω mnohoúhelníkového tvaru. Ta je pokryta uzavřenými trojúhelníky (množinu všech těchto trojúhelníků označíme \mathcal{T}_h) s navzájem disjunktními vnitřky, přičemž průnik dvou trojúhelníků je vždy buď prázdný, obsahuje jeden bod, nebo je tvořen jednou celou hranou společnou oběma trojúhelníků. Jinými slovy, uvažujeme konformní triangulaci. Označíme

$$h = \max_{K \in \mathcal{T}_h} (\text{diam}(K)).$$

Symbolem \mathcal{F}_h označíme množinu všech hran všech trojúhelníků $K \in \mathcal{T}_h$, dále označíme \mathcal{F}_h^I množinu všech hran uvnitř Ω , tedy vnitřních hran, a \mathcal{F}_h^B množinu všech hran na hranici Ω . Zřejmě tedy platí $\mathcal{F}_h^B \cap \mathcal{F}_h^I = \emptyset$ a $\mathcal{F}_h = \mathcal{F}_h^B \cup \mathcal{F}_h^I$. Navíc, pro každou hranu $\Gamma \in \mathcal{F}_h$ definujeme jednotkovou normálu \vec{n}_Γ . Předpokládáme, že pro všechny prvky \mathcal{F}_h^B má \vec{n}_Γ stejnou orientaci jako vnější normála $\partial\Omega$, pro prvky \mathcal{F}_h^I má orientaci libovolnou, ale pevnou.

3.2 Funkční prostory

Přibližné řešení budeme hledat v prostoru všech po částech polynomiálních funkcí stupně nejvýše $p \in \mathbb{N}$, tedy

$$\mathcal{S}_{h,p} = \left\{ \psi \in L^2(\Omega); \psi|_{\text{int } K} \in P^p(\text{int } K), K \in \mathcal{T}_h \right\}.$$

Zavedeme také označení pro skok $[v]_\Gamma$ a pro střední hodnotu $\langle v \rangle_\Gamma$ funkce v na hraně Γ . Pro $\Gamma \in \mathcal{F}_h^I$ položíme

$$\begin{aligned} \langle v \rangle_\Gamma(x) &= \frac{1}{2} \left(v|_\Gamma^{(+)}(x) + v|_\Gamma^{(-)}(x) \right), \\ [v]_\Gamma(x) &= v|_\Gamma^{(+)}(x) - v|_\Gamma^{(-)}(x), \end{aligned}$$

kde

$$v|_\Gamma^{(\pm)}(x) = \lim_{\delta \rightarrow 0^\pm} v(x + \delta \vec{n}_\Gamma).$$

Pro $\Gamma \in \mathcal{F}_h^B$ položíme $\langle v \rangle_\Gamma(x) = [v]_\Gamma(x) = v|_\Gamma(x)$. Závislost na x nebudeme už dále u $[v]_\Gamma$ a $\langle v \rangle_\Gamma$ explicitně označovat, podobně index Γ , pokud bude z kontextu zřejmé, o jakou hranu se jedná.

3.3 Numerické řešení

Nyní odvodíme diskrétní variantu (2.5). Buď u dostatečně hladké řešení úlohy 1.7, 1.8. Přenásobíme rovnici (1.7) funkcí $w \in \mathcal{S}_{h,p}$, integrujeme přes $K \in \mathcal{T}_h$ a posčítáme přes všechna K . Dostaneme

$$- \sum_{K \in \mathcal{T}_h} \int_K \operatorname{div} (\nu(x, |\nabla u|^2) \nabla u) w \, dx = \sum_{K \in \mathcal{T}_h} \int_K f w \, dx. \quad (3.1)$$

Nyní můžeme na levé straně aplikovat Greenovu větu, čímž dostaneme

$$\begin{aligned} & - \sum_{K \in \mathcal{T}_h} \int_K \operatorname{div} (\nu(x, |\nabla u|^2) \nabla u) w \, dx \\ &= \sum_{K \in \mathcal{T}_h} \int_K \nu(x, |\nabla u|^2) \nabla u \nabla w \, dx \\ & \quad - \sum_{K \in \mathcal{T}_h} \int_{\partial K} \nu(x, |\nabla u|^2) (\nabla u \cdot \vec{n}) w \, dx \end{aligned}$$

V tomto vztahu si povšimneme, že na pravé straně ve druhém integrálu vlastně integrujeme přes všechny hrany všech trojúhelníků. Navíc, přes všechny vnitřní hrany trojúhelníků integrujeme dvakrát, protože každá vnitřní hrana je společná dvěma trojúhelníků. Můžeme tedy rozepsat

$$\begin{aligned} & \sum_{K \in \mathcal{T}_h} \int_{\partial K} \nu(x, |\nabla u|^2) (\nabla u \cdot \vec{n}) w \, dx \\ &= \sum_{\Gamma \in \mathcal{F}_h^B} \int_{\Gamma} \nu(x, |\nabla u|^2) (\nabla u \cdot \vec{n}_{\Gamma}) w \, dx \\ & \quad + \sum_{\Gamma \in \mathcal{F}_h^I} \int_{\Gamma} \nu(x, |\nabla u|^2) (\nabla u) \cdot \left((w \vec{n})|_{K_{1,\Gamma}} + (w \vec{n})|_{K_{2,\Gamma}} \right) dx \\ &= \sum_{\Gamma \in \mathcal{F}_h^B} \int_{\Gamma} \langle \nu(x, |\nabla u|^2) \nabla u \cdot \vec{n}_{\Gamma} \rangle [w] \, dx \\ & \quad + \sum_{\Gamma \in \mathcal{F}_h^I} \int_{\Gamma} \nu(x, |\nabla u|^2) (\nabla u \cdot \vec{n}_{\Gamma}) \left(w|_{\Gamma}^+ - w|_{\Gamma}^- \right) \cdot \vec{n}_{\Gamma} \, dx \\ &= \sum_{\Gamma \in \mathcal{F}_h} \int_{\Gamma} \langle \nu(x, |\nabla u|^2) \nabla u \cdot \vec{n}_{\Gamma} \rangle [w] \, dx, \end{aligned}$$

kde $K_{1,\Gamma}$ je trojúhelník, kterému náleží hrana Γ a jehož vnější normála má opačnou orientaci než je \vec{n}_{Γ} , $K_{2,\Gamma}$ je pak trojúhelník, jemuž náleží hrana Γ a jehož vnější normála má souhlasnou orientaci s \vec{n}_{Γ} . Dále jsme využili předpokládanou regularitu řešení u , konkrétně jsme předpokládali, že

$$\nu(x, |\nabla u|^2) \nabla u = \langle \nu(x, |\nabla u|^2) \nabla u \rangle.$$

Kromě toho, na levou stranu (3.1) ještě přidáme dva členy

$$\sum_{\Gamma \in \mathcal{F}_h^I} \int_{\Gamma} \sigma_{\Gamma} [u] [w] \, dx + \sum_{\Gamma \in \mathcal{F}_h^B} \int_{\Gamma} \sigma_{\Gamma} (u - u_D) w \, dx,$$

kde

$$\sigma_{\Gamma} = \frac{C_w}{\operatorname{diam}(\Gamma)} \quad (3.2)$$

a C_w je kladná konstanta, jejíž hodnotu zvolíme později, u_D je Dirichletova okrajová podmínka. V našem případě je $u_D = 0$, píšeme ji zde proto, aby byl význam druhého integrálu jasnější. Smyslem těchto dvou členů je formou penalizace vynucovat spojitost resp. splnění Dirichletových okrajových podmínek. Můžeme poznamenat, že pro funkci u spojitou a splňující okrajovou podmínku se tyto dva členy vynulují. S touto přípravou můžeme konečně napsat slabou formulaci diskrétního problému analogicky k (2.5).

Definice 3. Řekneme, že $u_h \in \mathcal{S}_{h,p}$ je slabé řešení (3.1), pokud pro všechna $v \in \mathcal{S}_{h,p}$

$$a_h(u_h, u_h, v) = F(v) \quad \forall v \in \mathcal{S}_{h,p}, \quad (3.3)$$

kde

$$\begin{aligned} a_h(u, v, w) = & \sum_{K \in \mathcal{T}_h} \langle \nu(\cdot, |\nabla u|^2) \nabla v, \nabla w \rangle_{L^2(K)} \\ & - \sum_{\Gamma \in \mathcal{F}_h} \langle \langle \nu(\cdot, |\nabla u|^2) \nabla v \rangle \cdot \vec{n}_\Gamma, [w] \rangle_{L^2(\Gamma)} \\ & + \sum_{\Gamma \in \mathcal{F}_h^I} \langle \sigma[v], [w] \rangle_{L^2(\Gamma)} + \sum_{\Gamma \in \mathcal{F}_h^B} \langle \sigma v, w \rangle_{L^2(\Gamma)}, \end{aligned} \quad (3.4)$$

a na pravé straně je funkcional

$$F(v) = \sum_{K \in \mathcal{T}_h} \langle f, v \rangle_{L^2(K)} + \sum_{\Gamma \in \mathcal{F}_h^B} \langle \sigma u_D, u \rangle_{L^2(\Gamma)}, \quad (3.5)$$

kde funkce .

Nyní opět použijeme větu 2 k důkazu existence řešení a jednoznačnosti úlohy (3.3). Postup v této kapitole bude v principu kopírovat postup předchozí kapitoly s tím, že se budeme muset vypořádat s více členy ve formě a_h . Definujeme pro $u, v \in \mathcal{S}_{h,p}$

$$\langle u, v \rangle_{\mathcal{S}_{h,p}} = \sum_{K \in \mathcal{T}_h} \int_K \nabla u \cdot \nabla v \, dx + \sum_{\Gamma \in \mathcal{F}_h} \int_\Gamma \sigma[u][v] \, dx, \quad (3.6)$$

$$\langle u, v \rangle_{\mathcal{S}_{h,p}}^* = \sum_{K \in \mathcal{T}_h} \int_K \nabla u \cdot \nabla v + uv \, dx + \sum_{\Gamma \in \mathcal{F}_h} \int_\Gamma \sigma[u][v] \, dx. \quad (3.7)$$

Je přímočaré ověřit, že $\langle \cdot, \cdot \rangle_{\mathcal{S}_{h,p}}$ a $\langle \cdot, \cdot \rangle_{\mathcal{S}_{h,p}}^*$ jsou skalární součiny na $\mathcal{S}_{h,p}$ a vybavíme-li tento prostor jedním z těchto skalárních součinů a příslušnou indukovanou normou $\|\cdot\|_{\mathcal{S}_{h,p}}^2 = \langle \cdot, \cdot \rangle_{\mathcal{S}_{h,p}}$ resp. $(\|\cdot\|_{\mathcal{S}_{h,p}}^*)^2 = \langle \cdot, \cdot \rangle_{\mathcal{S}_{h,p}}^*$, dostaneme Hilbertův prostor. Protože $\mathcal{S}_{h,p}$ je konečnědimenzionální, normy $\|\cdot\|_{\mathcal{S}_{h,p}}$ a $\|\cdot\|_{\mathcal{S}_{h,p}}^*$ jsou ekvivalentní. Využitím této ekvivalence a srovnáním (3.5) s (3.7) dostaneme

$$\begin{aligned} F(v) & \leq \sum_{K \in \mathcal{T}_h} \|f\|_{L^2(K)} \|v\|_{L^2(K)} + \sum_{\Gamma \in \mathcal{F}_h^B} \sigma_\Gamma \|u_D\|_{L^2(\Gamma)} \|v\|_{L^2(\Gamma)} \\ & \leq \left(\|f\|_{L^2(\Omega)} + \|\sqrt{\sigma}\|_{L^\infty} \|u_D\|_{L^2(\partial\Omega)} \right) \|v\|_{\mathcal{S}_{h,p}}^* \\ & \leq C \left(\|f\|_{L^2(\Omega)} + \|\sqrt{\sigma}\|_{L^\infty} \|u_D\|_{L^2(\partial\Omega)} \right) \|v\|_{\mathcal{S}_{h,p}} \end{aligned}$$

Vidíme tedy, že F je spojitý lineární funkcionál na $(\mathcal{S}_{h,p}; \langle \cdot, \cdot \rangle_{\mathcal{S}_{h,p}})$ a dle Rieszovy věty existuje právě jedno $f_h \in (\mathcal{S}_{h,p}; \langle \cdot, \cdot \rangle_{\mathcal{S}_{h,p}})$, které ho reprezentuje. Předtím, než definujeme operátor \mathbb{A}_h analogicky k předchozí kapitole, musíme učinit přípravu. Budeme potřebovat dvě přípravná lemmata. V jejich důkazu budeme potřebovat následující větu.

Věta 4. *Bud' \mathcal{T}_h triangulace oblasti Ω , $K \in \mathcal{T}_h$, Γ je jedna hrana K , $v \in \mathcal{S}_{h,p}$, p je řád polynomiální aproximace. Pak existuje konstanta C_{ie} nezávislá na diskretizačních parametrech taková, že platí*

$$\|v\|_{L^2(\Gamma)}^2 \leq C_{ie} \frac{p^2}{\text{diam}(\Gamma)} \|v\|_{L^2(K)}^2 \quad (3.8)$$

$$\|\nabla v\|_{L^2(\Gamma)}^2 \leq C_{ie} \frac{p^2}{\text{diam}(\Gamma)} \|\nabla v\|_{L^2(K)}^2 \quad (3.9)$$

Důkaz. Důkaz viz (Dolejší a Feistauer, 2005) □

Lemma 5. *Bud' forma a_h definovaná jako v (3.4). Pak existuje konstanta $C > 0$ taková, že pro všechny $v_1, v_2, w \in \mathcal{S}_{h,p}$ platí nerovnost*

$$|a_h(v_1, v_1, w) - a_h(v_2, v_2, w)| \leq C \|v_1 - v_2\|_{\mathcal{S}_{h,p}} \|w\|_{\mathcal{S}_{h,p}}, \quad (3.10)$$

speciálně platí

$$a_h(v_1, v_1, w) \leq C \|v_1\|_{\mathcal{S}_{h,p}} \|w\|_{\mathcal{S}_{h,p}}. \quad (3.11)$$

Důkaz. S použitím (3.4) dostaneme následující odhad

$$\begin{aligned} & |a_h(v_1, v_1, w) - a_h(v_2, v_2, w)| \\ & \leq \sum_{K \in \mathcal{T}_h} \int_K |\mu(\cdot, \nabla v_1) \nabla v_1 - \mu(\cdot, \nabla v_2) \nabla v_2| |\nabla w| \, dx \\ & + \sum_{\Gamma \in \mathcal{F}_h^B} \int_{\Gamma} |\mu(\cdot, \nabla v_1) \nabla v_1 - \mu(\cdot, \nabla v_2) \nabla v_2| |w| \, dx \\ & + \sum_{\Gamma \in \mathcal{F}_h^I} \int_{\Gamma} \langle \mu(\cdot, \nabla v_1) \nabla v_1 - \mu(\cdot, \nabla v_2) \nabla v_2 \rangle [w] \, dx \\ & + \sum_{\Gamma \in \mathcal{F}_h} \int_{\Gamma} \sigma |[v_1 - v_2]| |[w]| \, dx \\ & = T_1 + T_2 + T_3 + T_4 \end{aligned}$$

Nyní postupně odhadneme jednotlivé členy T_1 až T_4 . Začneme s T_1 . S pomocí (2.8) a Hölderovy nerovnosti dostaneme

$$\begin{aligned} T_1 & = \sum_{K \in \mathcal{T}_h} \int_K |\mu(\cdot, \nabla v_1) \nabla v_1 - \mu(\cdot, \nabla v_2) \nabla v_2| |\nabla w| \, dx \\ & \leq C_a \left(\sum_{K \in \mathcal{T}_h} \int_K |\nabla(v_1 - v_2)|^2 \, dx \right)^{\frac{1}{2}} \left(\sum_{K \in \mathcal{T}_h} \int_K |\nabla w|^2 \, dx \right)^{\frac{1}{2}} \\ & \leq C_a \|v_1 - v_2\|_{\mathcal{S}_{h,p}} \|w\|_{\mathcal{S}_{h,p}} \end{aligned}$$

Pro odhad členu T_2 použijeme opět (2.8) a Hölderovu nerovnost. Dostaneme

$$\begin{aligned} T_2 &= \sum_{\Gamma \in \mathcal{F}_h^B} \int_{\Gamma} |\mu(\cdot, \nabla v_1) \nabla v_1 - \mu(\cdot, \nabla v_2) \nabla v_2| |w| \, dx \\ &\leq C_a \sum_{\Gamma \in \mathcal{F}_h^B} \int_{\Gamma} |\nabla(v_1 - v_2)| |w| \, dx \\ &\leq C_a \left(\sum_{\Gamma \in \mathcal{F}_h^B} \int_{\Gamma} \sigma^{-1} |\nabla(v_1 - v_2)|^2 \, dx \right)^{\frac{1}{2}} \left(\sum_{\Gamma \in \mathcal{F}_h^B} \int_{\Gamma} \sigma |w|^2 \, dx \right)^{\frac{1}{2}}. \end{aligned}$$

První integrál odhadneme pomocí (3.9), tím dostaneme

$$\begin{aligned} \sum_{\Gamma \in \mathcal{F}_h^B} \int_{\Gamma} \sigma^{-1} |\nabla(v_1 - v_2)|^2 \, dx &\leq \sum_{\Gamma \in \mathcal{F}_h^B} C_{ie} \frac{p^2}{\text{diam } \Gamma} \int_{K_{\Gamma}} \frac{\text{diam } \Gamma}{C_w} |\nabla(v_1 - v_2)|^2 \, dx \\ &\leq 2C_{ie} \frac{p^2}{C_w} \sum_{K \in \mathcal{T}_h} \int_K |\nabla(v_1 - v_2)|^2 \, dx, \end{aligned}$$

kde jsme použili označení K_{Γ} pro trojúhelník, kterému patří hrana Γ . Také jsme využili fakt, že každý trojúhelník může mít maximálně dvě hraniční hrany a tudíž jsme mohli přejít k sumě přes všechny trojúhelníky. Po dosazení tohoto mezivýpočtu do odhadu pro T_2 máme

$$\begin{aligned} T_2 &\leq C_a \sqrt{2C_{ie} \frac{p^2}{C_w}} \left(\sum_{K \in \mathcal{T}_h} \int_K |\nabla(v_1 - v_2)|^2 \, dx \right)^{\frac{1}{2}} \left(\sum_{\Gamma \in \mathcal{F}_h^B} \int_{\Gamma} \sigma |w|^2 \, dx \right)^{\frac{1}{2}} \\ &\leq C_a \sqrt{2C_{ie} \frac{p^2}{C_w}} \|v_1 - v_2\|_{\mathcal{S}_{h,p}} \|w\|_{\mathcal{S}_{h,p}}. \end{aligned}$$

Při odhadu členu T_3 budeme postupovat analogicky, opět začneme uplatněním (2.8) a Hölderovy nerovnosti,

$$\begin{aligned} T_3 &= \sum_{\Gamma \in \mathcal{F}_h^I} \int_{\Gamma} \langle |\mu(\cdot, \nabla v_1) \nabla v_1 - \mu(\cdot, \nabla v_2) \nabla v_2| \rangle [w] \, dx \\ &\leq C_a \sum_{\Gamma \in \mathcal{F}_h^I} \int_{\Gamma} \langle |\nabla(v_1 - v_2)| \rangle [|w|] \, dx \\ &\leq C_a \left(\sum_{\Gamma \in \mathcal{F}_h^I} \int_{\Gamma} \sigma^{-1} \langle |\nabla(v_1 - v_2)| \rangle^2 \, dx \right)^{\frac{1}{2}} \left(\sum_{\Gamma \in \mathcal{F}_h^I} \int_{\Gamma} \sigma [w]^2 \, dx \right)^{\frac{1}{2}}. \end{aligned}$$

Nyní si provedeme mezivýpočet, kde odhadneme první z uvedených integrálů,

přičemž využijeme (3.9),

$$\begin{aligned}
& \sum_{\Gamma \in \mathcal{F}_h^I} \int_{\Gamma} \sigma^{-1} \langle \nabla(v_1 - v_2) \rangle^2 dx \\
& \leq \sum_{\Gamma \in \mathcal{F}_h^I} \frac{1}{2} \sigma^{-1} \int_{\Gamma} \left| \nabla(v_1 - v_2)|_{\Gamma}^{(+)} \right|^2 + \left| \nabla(v_1 - v_2)|_{\Gamma}^{(-)} \right|^2 dx \\
& \leq C_{ie} \frac{p^2}{C_w} \sum_{\Gamma \in \mathcal{F}_h^I} \frac{1}{2} \left(\int_{K_{\Gamma,1}} |\nabla(v_1 - v_2)|^2 dx + \int_{K_{\Gamma,2}} |\nabla(v_1 - v_2)|^2 dx \right) \\
& \leq C_{ie} \frac{p^2}{C_w} \sum_{\Gamma \in \mathcal{F}_h^I} \max \left(\int_{K_{\Gamma,1}} |\nabla(v_1 - v_2)|^2 dx, \int_{K_{\Gamma,2}} |\nabla(v_1 - v_2)|^2 dx \right) \\
& \leq 3C_{ie} \frac{p^2}{C_w} \sum_{K \in \mathcal{T}_h} \int_K |\nabla(v_1 - v_2)|^2 dx,
\end{aligned}$$

kde $K_{\Gamma,1}, K_{\Gamma,2}$ je první resp. druhý trojúhelník, kterému náleží hrana Γ . Při přechodu od sumy přes hrany k sumě přes trojúhelníky jsme využili, že každému trojúhelníku náleží nejvýše tři vnitřní hrany, odtud trojka v konstantě před sumou. Tento mezivýsledek dosadíme zpět do odhadu členu T_3 ,

$$\begin{aligned}
T_3 & \leq C_a \sqrt{3C_{ie} \frac{p^2}{C_w}} \left(\sum_{K \in \mathcal{T}_h} \int_K |\nabla(v_1 - v_2)|^2 dx \right)^{\frac{1}{2}} \left(\sum_{\Gamma \in \mathcal{F}_h^I} \int_{\Gamma} \sigma [w]^2 dx \right)^{\frac{1}{2}} \\
& \leq C_a \sqrt{3C_{ie} \frac{p^2}{C_w}} \|v_1 - v_2\|_{\mathcal{S}_{h,p}} \|w\|_{\mathcal{S}_{h,p}}.
\end{aligned}$$

Odhad členu T_4 provedeme přímočaře pomocí Hölderovy nerovnosti,

$$\begin{aligned}
T_4 & = \sum_{\Gamma \in \mathcal{F}_h} \int_{\Gamma} \sigma |[v_1 - v_2]| |[w]| dx \\
& \leq \left(\sum_{\Gamma \in \mathcal{F}_h} \int_{\Gamma} \sigma |[v_1 - v_2]|^2 dx \right)^{\frac{1}{2}} \left(\sum_{\Gamma \in \mathcal{F}_h} \int_{\Gamma} \sigma |[w]|^2 dx \right)^{\frac{1}{2}} \\
& \leq C_a \|v_1 - v_2\|_{\mathcal{S}_{h,p}} \|w\|_{\mathcal{S}_{h,p}}.
\end{aligned}$$

Dosadíme-li odhady pro jednotlivé členy T_1 až T_4 do původního odhadu, vidíme, že tvrzení lemmatu platí. \square

Lemma 6. *Bud' forma a_h definovaná jako v (3.4). Zvolme C_w tak, že*

$$C_w \geq \frac{5p^2 C_a^2 C_{ie}}{C_b}. \quad (3.12)$$

Pak pro všechny $w_1, w_2 \in \mathcal{S}_{h,p}$ platí nerovnost

$$a_h(w_1, w_1, w_1 - w_2) - a_h(w_2, w_2, w) \geq \frac{1}{2} \min \{C_b, 1\} \|w\|_{\mathcal{S}_{h,p}}^2, \quad (3.13)$$

kde $w = w_1 - w_2$.

Důkaz. S použitím nerovností (2.8) odhadneme levou stranu (3.13), dostaneme

$$\begin{aligned}
& a_h(w_1, w_1, w_1 - w_2) - a_h(w_2, w_2, w) \geq \\
& C_b \sum_{K \in \mathcal{T}_h} \int_K |\nabla w|^2 dx \\
& - \sum_{\Gamma \in \mathcal{F}_h^B} \int_{\Gamma} |\mu(\cdot, \nabla w_1) \nabla w_1 - \mu(\cdot, \nabla w_2) \nabla w_2| |w| dx \\
& - \sum_{\Gamma \in \mathcal{F}_h^I} \int_{\Gamma} \langle |\mu(\cdot, \nabla w_1) \nabla w_1 - \mu(\cdot, \nabla w_2) \nabla w_2| \rangle [w] dx \\
& + \sum_{\Gamma \in \mathcal{F}_h^B} \int_{\Gamma} \sigma w^2 dx + \sum_{\Gamma \in \mathcal{F}_h^I} \int_{\Gamma} \sigma [w]^2 dx \\
& = T_1 - T_2 - T_3 + T_5 + T_5
\end{aligned}$$

Členy T_2 a T_3 odhadneme analogicky jako v důkazu předchozího lemmatu, získáme

$$\begin{aligned}
T_2 & \leq C_a \sqrt{2C_{ie} \frac{p^2}{C_w}} \left(\sum_{K \in \mathcal{T}_h} \int_K |\nabla w|^2 dx \right)^{\frac{1}{2}} \left(\sum_{\Gamma \in \mathcal{F}_h^B} \int_{\Gamma} \sigma w^2 dx \right)^{\frac{1}{2}}, \\
T_3 & \leq C_a \sqrt{3C_{ie} \frac{p^2}{C_w}} \left(\sum_{K \in \mathcal{T}_h} \int_K |\nabla w|^2 dx \right)^{\frac{1}{2}} \left(\sum_{\Gamma \in \mathcal{F}_h^I} \int_{\Gamma} \sigma [w]^2 dx \right)^{\frac{1}{2}}.
\end{aligned}$$

Na tyto odhady aplikujeme nerovnost $ab \leq \frac{1}{2}a^2 + \frac{1}{2}b^2$ a dostaneme

$$\begin{aligned}
T_2 & \leq \frac{2C_a^2 C_{ie} p^2}{2C_w} \sum_{K \in \mathcal{T}_h} \int_K |\nabla w|^2 dx + \frac{1}{2} \sum_{\Gamma \in \mathcal{F}_h^B} \int_{\Gamma} \sigma w^2 dx, \\
T_3 & \leq \frac{3C_a^2 C_{ie} p^2}{2C_w} \sum_{K \in \mathcal{T}_h} \int_K |\nabla w|^2 dx + \frac{1}{2} \sum_{\Gamma \in \mathcal{F}_h^I} \int_{\Gamma} \sigma [w]^2 dx.
\end{aligned}$$

Tyto nerovnosti dosadíme do původního odhadu a po úpravě dostaneme

$$\begin{aligned}
& a_h(w_1, w_1, w_1 - w_2) - a_h(w_2, w_2, w) \geq \\
& C_b \left(1 - \frac{5C_a^2 C_{ie} p^2}{2C_b C_w} \right) \sum_{K \in \mathcal{T}_h} \int_K |\nabla w|^2 dx \\
& + \frac{1}{2} \sum_{\Gamma \in \mathcal{F}_h^B} \int_{\Gamma} \sigma w^2 dx + \frac{1}{2} \sum_{\Gamma \in \mathcal{F}_h^I} \int_{\Gamma} \sigma [w]^2 dx.
\end{aligned}$$

Z předpokladu (3.12) plyne

$$1 - \frac{5C_a^2 C_{ie} p^2}{2C_b C_w} \geq \frac{1}{2},$$

a tudíž tvrzení lemmatu platí. \square

Od této chvíle budeme předpokládat, že C_w splňuje (3.12). Podobně jako v předchozí kapitole, definujeme operátor $\mathbb{A}_h: \mathcal{S}_{h,p} \mapsto \mathcal{S}_{h,p}$ vztahem

$$\langle \mathbb{A}_h w, v \rangle_{\mathcal{S}_{h,p}} = a_h(w, w, v) \quad w, v \in \mathcal{S}_{h,p}. \quad (3.14)$$

Pro každé dané $w \in \mathcal{S}_{h,p}$ je $a_h(w, w, \cdot)$ lineární funkcionál na $\mathcal{S}_{h,p}$, navíc dle (3.11) je spojitý. Dle Rieszovy tedy existuje právě jedno $\mathbb{A}_h w \in \mathcal{S}_{h,p}$, tudíž operátor \mathbb{A} je dobře definovaný.

Pro aplikaci věty 2.1 nyní musíme ověřit, že \mathbb{A}_h je silně monotónní a lipschitzovsky spojitý. Začneme se silnou monotonií. Buďte $w_1, w_2 \in \mathcal{S}_{h,p}$, pak dle (3.14) a dle (3.13)

$$\begin{aligned} & \langle \mathbb{A}_h w_1 - \mathbb{A}_h w_2, w_1 - w_2 \rangle_{\mathcal{S}_{h,p}} \\ &= \langle \mathbb{A}_h w_1, w_1 - w_2 \rangle_{\mathcal{S}_{h,p}} - \langle \mathbb{A}_h w_2, w_1 - w_2 \rangle_{\mathcal{S}_{h,p}} \\ &= a_h(w_1, w_1, w_1 - w_2) - a_h(w_2, w_2, w_1 - w_2) \\ &\geq \frac{1}{2} \min \{C_b, 1\} \|w\|_{\mathcal{S}_{h,p}}^2 \end{aligned}$$

a vidíme tedy, že \mathbb{A}_h je silně monotónní. Nyní ověříme lipschitzovskou spojitost. Buďte opět $w_1, w_2 \in \mathcal{S}_{h,p}$, pak pomocí (3.14) a (3.10) dostaneme

$$\begin{aligned} \|\mathbb{A}w_1 - \mathbb{A}w_2\|_{\mathcal{S}_{h,p}} &= \sup_{\substack{w \neq 0 \\ w \in \mathcal{S}_{h,p}}} \frac{\langle \mathbb{A}w_1 - \mathbb{A}w_2, w \rangle_{\mathcal{S}_{h,p}}}{\|w\|_{\mathcal{S}_{h,p}}} \\ &= \sup_{\substack{w \neq 0 \\ w \in \mathcal{S}_{h,p}}} \frac{a_h(w_1, w_1, w) - a_h(w_2, w_2, w)}{\|w\|_{\mathcal{S}_{h,p}}} \\ &\leq C \sup_{\substack{w \neq 0 \\ w \in \mathcal{S}_{h,p}}} \frac{\|w_1 - w_2\|_{\mathcal{S}_{h,p}} \|w\|_{\mathcal{S}_{h,p}}}{\|w\|_{\mathcal{S}_{h,p}}} \\ &= C \|w_1 - w_2\|_{\mathcal{S}_{h,p}}. \end{aligned}$$

Operátor \mathbb{A}_h je tedy i lipschitzovský. Dle věty 2.1 tedy existuje právě jedno řešení problému (3.3).

3.4 Soustava algebraických rovnic

Úlohu (3.3) nyní převedeme na soustavu nelineárních algebraických rovnic. Buď $\mathcal{B}_h = \{\psi_i(x)\}_{i=1}^N$ báze prostoru $\mathcal{S}_{h,p}$ tvořená lineárně nezávislými funkcemi. Každou funkci $w_h \in \mathcal{S}_{h,p}$ pak můžeme zapsat ve tvaru

$$w_h = \sum_{j=1}^N \xi_j \psi_j,$$

tedy ξ_j jsou souřadnice funkce w_h v bázi \mathcal{B}_h . Nyní definujeme zobrazení $F_h: \mathbb{R}^N \mapsto \mathbb{R}^N$ předpisem

$$(F_h(\xi))_i = a_h(w_h, w_h, \psi_i) - F(\psi_i) \quad \forall i \in 1 \dots N$$

Úlohu (3.3) tedy můžeme ekvivalentně formulovat takto: Nalezněte $\xi \in \mathbb{R}^N$ takové, že bude platit

$$(F_h(\xi))_i = 0 \quad \forall i \in 1 \dots N,$$

máme tedy N nelineárních rovnic pro N neznámých. Úlohu v tomto tvaru je možné řešit například Newtonovou metodou. Pro aplikaci Newtonovy metody je ovšem nutné vyčíslit matici $\frac{\partial(F_h(\xi))_i}{\partial\xi_j}$, což je náročný úkol. Tomu se můžeme vyhnout tím, že linearizujeme formu a_h a místo matice $\frac{\partial(F_h(\xi))_i}{\partial\xi_j}$ použijeme tzv. matici toku. Tento přístup byl původně vyvinut pro řešení stlačitelných Navier-Stokesových rovnic, např. viz (Dolejší a kol., 2011). Matici toku $C_h \in \mathbb{R}^{N \times N}$ definujeme předpisem

$$(C_h(\xi))_{i,j} = a_h(w_h, \psi_i, \psi_j) \quad \forall i, j \in 1 \dots N.$$

Stále předpokládáme, že ξ je vektor souřadnic funkce w_h v bázi \mathcal{B}_h . O matici toku C_h pak můžeme říci, že pro všechna $i, j \in 1 \dots N$ platí

$$\begin{aligned} (F_h(\xi))_i &\approx (C_h(\xi)\xi)_i - F(\psi_i), \\ \frac{(F_h(\xi))_i}{\partial\xi_j} &\approx (C_h(\xi))_{i,j} \end{aligned}$$

Myšlenka linearizace tedy spočívá v tom, že neuvažujeme nelinearitu formy a_h v její první proměnné. Nyní použijeme algoritmus pro tlumenou Newtonovu metodu, viz (Deuffhard, 2004). Postupujeme podle následujících kroků:

1. Je dáno $\xi^0 \in \mathbb{R}^N$
2. Pro známé ξ^l najdeme řešení $d^l \in \mathbb{R}^N$ rovnice

$$C_h(\xi^l)d = -F_h(\xi^l)$$

3. Položíme

$$\xi^{l+1} = \xi^l + \lambda^l d^l,$$

kde $\lambda^l \in (0, 1]$ zvolíme tak, aby platilo $\|F_h(\xi^{l+1})\| / \|F_h(\xi^l)\| < 1$

4. Pokud není splněno zastavovací kritérium, tak zpět na krok 2.

V každé iteraci algoritmu je třeba v druhém kroku spočítat soustavu N lineárních rovnic o N neznámých. Lineární řešič použitý pro tuto soustavu byl GMRES.

4. Tvorba sítě

Sítě pro *adgfem* byly doposud vytvářeny buď zcela ručně (pro ty úplně nejjednodušší případy), nebo pomocí ad hoc programů nebo skriptů na jedno použití. Tento přístup doposud stačil, protože *adgfem* má schopnost sítě adaptivně upravovat, takže stačilo vygenerovat relativně hrubou síť s malým počtem elementů. Pro tuto práci jsem si ovšem vybral geometrii, která byla podstatně komplexnější, než ty, které se s pomocí *adgfem* počítaly doposud. Bylo by sice možné napsat skript, který síť pro tuto geometrii vygeneroval, bylo by to ovšem velmi pracné. Rozhodl jsem se proto místo toho vytvořit obecnější postup, který bude znovu použitelný i pro úplně jiné geometrie.

Tomu odpovídá i způsob zpracování této kapitoly, kde zabíhám do detailů více, než je u ostatních prací obvyklé. Tento přístup jsem zvolil záměrně, aby byl postup zde uvedený zdokumentovaný co nejsnáze reprodukovatelný.

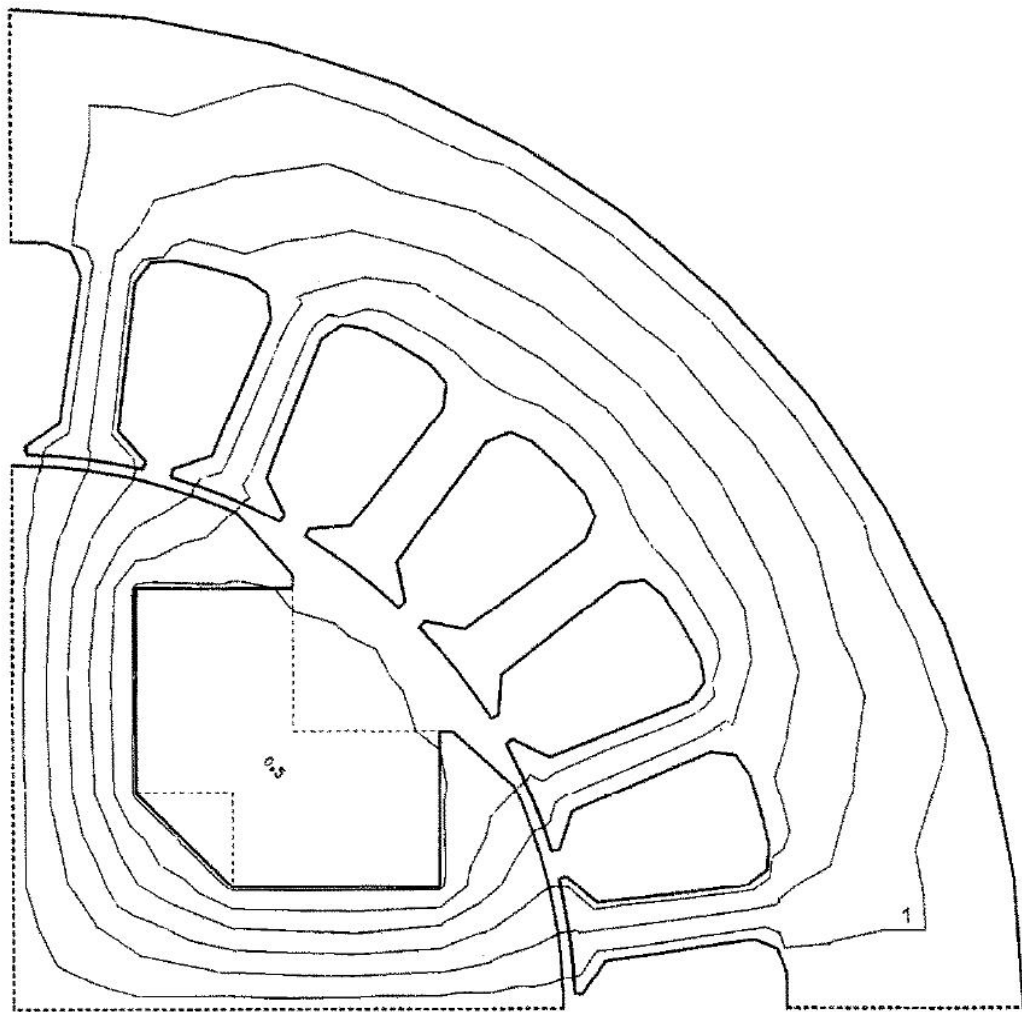
4.1 Definice geometrie

Geometrie pro tuto práci je převzata z Glowinski a Marrocco (1974). Geometrii použitou pro výpočet můžete vidět na obrázku 4.1. Ta představuje čtvrtinu z řezu alternátorem (viz. 4.2). Podobně jako autor původního článku předpokládáme pro jednoduchost symetrii okolo vodorovné a svislé osy, takže si můžeme dovolit počítat jen jednu čtvrtinu. Původní článek neobsahuje kromě obrázku žádnou přesnější definici geometrie, bylo proto nutné se spokojit s tím. Abych z obrázku pokud extrahoval pokud možno co nejpřesnější geometrii, použil jsem nástroj *WebPlotDigitizer* (viz Rohatgi). *WebPlotDigitizer* je svobodná open-source aplikace (licence GPLv3), kterou lze spustit buď přímo ve webovém prohlížeči, nebo stáhnout a spustit na vlastním počítači. Zdrojový kód je k dispozici na GitHubu. *WebPlotDigitizer* je přímo určen pro odečítání hodnot z obrázků. V tomto případě se jednalo o obrázek 4.1. Stačí pak zadat dva referenční body pro každou osu, čímž je určen souřadný systém, a pak lze odečítat souřadnice kteréhokoliv bodu na obrázku. Tím jsem získal soubor se souřadnicemi význačných bodů, který lze nalézt v příloze 5.2. Soubor jsem si pojmenoval `points.xy`. Původní článek neobsahuje informaci o rozměru alternátoru, předpokládal jsem tedy, že jeho průměr je 1 m. Protože je na první pohled vidět, že stator i rotor má několikačetnou symetrii, ušetřil jsem si práci tím, že jsem vynesl jen ty body, které nebylo možné získat pomocí nějakého zrcadlení nebo otočení. Tím se počet bodů podařilo udržet daleko snesitelnější pro ruční zacházení.

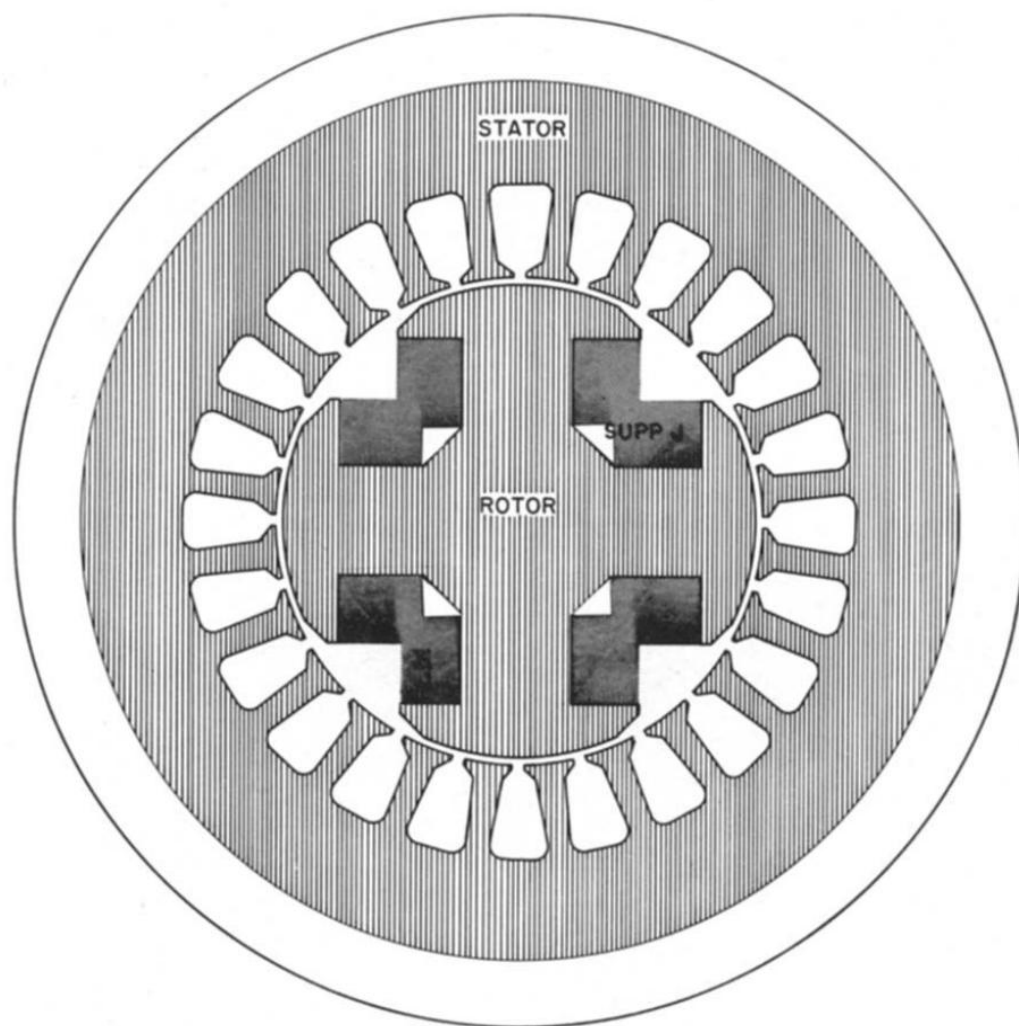
4.2 SALOME

Dalším nástrojem v řetězci je program *SALOME*. *SALOME* je multiplatformní open-source program určený pro preprocessing a postprocessing numerických výpočtů, viz SAL. Krom jiného obsahuje nástroje na práci s geometrií a síťovače, které použijeme.

Jako první krok, který je potřeba učinit, musíme načíst souřadnice bodů, které jsme dostali z *WebPlotDigitizeru*. Bylo by samozřejmě možné načíst body ručně



Obrázek 4.1: Geometrie alternátoru z Glowinski a Marrocco (1974) použitá jako vzor pro vytváření sítě. Zdroj bohužel nezmiňuje rozměry, pro jednoduchost jsme tedy předpokládali, že průměr alternátoru je 1 m.



Obrázek 4.2: Celý alternátor z Glowinski a Marrocco (1974). Na obrázku je popsán stator a rotor (svisle šrafované části). Šedá část popsaná jako SUPP J je vinutí cívek rotoru, kde bude proudit proud budící magnetické pole. Vinutí cívek statoru je zanedbáno. Bílé části jsou vzduch.

kopírováním ze souboru `points.xy` číslo po čísle, ale bylo by to pracné. Místo toho jsem hledal cestu, jak proces automatizovat. *SALOME* pro tento účel obsahuje funkci *Dump Study*. Ta zapíše právě otevřenou studii do Pythonového skriptu, který je pak možné ze *SALOME* znovu otevřít a skript vytvoří všechny objekty, které původní studie obsahovala. Mezi tím je ovšem možné skript modifikovat tak, aby kromě původní studie načel i další objekty. Základy Pythonu není těžké si osvojit a tudíž se otvírá možnost velkou část repetitivní práce automatizovat.

Právě tímto způsobem jsem vytvořil Pythonový skript (viz příloha 5.2), který načte ze souboru všechny body předtím odečtené ve *WebPlotDigitizeru*. Většinu skriptu vytvořilo samo *SALOME*, část, kterou bylo třeba měnit jsem vyznačil a stručně okomentoval. Následuje postup tvorby geometrie krok po kroku:

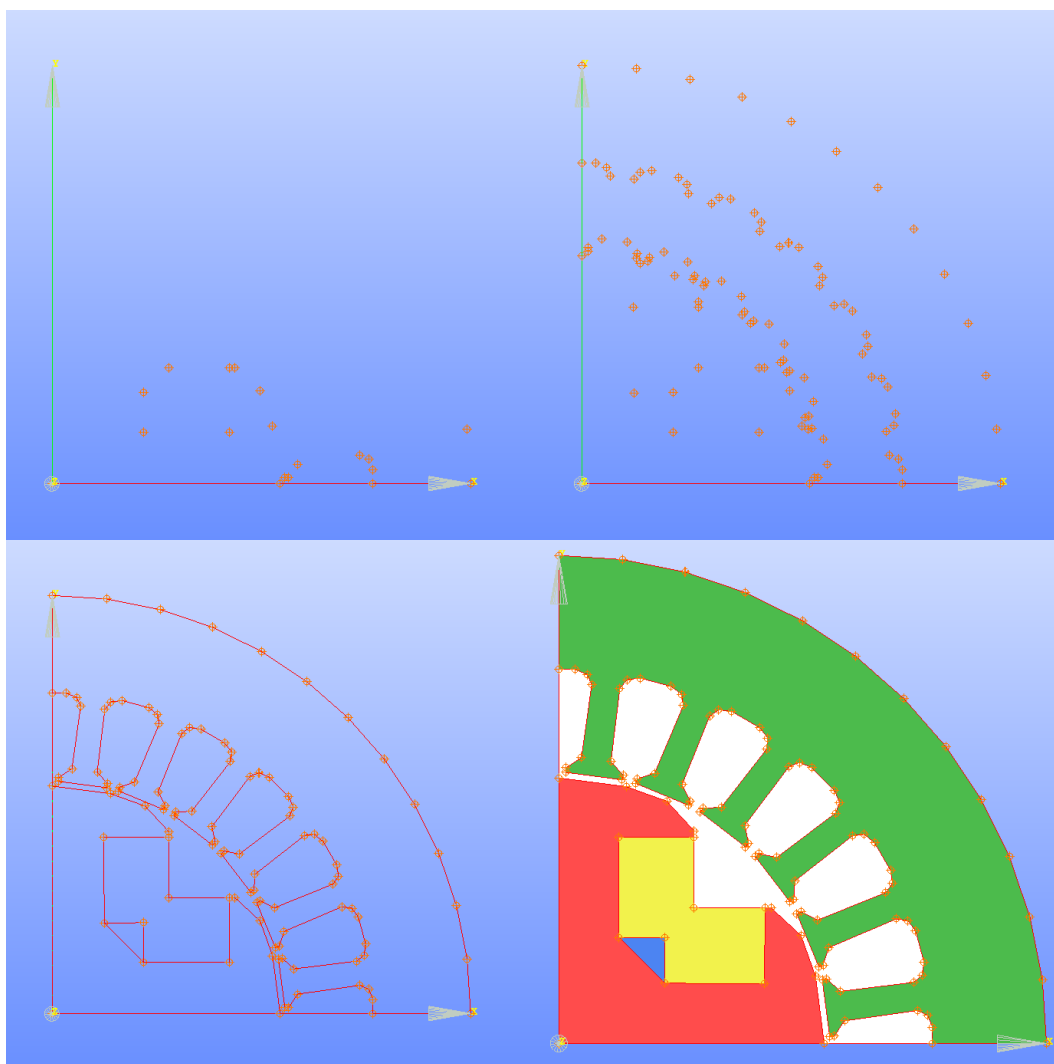
1. Skript spustíme pomocí *File > Load Script*. Tím načteme body ze souboru `points.xy`. Měli bysme se tím dostat do situace nakreslené na obrázku 4.3 vlevo nahoře.
2. Dalším krokem je rozkopírování bodů pomocí funkcí *Operations > Transformation > Mirror Image* a *Operations > Transformation > Rotation*. Jednotlivé operace jistě není třeba rozepisovat dopodrobna. Tím se dostáváme do situace nakreslené na obrázku 4.3 vpravo nahoře.
3. Body pospojujeme pomocí objektu *curve*, kde zaklikneme možnost *Build a closed curve* a manuálně vyklikáme, které body chceme pospojovat. Tím se na obrázku 4.3 posouváme do situace zachycené vlevo dole.
4. Nyní vytvoříme každou oblast zvlášť pomocí objektu *Face*. Tím si zajistíme, že síť bude pro tyto oblasti generována zvlášť a vnitřní hrany oddělující jednotlivé oblasti budou respektovány. Dostáváme se do situace na obrázku 4.3 vpravo dole. Nakonec ještě vytvoříme jeden objekt typu *shell*, do kterého zahrneme všechny objekty typu *face*. Právě na tento objekt se budeme odkazovat během tvorby sítě.

Nyní máme připravenou geometrii a můžeme se pustit do samotného síťování. V *SALOME* přepneme z modulu *Geometry* do modulu *Mesh*. Zvolíme možnost *Create Mesh* a v dialogovém okně vyplníme parametry síťování. V tabulce 4.1 jsem vypsal možnosti, které jsem použil já a které vedou na použitelné sítě, nicméně laskavý čtenář může experimentovat. Jak je vidět u parametru *Length*, který řídí, jak jemné je dělení hran, použil jsem dvě různé hodnoty a vygeneroval dvě různé sítě, viz obrázky 4.4 a 4.5. Sítě mají 7294 resp. 2154 trojúhelníků.

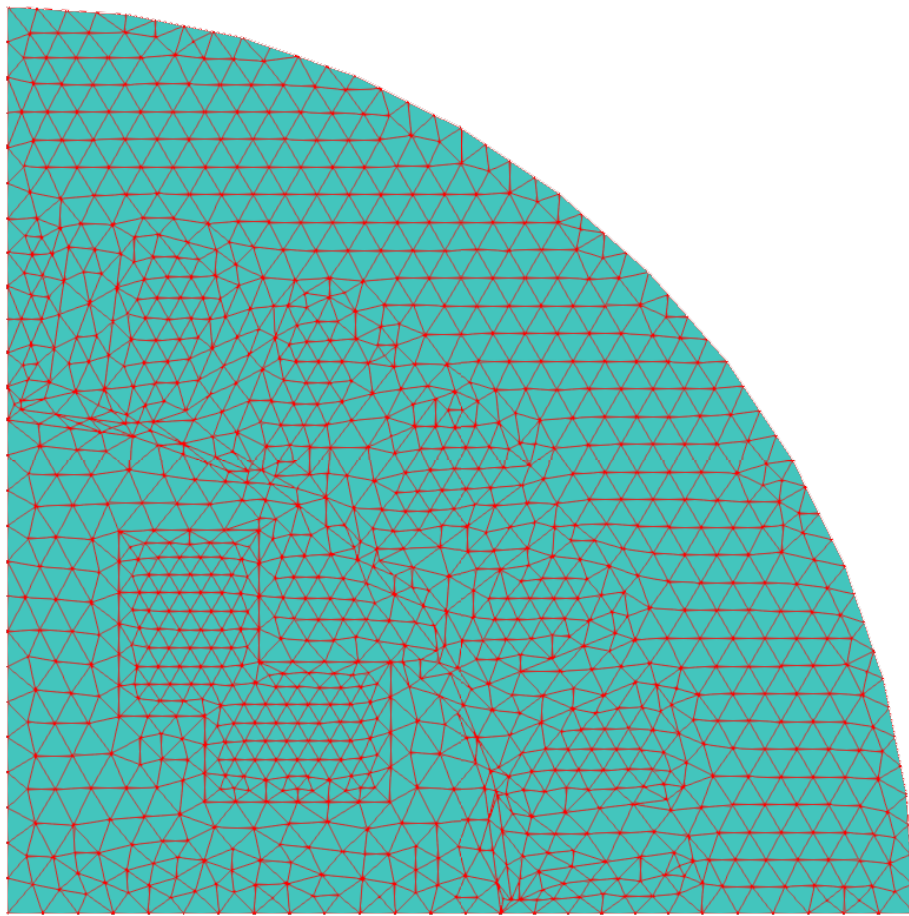
Posledním krokem v *SALOME* je export sítě. Ten uděláme např. pomocí kontextové nabídky vyvolané pravým kliknutím na síť a pak zadáme *Export > DAT file*. *SALOME* umí exportovat síť do mnoha formátů, nicméně `.dat` je ze všech zdaleka nejjednodušší pro další zacházení. Máme tedy připravené tři sítě na konverzi z formátu `DAT` do formátu vhodného pro *adgfem*.

4.3 datToAdgfem

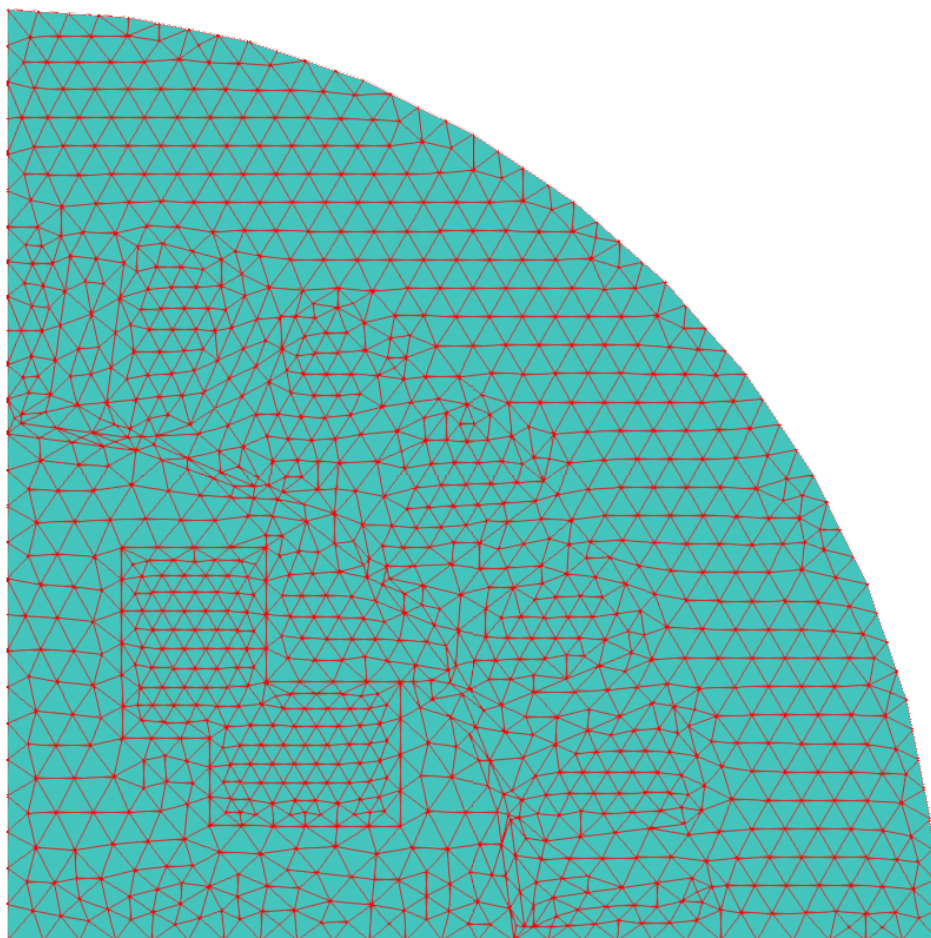
Pro účely konverze sítě z formátu `.dat` jsem napsal program *datToAdgfem*. Tím byl vytvořen důležitý most mezi programem *SALOME*, který je mocným



Obrázek 4.3: Vlevo nahoře: Načtené body ze souboru. Vpravo nahoře: Body rozkopírované pomocí symetrií. Vlevo dole: Body pospojované křivkami. Vpravo dole: Vytvořené oblasti (typ *face*), pro přehlednost ručně obarveno.



Obrázek 4.4: Hrubá síť obsahují 2154 trojúhelníků, později použitá k výpočtu.



Obrázek 4.5: Jemná síť obsahující 7294 trojúhelníků, později také použitá k výpočtu.

nástrojem na přípravu sítí a programem *adgfem*, který doposud ucelený síťovací nástroj postrádal. Program *datToAdgfem* je napsán v jazyce C++. Dokumentace formátu, který *adgfem* očekává, je v (Dolejší, 2005). Před použitím je potřeba jej zkompileovat. Program je tvořen následujícími soubory:

classes.cpp je soubor kde jsou implementovány všechny třídy použité v *datToAdgfem*, tedy kde se odehrává většina skutečné práce, kterou *datToAdgfem* vykonává.

classes.h hlavičkový soubor, kde jsou definovány všechny použité třídy. Všechny použité funkce jsou okomentovány, což by mělo usnadnit případné budoucí úpravy.

main.cpp je soubor, který ovládá třídy vytvořené v **classes.cpp** a v podstatě tvoří uživatelské rozhraní, pokud by uživatel nebyl spokojen s tím, jak program funguje a chtěl by si jej upravit. Soubor je opět okomentovaný.

makefile je standardní soubor obsahující nastavení pro kompilaci na Unix-like systémech.

Kompilaci na Unix-like systému za nás obstará utilita *make*. Stačí výše zmíněné soubory zkopírovat do cílové složky a spustit příkaz *make*:

```
>make
g++ -Wall -c main.cpp
g++ -Wall -c classes.cpp
g++ -Wall -o datToAdgfem main.o classes.o
```

Výsledkem pak je spustitelný soubor *datToAdgfem*. Abychom mohli program spustit, očekává v adresáři, ve kterém je spuštěn, dva soubory:

Parametr	Hodnota
OD	
Algorithm	Segments around Vertex
Hypothesis	Length Near Vertex_1
Length	1
1D	
Algorithm	Wire Discretization
Hypothesis	Local Length_1
Length	0.01; 0.02; 0.04
Precision	1e-7
2D	
Algorithm	Triangle: Mephisto
Hypothesis	Length From Edges_1

Tabulka 4.1: Hodnoty parametrů použité pro síťování v *SALOME*.

mesh.dat je přímo ten soubor, který exportujeme ze *SALOME*.

components je soubor, který slouží k očíslování komponent, na které chceme síť rozdělit. V našem případě má síť pět částí: stator, rotor, vinutí cívek a dvě části, kde je vzduch. Příklad, jak takový soubor **components** může vypadat:

```
1 0.01 0.01
2 0.6 0.6
3 0.4 0.2
4 0.2 0.2
4 0.4 0.4
```

V prvním sloupci je číslo, které chceme komponentě přiřadit, *datToAdgfem* očekává kladné celé číslo. Je vidět, že komponenta číslo 4 se opakuje, tudíž se skládá ze dvou částí. Druhý a třetí sloupec je x-ová a y-ová souřadnice bodu, který leží uvnitř dané komponenty. Komponenty musí být disjunktní, jinak program skončí chybou. Pokud nějaká komponenta není vybrána, dostane automaticky přiřazeno číslo -1. Tuto funkci není nutné využít, stačí soubor ponechat prázdný.

Máme-li k dispozici výše zmíněné soubory, můžeme program spustit. Výstup z programu by měl vypadat takto:

```
>./datToAdgfem
reading file mesh.dat
ordering the triangles
0 triangles reordered
sorting edges
reordering boundary edges
14351 points, 372 boundary edges, 645 internal edges and 28328\
  triangles read
checking mesh integrity
minimum triangles for one point is 2, average is 5.92182
edges oriented properly
reading file components
creating components
looking for neighbours
creating component 1
creating component 2
creating component 3
creating component 4
creating component 4
writing the mesh into file triang.grid
writing internal boundary into file internalBoundary
printing boundary edges into file boundaryEdges
printing internal edges into file internalEdges
printing internal mesh into file internalMesh
printing component 1 into file internalMesh_1
printing component 2 into file internalMesh_2
printing component 3 into file internalMesh_3
printing component 4 into file internalMesh_4
printing internal mesh into file internalMeshVectors
```

Rozeberme si nyní podrobněji, co se při jednotlivých krocích děje:

```
1. reading file mesh.dat
   ordering the triangles
   0 triangles reordered
```

Jsou čtena data ze souboru `mesh.dat`. Trojúhelníky jsou očíslovány proti směru hodinových ručiček, jak *adgfem* očekává. V tomto případě jsou všechny očíslovány správně, ale testováním bysme došli k závěru, že se na to nelze spolehnout.

```
2. sorting edges
   reordering boundary edges
```

Nyní se třídí hrany. Formát `.dat` nerozlišuje mezi vnitřními hranami a hranami na hranici oblasti, takže je *datToAdgfem* roztřídí. Potom je správně zorientuje, aby byl vnitřek oblasti z pohledu hrany vlevo. Nakonec jsou hrany seřazeny tak, aby na sebe navazovaly jedna za druhou. Obě dvě tyto vlastnosti *adgfem* očekává.

```
3. 14351 points, 372 boundary edges, 645 internal edges and 28328\
    triangles read
    checking mesh integrity
    minimum triangles for one point is 2, average is 5.92182
    edges oriented properly
```

Statistiky a ověřování sítě. Ověřuje se, že každý bod náleží alespoň do jednoho trojúhelníku, že každá hrana náleží nejvýše do dvou trojúhelníků, že existují alespoň tři hrany na hranici oblasti. Kontroluje se, zda jsou hrany skutečně správně srovnány jedna za druhou.

```
4. reading file components
   creating components
   looking for neighbours
   creating component 1
   creating component 2
   creating component 3
   creating component 4
   creating component 4
```

Čte se soubor `components`. Pokud obsahuje alespoň jednu komponentu, *datToAdgfem* vybuduje topologii sítě, tj. najde všem trojúhelníkům všechny sousedy. Tato funkce je implementována velmi jednoduchým, ale pravděpodobně neefektivním způsobem, takže má kvadratickou náročnost v počtu trojúhelníků. Nicméně, pro velikosti sítí, které *adgfem* typicky řeší, je to řešení dostatečné. Po vybudování topologie jsou trojúhelníkům přiřazena čísla komponent.

- ```
5. writing the mesh into file triang.grid
 writing internal boundary into file internalBoundary
```

Principiální část výstupu. Je zapsán soubor `triang.grid`, což je samotná síť ve formátu již vhodném pro *adgfem*. Kromě standartních informací obsahuje u každého elementu ještě jeden sloupec navíc, kde je zaznamenáno číslo komponenty. Pak se zapíše soubor `internalBoundary`, který obsahuje vnitřní hrany, také ve formátu připraveném pro *adgfem*.

- ```
6. printing boundary edges into file boundaryEdges
   printing internal edges into file internalEdges
   printing internal mesh into file internalMesh
   printing component 1 into file internalMesh_1
   printing component 2 into file internalMesh_2
   printing component 3 into file internalMesh_3
   printing component 4 into file internalMesh_4
   printing internal mesh into file internalMeshVectors
```

Vedlejší výstup. Zapisují se soubory `boundaryEdges`, `internalEdges`, `internalMesh`, `internalMeshVectors` a pro každou komponentu soubor `internalMesh_n`, kde *n* je číslo komponenty. Tyto soubory jsou určeny pro snadnější vizualizaci a kontrolu výsledné sítě.

4.4 Vizualizace výsledné sítě

Kromě souborů `classes.cpp`, `classes.h`, `main.cpp` a `makefile`, které jsou přímo součástí *datToAdgfem*, jsem napsal ještě jednoduchý skript `mesh.gp` pro *gnuplot*, který využívá soubory `boundaryEdges`, `internalEdges` a `internalMesh`. Díky němu je možné snadno vizuálně zkontrolovat výslednou síť a orientaci hran na hranici. Výsledek můžete vidět na obrázku 4.6.

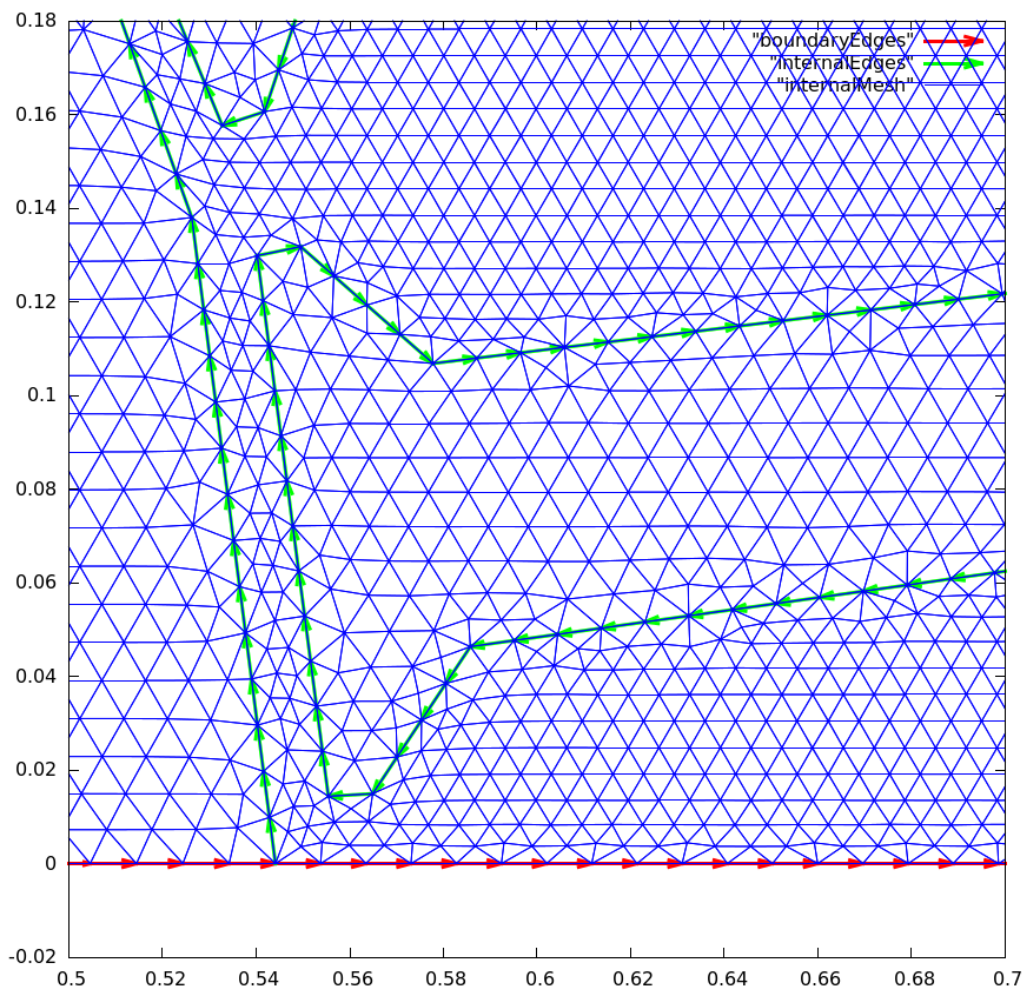
Podobný smysl mají i soubory `internalMesh_n`, které obsahují jen trojúhelníky z určité komponenty a lze je vykreslit (viz 4.7) jediným příkazem v *gnuplotu*:

```
gnuplot> plot "internalMesh_4" with lines ti "komponenta 4"
```

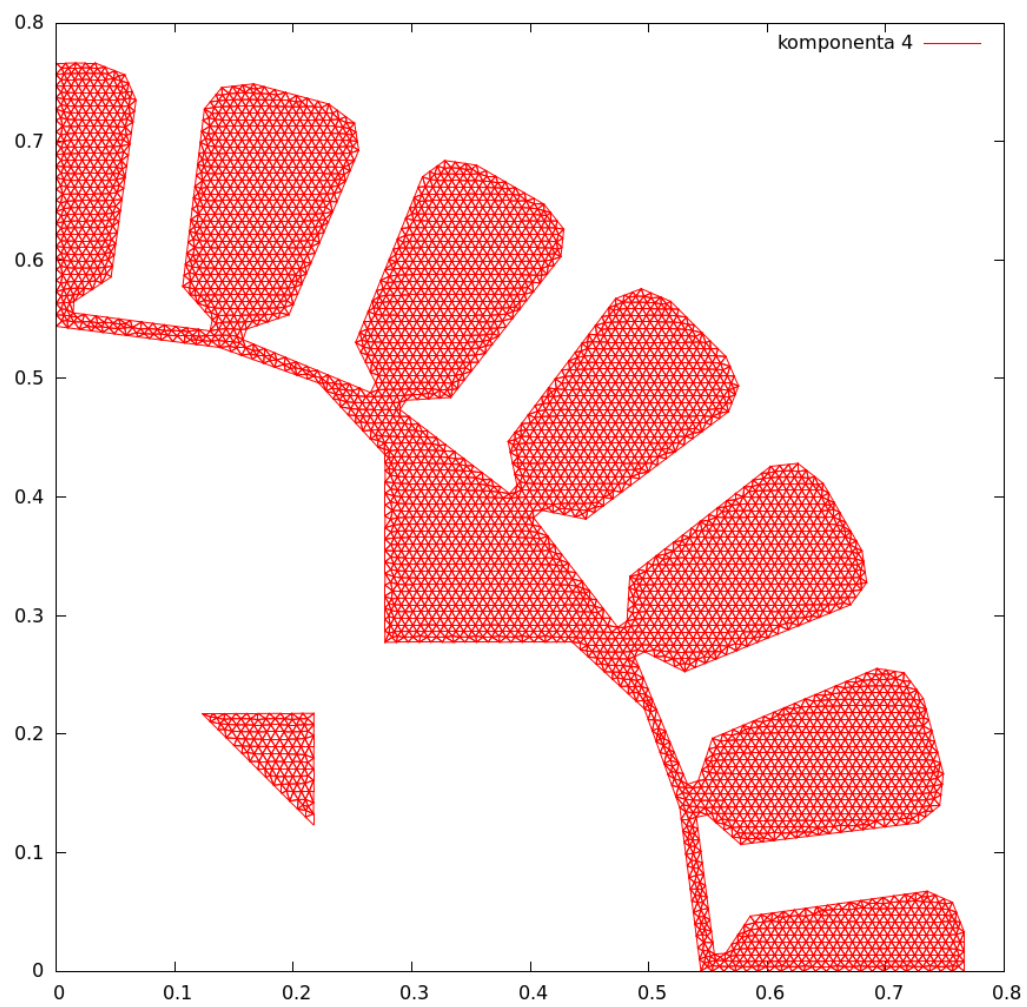
4.5 Omezení

Program *datToAdgfem* má zatím nějaká omezení, na která by uživatel mohl narazit:

- Program *datToAdgfem* zvládá pouze trojúhelníky.
- Hranice sítě musí být jednoduše souvislá, jinak ji *datToAdgfem* nedokáže správně uspořádat.
- Nezvládá křivočaré hranice. Nicméně uživatel se s nimi může vypořádat ručně.



Obrázek 4.6: Obrázek sítě, výřez. Hrany na hranici sítě (červené šipky) jsou správně orientovaná vůči vnitřku sítě.



Obrázek 4.7: Vykreslená čtvrtá komponenta.

- Výroba komponent může být pomalá pro velké sítě (cca nad 100 000).

5. Výpočty v *adgfem*

Veškerou práci s *adgfem*, o které budu v této kapitole mluvit, jsem prováděl s revizí 1485 z 30. 11. 2017, ke které jsou připojeny moje vlastní modifikace, viz příložené soubory. Všechny potřebné soubory k replikování výpočtů jsou zahrnuty v elektronické příloze.

5.1 Magnetické pole

Abych se mohl pustit do výpočtu magnetického pole, bylo potřeba učinit tři zásadní kroky:

1. Bylo nejprve potřeba *adgfem* upravit tak, aby uměl načíst informaci o tom, který trojúhelník patří do které komponenty. To si vyžádalo modifikaci souboru `o_element.f90`.
2. Bylo třeba v souboru `model.f90` vytvořit nový model, který by odpovídal zadání a odkazoval na správnou pravou stranu a správný difusní koeficient. Na tyto změny je pak odkázáno ze souboru `o_scalar.f90`.
3. A konečně, bylo třeba vytvořit funkce pro pravou stranu a pro difusní koeficient, které mají přístup k informaci o své komponentě a podle toho nastaví správnou hodnotu. Tyto změny se projeví především v souboru `problem.f90`

5.2 Nastavení výpočtů

Vzorový soubor s nastavením výpočtů je v příloze práce a kompletní složka s vzorovým výpočtem je příložená v elektronické příloze. Zde uvedu jen vybrané parametry provedených výpočtů. Výpočty byly prováděny buď s aproximací prvního nebo druhého řádu. Nelineární rovnice vzniklé diskretizací (3.4) byly řešeny modifikovanou Newtonovou metodou, lineární systémy rovnic pak metodou GMRES. Jako počáteční podmínka byla použita $u = 0$ v celé oblasti Ω . Konstantu C_w jsem zvolil $C_w = 40$.

Jistým problémem bylo nastavení pravé strany, tj. proudové hustoty f . Původní článek (Glowinski a Marrocco, 1974) totiž sice zmiňuje velikost proudové hustoty, ale protože nejsou známy rozměry alternátoru, dostali bychom při stejné hustotě proudu jiné výsledky. Zvolil jsem proto proudovou hustotu $f = 5 \cdot 10^4 \text{ Am}^{-2}$, protože pak se výsledná magnetická indukce $B \approx 3$ (v maximu) a tudíž je dost velká na to, aby se projevila nelinearita magnetické reluktivity.

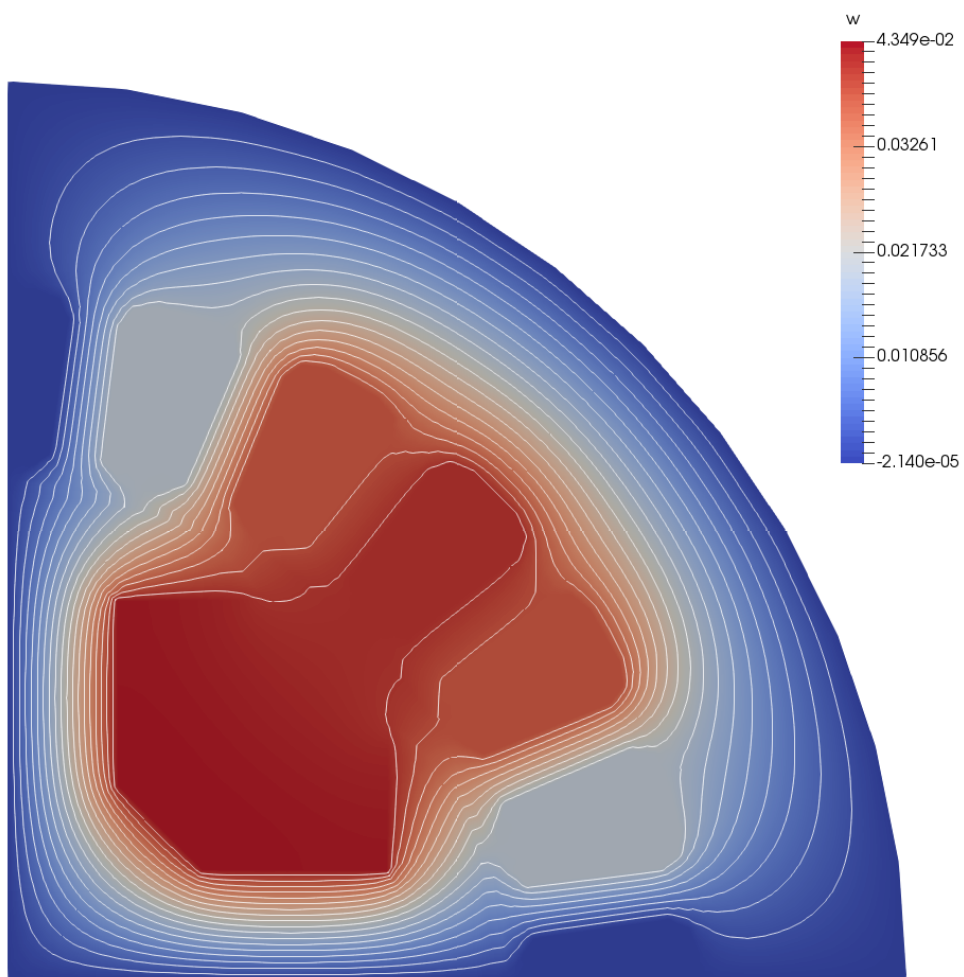
Přestože úloha prezentovaná v této práci je stacionární, *adgfem* je vnitřně stavěn na nestacionární úlohy a stacionárního řešení se dosáhe tím, že se necháme řešení ustálit. Zvolil jsem fixní časový krok 100 s, za diskretizační metodou v čase jsem zvolil opět nespojitou galerkinovu metodu, tentokrát nultého řádu. Zastavovací kritérium bylo, pokud se norma řešení v následujících časových krocích změní o méně než 10^{-8} , výpočet se zastaví.

Výsledky několika variant výpočtu jsou shrnuty v tabulce 5.1. Výpočty byly provedeny na dvou sítích, které můžeme vidět na obrázcích 4.4 (hrubá síť) a 4.5

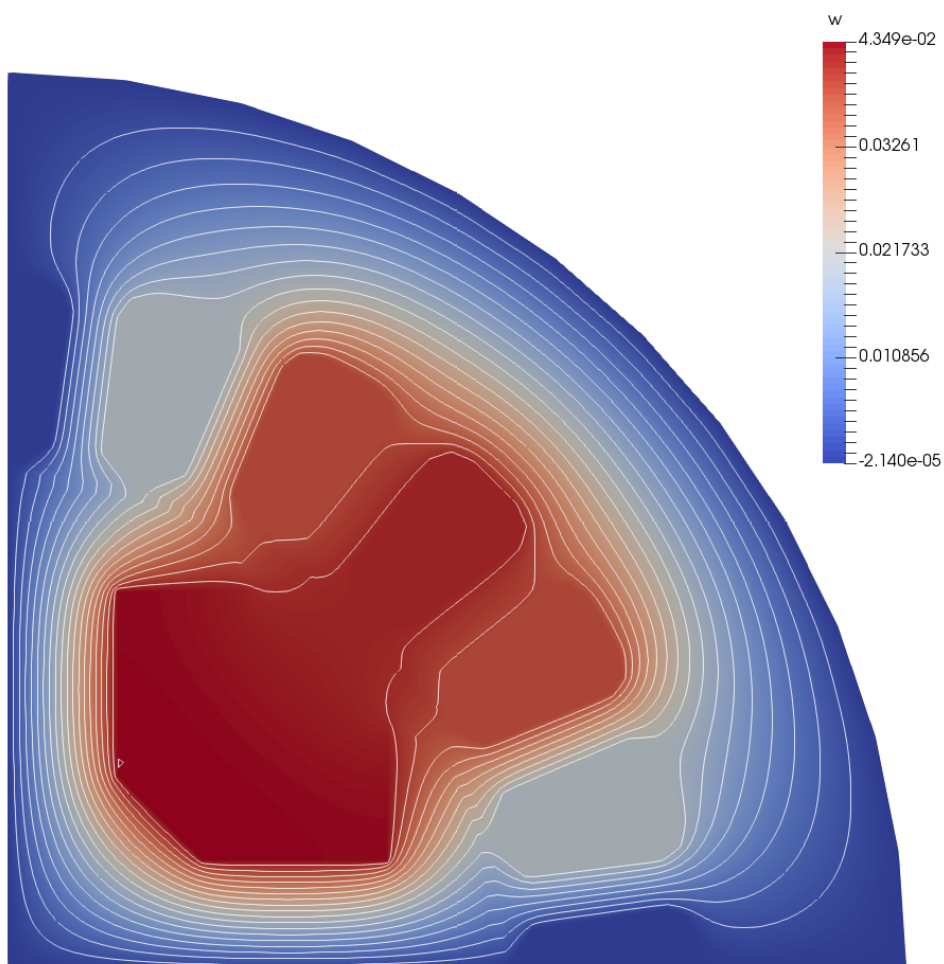
(jemná síť). Výsledky dvou variant výpočtu můžeme vidět na obrázcích 5.1 a 5.2. Srovnáním s obrázkem 4.1 můžeme vidět dobrou kvalitativní shodu řešení. Magnetické pole se chová tak, jak by se dalo v alternátoru očekávat, většina siločar prochází jádru cívek rotoru a k nim blízkým cívkám statoru.

Síť	Aproximace	Stupně volnosti	Počet kroků	Čas výpočtu [h]
hrubá	P_1	6462	80	0.23
hrubá	P_2	12924	80	0.24
jemná	P_1	21882	443	5.12
jemná	P_2	43764	210	5.09

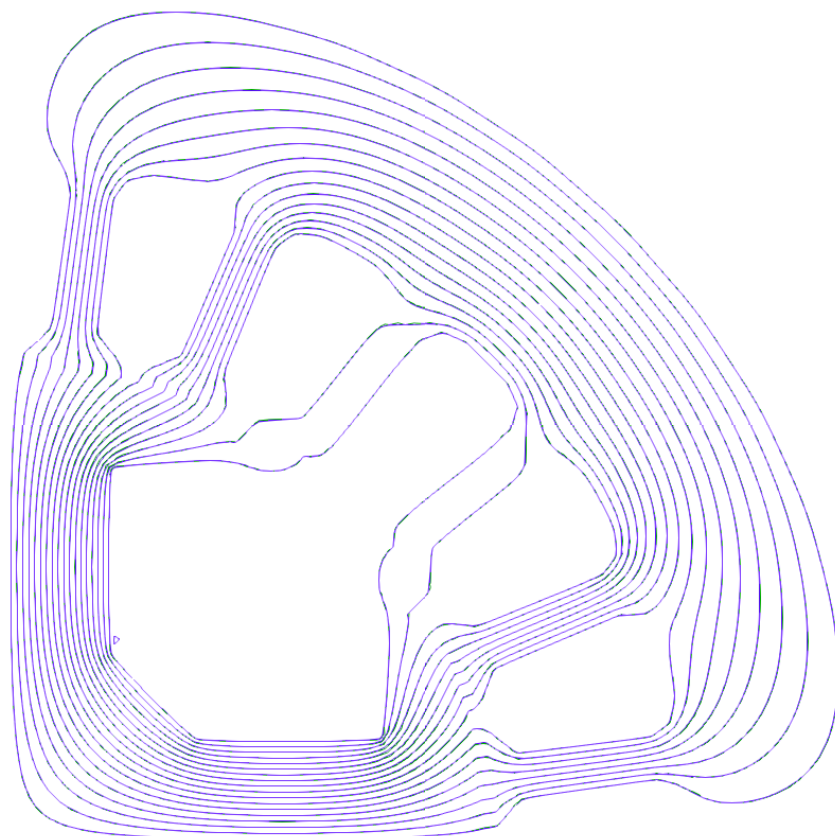
Tabulka 5.1: Porovnání vybraných parametrů provedených výpočtů.



Obrázek 5.1: První výpočet, aproximace polynomy prvního řádu, hrubá síť. Bílé čáry jsou siločáry magnetického pole, barvou je vyvedena hodnota potenciálu.



Obrázek 5.2: Čtvrtý výpočet, aproximace polynomu druhého řádu, jemná síť. Bílé čáry jsou siločáry magnetického pole, barvou je vyvedena hodnota potenciálu. Pozorný čtenář si všimne, že ve srovnání s předchozím výsledkem jsou siločáry hladší, zuby v siločárách už nejsou pouhým okem patrné.



Obrázek 5.3: Srovnání siločar z výpočtu druhým řádem na hrubé síti (zelená) a prvním řádem na jemné síti (modrá). Je vidět, že se velmi přesně překrývají, takže zelená není skoro vidět.

Závěr

V této práci se mi podařilo rozšířit spektrum schopností softwaru *adgfem* o výpočet nelineárního magnetického pole s reluktivitou závislou jak na velikosti magnetické indukce, tak na poloze, obojí fyzikálně realistickým způsobem.

Tuto úlohu jsem otestoval na reálné geometrii alternátoru. Výpočet jsem provedl na dvou různých sítích s aproximací polymomy prvního a druhého řádu. Tyto výsledky se kvalitativně velmi dobře shodují s předlohou, pro detailnější porovnání však chybí podrobnější výsledky i porovnání s reálným měřením.

Pro účel těchto výpočtů jsem pospojováním svobodného softwaru a vlastní aplikace vyvinul síťovací proces, který podstatně rozšiřuje možnosti programu *adgfem* a lze jej využít jak k importu již existujících 2D sítí, tak k podstatnému usnadnění tvorby vlastních nových 2D sítí s realistickou geometrií. Tento proces navíc umožňuje rozdělit síť na podsítě (které v práci nazývám komponenty), které jsem použil na oddělení jednotlivých materiálů. Pro tyto komponenty je navíc automaticky generován soubor s vnitřními hranicemi, které *adgfem* respektuje i při použití adaptace sítě, takže tvar jednotlivých komponent je zachován.

Seznam použité literatury

- Salome. <http://www.salome-platform.org/>. dne: 1. 2. 2017.
- DEUFLHARD, P. (2004). *Newton Methods for Nonlinear Problems*, volume 35. Springer. Springer Series in Computational Mathematics.
- DOLEJŠÍ, V. (2005). Angener user's guide. <http://www.karlin.mff.cuni.cz/~dolejsi/angen/angen3.1.htm>. Version 3.1.
- DOLEJŠÍ, V. a FEISTAUER, M. (2005). Error estimates of the discontinuous galerkin method for nonlinear nonstationary convection-diffusion problems. *Numerical Functional Analysis and Optimization*, **3**, 25–26. ISSN 2709–2733.
- DOLEJŠÍ, V. a M., F. (2015). *Discontinuous Galerkin Method*. Springer International Publishing. ISBN 978-3-319-19266-6.
- DOLEJŠÍ, V., HOLÍK, M. a HOZMAN, J. (2011). Efficient solution strategy for semi-implicit discontinuous galerkin discretization of navier-stokes equations. *Journal of Computational Physics*. ISSN 4176–4200.
- GIRAULT, V. a RAVIART, P.-A. (1986). *Finite Element Methods for Navier-Stokes Equations*. Springer-Verlag. ISBN 978-3-642-64888-5.
- GLOWINSKI, R. a MARROCCO, A. (1974). Analyse numerique du champ magnetique d'un alternateur par elements finis et sur-relaxation ponctuelle non lineaire. *Computer Methods in Applied Mechanics and Engineering*, **3**, 55–85.
- KŘÍŽEK, M. a SEGETH, K. (2001). *Numerické modelování problémů elektrotechniky*. Karolinum, Praha. ISBN 80-246-0287-3.
- ROHATGI, A. Webplotdigitizer. <http://arohatgi.info/WebPlotDigitizer/>. dne: 1. 2. 2017.

Seznam obrázků

4.1	Geometrie alternátoru z Glowinski a Marrocco (1974) použitá jako vzor pro vytváření sítě. Zdroj bohužel nezmiňuje rozměry, pro jednoduchost jsme tedy předpokládali, že průměr alternátoru je 1 m.	19
4.2	Celý alternátor z Glowinski a Marrocco (1974). Na obrázku je popsán stator a rotor (svisle šrafované části). Šedá část popsaná jako SUPP J je vinutí cívek rotoru, kde bude proudit proud budící magnetické pole. Vinutí cívek statoru je zanedbáno. Bílé části jsou vzduch.	20
4.3	Vlevo nahoře: Načtené body ze souboru. Vpravo nahoře: Body rozkopírované pomocí symetrií. Vlevo dole: Body pospojované křivkami. Vpravo dole: Vytvořené oblasti (typ <i>face</i>), pro přehlednost ručně obarveno.	22
4.4	Hrubá síť obsahují 2154 trojúhelníků, později použita k výpočtu.	23
4.5	Jemná síť obsahují 7294 trojúhelníků, později také použita k výpočtu.	24
4.6	Obrázek sítě, výřez. Hrany na hranici sítě (červené šipky) jsou správně orientovaná vůči vnitřku sítě.	29
4.7	Vykreslená čtvrtá komponenta.	30
5.1	První výpočet, aproximace polynomy prvního řádu, hrubá síť. Bílé čáry jsou siločáry magnetického pole, barvou je vyvedena hodnota potenciálu.	34
5.2	Čtvrtý výpočet, aproximace polynomy druhého řádu, jemná síť. Bílé čáry jsou siločáry magnetického pole, barvou je vyvedena hodnota potenciálu. Pozorný čtenář si všimne, že ve srovnání s předchozím výsledkem jsou siločáry hladší, zuby v siločárách už nejsou pouhým okem patrné.	35
5.3	Srovnání siločar z výpočtu druhým řádem na hrubé síti (zelená) a prvním řádem na jemné síti (modrá). Je vidět, že se velmi přesně překrývají, takže zelená není skoro vidět.	36

Přílohy

Souřadnice vrcholů geometrie

Toto je výpis souboru se souřadnicemi získaný z *WebPlotDigitizer*. V prvním sloupci je x-ová souřadnice, ve druhém y-ová.

```
-0.0008725605416075102 -0.00043056036216104765
0.5438473935999029 -0.00036772300538401304
0.54338591605173 0.020503328024061185
0.5415701677902928 0.056044137017311035
0.5377003920105186 0.08256753396358907
0.5307390182773003 0.12038255257332375
0.5228758027996142 0.15030419377411885
0.5115952405109503 0.18418609654844786
0.4985954481408592 0.21807403170902728
0.437013833101381 0.27638609340066833
0.42211383906234373 0.27643837408150695
0.4218574626466923 0.12132863181696296
0.21612594576234231 0.12261453548605394
0.12321211653713157 0.21600693402048077
0.12144462736569922 0.42132322165124614
0.2773184718886555 0.42021225718342314
0.2779548886380961 0.43205483679110335
0.22156565006384102 0.49683512463997404
0.17781576878120248 0.5139097939223605
0.12804104173075478 0.5295955062705413
0.07451266233585732 0.5399360217018165
0.0006068821863576199 0.5441436111116256
```

Skript pro načtení souřadnic do SALOME

Toto je skript v Pythonu použitý k načtení souboru se souřadnicemi do *SALOME*.

```
# -*- coding: utf-8 -*-

###
### This file is generated automatically by SALOME v8.2.0
### with dump python functionality
###

import sys
import salome

salome.salome_init()
theStudy = salome.myStudy

import salome_notebook
```

```

notebook = salome_notebook.NoteBook(theStudy)
sys.path.insert( 0, r'/home/vojta/Meshing/alternator')

###
### GEOM component
###

import GEOM
from salome.geom import geomBuilder
import math
import SALOMEDS

geompy = geomBuilder.New(theStudy)

O = geompy.MakeVertex(0, 0, 0)
OX = geompy.MakeVectorDXDYDZ(1, 0, 0)
OY = geompy.MakeVectorDXDYDZ(0, 1, 0)
OZ = geompy.MakeVectorDXDYDZ(0, 0, 1)
geompy.addToStudy( O, 'O' )
geompy.addToStudy( OX, 'OX' )
geompy.addToStudy( OY, 'OY' )
geompy.addToStudy( OZ, 'OZ' )

#
# Begining of user edited part
#

# open file with coordinates
f = open("points.xy")

# counter of points
n = 0

# for each line in file
for l in f:

    # parse line into variables x and y
    x, y = [ float(v) for v in l.split() ]

    # create new vertex,
    # x, y are coordinates from previous step, z coordinate is 0
    pt = geompy.MakeVertex(x, y, 0)

    # add new vertex to the study
    geompy.addToStudy(pt, "Pt_%s"%(n))
    n += 1
    pass

# update GUI, so user can see the points
import salome

```

```

salome.sg.updateObjBrowser(0)

#
# End of user edited part
#

if salome.sg.hasDesktop():
    salome.sg.updateObjBrowser(True)

```

Inicializační soubor

Nastavení inicializačního souboru si můžete prohlédnout [zde](#):

```

scalar 1.26E-06 79 0.0000E+00
1E+20
1.00E-08 1.00E+05 1.00E+06 1.00E+05
2 'alternator_coarse.grid'
2 'alternator_coarse.prof'
1
test
IIPG 4.000E+01 1
STDG 0 BDF 1
fixed 1.0E+02 adapt 1.000E-01 tRES
RES 1.0000E+00 1.000E-01 -1.00 0
AMAhp 0000 n
500
Newton aRES 1.0000E-03 10 20
none GMRES_ILU 1E-3
0.0000E-00 0
1
1 0 1.0000E+00
2.00 0.00 0.50 0.00
0.0 0.0 0.00 0

```

Samozřejmě, protože výpočet proběhl v několika verzích, některé parametry se měnily.

Příložené soubory

Elektronická příloha obsahuje celkem tři složky. První z nich je `datToAdgfem` spolu se vzorovou sítí ve formátu `.dat` a gnuplotovým skriptem `mesh.gp`. Další dvě jsou `vzorovy_case` kde jsou všechny soubory nutné pro spuštění výpočtů diskutovaných v kapitole 5. Poslední složka obsahuje všechny soubory, které jsem při své práci v *adgfem* změnil oproti revizi 1404.