

Západočeská univerzita v Plzni
Fakulta aplikovaných věd
Katedra matematiky



Diplomová práce

Efektivní numerické metody řešení advektivně-difuzní rovnice

Plzeň, 2014

Jaroslav Tuma

University of West Bohemia
Faculty of Applied Sciences
Department of Mathematics



Diploma Thesis

**Efficient numerical methods for
solution of advection-diffusion
equation**

Plzeň, 2014

Jaroslav Tuma

Prohlášení

Prohlašuji, že jsem diplomovou práci vypracoval samostatně a výhradně s použitím citovaných pramenů.

V Plzni dne 15. 8. 2014

.....
Jaroslav Tuma

Poděkování

Rád bych vyjádřil své díky vedoucímu práce Doc. Ing. Marku Brandnerovi, Ph.D. za velmi trpělivou a vstřícnou pomoc při vytváření této práce. Také bych chtěl poděkovat svým rodičům za podporu při celých mých studiích. Děkuji také své manželce a dceři za morální podporu při psaní této práce.

Abstrakt

Tato práce se zabývá numerickými metodami pro řešení difuzní a advekčně-difuzní rovnice. Problémy toho typu jsou zásadní v mnoha modelech mechaniky tekutin, ale i dalších fyzikálních fenoménů. Pro diskretizaci úlohy jsou použity metoda konečných diferencí a metoda konečných prvků. Zabýváme se iteračními metodami pro řešení soustav lineárních algebraických rovnic vzniklých při použití těchto metod. Na závěr je předvedena metoda výpočtu řešení advekčně-difuzní rovnice založená na převedení rovnice na hyperbolickou soustavu parciálních diferenciálních rovnic.

Klíčová slova

Advekčně-difuzní rovnice, numerické modelování, matematické modelování, parciální diferenciální rovnice, numerické metody, metoda konečných prvků

Title

Efficient numerical methods for solution of advection-diffusion equation

Abstract

This work deals with methods for solution of advection-diffusion equation. Those problems arise in several models of flows, but also in other physical phenomena. For discretization there is used finite differences method and finite elements method. This work also focuses on iterative method for solution of resulting system of linear algebraic equations. Finally, there is shown method for solution of advection-diffusion equation based on transformation into hyperbolic system of partial differential equations.

Keywords

Advection-diffusion equation, numerical modeling, mathematical modeling, partial differential equations, numerical methods, finite elements method

Obsah

1	Difuzní rovnice	2
1.1	Difuzní rovnice v jedné prostorové dimenzi	2
1.1.1	Metoda konečných diferencí	3
1.2	Difuzní rovnice ve dvou prostorových dimenzích	5
1.2.1	Prostory funkcí	6
1.2.2	Slabé řešení	8
1.2.3	Metoda konečných prvků	9
1.3	Metoda sdružených gradientů	13
1.3.1	Předpokládání	15
2	Advekčně difuzní rovnice	17
2.1	Advekčně difuzní rovnice v jedné dimenzi	17
2.1.1	Diskretizace metodou konečných diferencí	18
2.2	Advekčně difuzní rovnice ve dvou dimenzích	19
2.2.1	Slabé řešení	19
2.3	Výpočet s dominantní advekcí	20
2.4	Závislost chyby řešení na tvaru sítě	22
3	Metody řešení soustav lineárně algebraických rovnic	25
3.1	Metody Krylovova podprostoru	25
3.1.1	Báze Krylovova prostoru	26
3.1.2	CG jako projekční metoda	27
3.1.3	Zobecněná metoda minima residuů (GMRES)	28
3.1.4	Metoda bikonjugovaných gradientů	29
3.1.5	Metoda pro antisymetrickou matici	29

3.1.6	Konvergence projekčních metod	30
3.1.7	Předpodmínění	30
3.1.8	Metody v prostředí MATLAB	31
4	Numerické experimenty	32
4.1	Problém dominantní advekce	32
4.2	Počet iterací potřebný k dosažení dané přesnosti	36
4.2.1	Rovnice v jedné dimenzi	36
4.2.2	Rovnice ve dvou dimenzích	37
4.3	Rychlost konvergence GMRES	39
4.4	Shrnutí numerický experimentů	41
5	Hyperbolická soustava parciálních diferenciálních rovnic	42
5.1	Metoda konečných prvků	43
5.2	Rychlost konvergence metody GMRES	45
6	Závěr	46

Úvod

Tato práce se zabývá numerickými metodami pro řešení difuzní a advektivně-difuzní rovnice v jedné a ve dvou prostorových dimenzích. Problémy toho typu jsou zásadní v mnoha modelech mechaniky tekutin, ale i dalších aplikací.

V první kapitole se seznámíme s difuzní rovnicí a s numerickými metodami pro její řešení. V jedné dimenzi použijeme metodu konečných diferencí. Pro dvoudimenzionální případ potom představíme metodu konečných prvků.

Druhá kapitola je potom zaměřená na advektivně-difuzní rovnici. Opět pro její řešení použijeme metodu konečných diferencí v jedné dimenzi a metodu konečných prvků ve dvou dimenzích. Dále zde uvedeme několik aspektů výpočtu. Problematiku řešení advektivně-difuzní rovnice s dominantní advekcí potom budeme dále rozvíjet v dalších kapitolách.

Jedním z klíčových bodů při hledání přibližného řešení použitými metodami je vyřešení soustavy lineárních algebraických rovnic. Ve třetí kapitole tedy uvedeme přehled běžně používaných metod právě ve spojení s metodou konečných prvků.

Čtvrtou kapitolu potom zaměříme na numerické experimenty. Pomocí nich ukážeme efektivitu jednotlivých metod pro řešení soustav lineárně algebraických rovnic, ale také vztah mezi rychlostí konvergence těchto metod a parametry advektivně-difuzní rovnice.

Na závěr předvedeme metodu pro řešení jak difuzní, tak advektivně-difuzní rovnice založenou na hyperbolické soustavě parciálních diferenciálních rovnic. Jaké jsou možné výhody této metody vyvozujeme právě z numerických experimentů provedených v předchozí kapitole.

Kapitola 1

Difuzní rovnice

V této kapitole se zaměříme na numerické metody řešení *stacionární difuzní rovnice se zdrojem* v jedné a ve dvou prostorových dimenzích. Z fyzikálního hlediska tato rovnice popisuje kromě difuze například také průhyb membrány, nebo šíření tepla. Stacionární rovnice modeluje daný problém v ustáleném stavu, čili například ustálené rozložení teploty nebo ustálený průhyb membrány.

Jedná se o základní typ eliptické parciální diferenciální rovnice. Pro řešení difuzní rovnice v jedné dimenzi použijeme metodu konečných diferencí a ve dvou dimenzích metodu konečných prvků. Především metoda konečných prvků je v praxi využívána právě k řešení eliptických problémů. Obě tyto metody budeme později aplikovat na advektivně-difuzní rovnici.

Při psaní této kapitoly jsme čerpali zejména z publikací [5], [1], [11]

1.1 Difuzní rovnice v jedné prostorové dimenzi

V jedné prostorové dimenzi má evoluční difuzní rovnice tvar

$$u_t - \varepsilon u_{xx} = f(x),$$

kde $u(x, t)$ ¹ je hledaná funkce a $\varepsilon \in \mathbb{R}$ značí kladný difuzní parametr. Funkci $f(x)$ nazýváme *zdrojový člen*. Často je zapotřebí nalézt pouze stacionární řešení. Tím myslíme takovou funkci $u(x)$ vyhovující vztahu

$$-\varepsilon u_{xx} = f(x). \tag{1.1}$$

Stacionární rovnice popisuje chování modelovaného problému v ustáleném stavu. Jedná se o obyčejnou diferenciální rovnici druhého řádu. Aby měla rovnice právě jedno řešení,

¹ u_x značí derivaci $\frac{\partial u(x,t)}{\partial x}$ a u_{xx} potom druhou derivaci $\frac{\partial^2 u(x,t)}{\partial x^2}$

doplníme okrajové podmínky. Standardně rozlišujeme několik základních typů okrajových podmínek:

Dirichletovy okrajové podmínky

$$u(a) = g_1, u(b) = g_2$$

Neumannovy okrajové podmínky²

$$u_x(a) = g_1, u_x(b) = g_2$$

Newtonovy okrajové podmínky

$$u_x(a) + u(a) = g_1, u_x(b) + u(b) = g_2$$

Pro naše účely však není tvar okrajové podmínky podstatný. My se tedy omezíme pouze na homogenní Dirichletovy okrajové podmínky

$$u(0) = 0, u(1) = 0. \tag{1.2}$$

1.1.1 Metoda konečných diferencí

Metodou konečných diferencí *aproximujeme diferenciální rovnici* tak, že derivace nahradíme diferencemi. Pokud chceme takto řešit okrajovou úlohu (1.1, 1.2) nejprve rozdělíme interval $\langle 0, 1 \rangle$ na $N + 1$ částí pomocí bodů $x_0, x_1, x_2, \dots, x_N$ tak, aby platilo

$$0 = x_0 < x_1 < x_2 < \dots < x_N = 1.$$

Množinu $\{x_0, x_1, x_2, \dots, x_N\}$ nazveme síť, vzdálenost $h_k = x_{k-1} - x_k$ krok sítě a body x_k uzly sítě.

První derivaci nahradíme některou z následujících diferencí

$$\frac{u(x+h) - u(x)}{h} \tag{1.3}$$

$$\frac{du(x)}{dx} \approx \frac{u(x) - u(x-h)}{h} \tag{1.4}$$

$$\frac{u(x+h) - u(x-h)}{2h}, \tag{1.5}$$

² $u_x(x_0)$ značí derivaci v bodě x_0 $\frac{du(x_0)}{dx}$

kteře snadno odvodíme například pomocí Taylorova polynomu. Tyto vztahy se nazývají dopředná diference (1.4), zpětná diference (1.5) a centřální diference (1.5). Dřuhou derivaci nahradíme podle předpisu

$$\frac{d^2u(x)}{dx^2} \approx \frac{u(x+h) - 2u(x) + u(x-h)}{h^2} \quad (1.6)$$

Mějme konstantní krok sítě $h_k = h$, takovou sít' označíme jako *ekvidistantní sít'*. Označíme $U_i = u(x_i)$, potom po nahrazení první derivace centřální diferencí (1.5) a druhé derivace vzorcem (1.6) musí být splněna v každém bodě rovnost

$$-\varepsilon \frac{U_{k+1} - 2U_k + U_{k-1}}{h^2} = f(x_k).$$

Takto získáme $N - 1$ rovnic. Díky okrajové podmínce známe hodnoty U_0 a U_N , máme tedy také $N - 1$ neznámých. Pro jednoduchost uvažujme homogenní okrajové podmínky

$$u(0) = u(1) = 0. \quad (1.7)$$

Přibližné řešení okrajové úlohy (2.1), (1.7) v uzlových bodech tedy získáme vyřešením soustavy rovnic

$$\begin{aligned} 2\varepsilon U_1 - \varepsilon U_2 &= h^2 f(x_1) \\ -\varepsilon U_1 + 2\varepsilon U_2 - \varepsilon U_3 &= h^2 f(x_2) \\ &\vdots \\ -\varepsilon U_{N-4} + 2\varepsilon U_{N-3} - \varepsilon U_{N-2} &= h^2 f(x_{N-2}) \\ 2\varepsilon U_{N-2} - \varepsilon U_{N-1} &= h^2 f(x_{N-1}) \end{aligned}$$

Tu můžeme přepsat do maticového tvaru

$$\mathbf{AU} = \mathbf{F}, \quad (1.8)$$

kde

$$\mathbf{F} = \begin{bmatrix} h^2 f(x_1) \\ h^2 f(x_2) \\ \vdots \\ h^2 f(x_{N-1}) \end{bmatrix},$$

je vektor pravé strany,

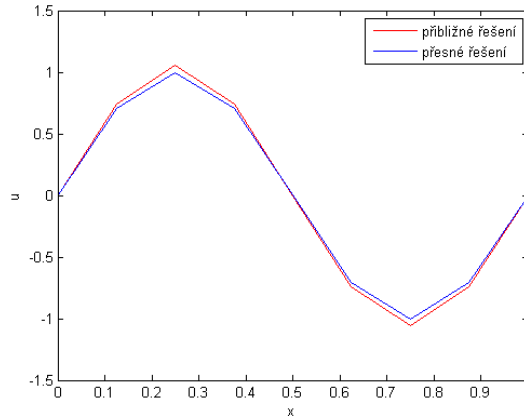
$$\mathbf{U} = \begin{bmatrix} U_1 \\ U_2 \\ \vdots \\ U_{N-1} \end{bmatrix},$$

je vektor hledaných hodnot přibližného řešení v uzlových bodech a

$$\mathbf{A} = \begin{bmatrix} 2\varepsilon & -\varepsilon & 0 & 0 & 0 & 0 \\ -\varepsilon & 2\varepsilon & -\varepsilon & 0 & 0 & 0 \\ 0 & -\varepsilon & 2\varepsilon & \ddots & 0 & 0 \\ 0 & 0 & \ddots & \ddots & -\varepsilon & 0 \\ 0 & 0 & 0 & -\varepsilon & 2\varepsilon & -\varepsilon \\ 0 & 0 & 0 & 0 & -\varepsilon & 2\varepsilon \end{bmatrix}$$

je čtvercová matice velikosti $N - 1$.

Matice \mathbf{A} je zřejmě symetrická a také pozitivně definitní. Soustavu (1.8) můžeme tedy vyřešit například metodou sdružených gradientů. Tu popíšeme na konci této kapitoly.



Obrázek 1.1: Příklad řešení pro $f(x) = 4\pi^2\varepsilon \sin(2\pi x) + 2\pi\sigma \cos(2\pi x)$, $\Delta x = 0.125$

1.2 Difuzní rovnice ve dvou prostorových dimenzích

Evoluční difuzní rovnice ve dvou prostorových dimenzích má tvar

$$u_t - \varepsilon \Delta u = f(x, y),$$

kde $\Delta u = u_{xx} + u_{yy}$ je Laplaceův operátor. Obdobně jako v případě rovnice v jedné dimenzi $u(x, y, t)$ značí hledanou funkci a $f(x, y)$ zdrojový člen. Také tentokrát budeme hledat stacionární řešení. To znamená vyřešit parciální diferenciální rovnici druhého řádu

$$-\varepsilon \Delta u = f(x, y). \quad (1.9)$$

Doplníme okrajové podmínky. Tentokrát jí však neřešíme na intervalu, ale na omezené oblasti $\Omega \in \mathbb{R}^2$. Její hranici budeme značit $\partial\Omega$. Podobně jako v jednodimenzionálním případě rozlišujeme běžně tři druhy okrajových podmínek:

Dirichletovy okrajové podmínky:

$$u(x, y) = g(x, y) \quad \forall (x, y) \in \partial\Omega$$

Neumannovy okrajové podmínky:

$$\frac{\partial u(x, y)}{\partial n} = g(x, y) \quad \forall (x, y) \in \partial\Omega,$$

kde $\frac{\partial u(x, y)}{\partial n}$ značí derivaci ve směru vnější normály k $\partial\Omega$

Newtonovy okrajové podmínky:

$$Au(x, y) + B \frac{\partial u(x, y)}{\partial n} = g(x, y) \quad \forall (x, y) \in \partial\Omega,$$

kde A, B jsou dané nemulové konstanty

Opět se však omezíme pouze na homogenní Dirichletovy okrajové podmínky

$$u(x, y) = 0 \quad \forall (x, y) \in \partial\Omega \tag{1.10}$$

Okrajovou úlohu (1.9, 1.10) budeme řešit metodou konečných prvků. Než začneme s popisem této metody musíme však zavést několik pojmů. Tím nám poslouží následující kapitoly.

1.2.1 Prostory funkcí

Mějme otevřenou omezenou množinu $\Omega \in \mathbb{R}^N$. Její hranici označíme $\partial\Omega$ a $\bar{\Omega} = \Omega \cup \partial\Omega$ je uzávěr množiny.

Prostor všech funkcí $u = u(\mathbf{x})$ definovaných a spojitých na oblasti Ω označíme $C(\Omega)$. $C^k(\bar{\Omega})$ potom značí prostor všech spojitých funkcí, které mají na Ω také spojitou k-tou derivaci. Dále zavedeme

$$\|u\|_{C(\Omega)} = \max_{\mathbf{x} \in \Omega} |u(\mathbf{x})|$$

Prostor všech funkcí u , pro něž je

$$\int_{\Omega} u(\mathbf{x})^p dx$$

konečný budeme značit $L^p(\Omega)$ a nazveme *prostorem integrovatelných funkcí*. Prostor *lokálně integrovatelných funkcí* L^p_{loc} zavedeme jako množinu všech funkcí u měřitelných na Ω , pro které je

$$\int_{\Omega^*} u(\mathbf{x})^p dx$$

konečný pro každou omezenou oblast $\Omega^* \subset \Omega$.

Na prostoru $L^p(\Omega)$ definujeme normu

$$\|u\|_{L^p} = \left(\int_{\Omega} |u(\mathbf{x})|^p d\mathbf{x} \right)^{\frac{1}{p}}$$

Speciálně pro L^2 definujeme skalární součin

$$(u, v) = \int_{\Omega} u(\mathbf{x})v(\mathbf{x})d\mathbf{x}$$

a normu

$$\|u\|_{L^2} = (u, u)^{\frac{1}{2}} = \left(\int_{\Omega} u(\mathbf{x})^2 d\mathbf{x} \right)^{\frac{1}{2}}$$

Zavedeme *multiindex* jako vektor $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \dots, \alpha_n)$, $|\boldsymbol{\alpha}| = \sum_{i=1}^n \alpha_i$, který využijeme k zápisu derivace

$$D^{\boldsymbol{\alpha}}u = \frac{\partial^{|\boldsymbol{\alpha}|}u}{\partial x_1^{\alpha_1} \partial x_2^{\alpha_2} \partial x_N^{\alpha_N}}.$$

Tento zápis nezohledňuje pořadí derivování. Pokud však všechny derivace existují a jsou spojitě, potom na pořadí derivování nezáleží. Budeme tedy jejich existenci a spojitost předpokládat.

Uvažujeme funkci $u \in L^p_{loc}$, funkci w nazveme *zobecněnou derivací* vzhledem k multiindexu $\boldsymbol{\alpha}$, pokud splňuje rovnost

$$\int_{\Omega} u(\mathbf{x})D^{\boldsymbol{\alpha}}\phi(\mathbf{x})dx = (-1)^{|\boldsymbol{\alpha}|} \int_{\Omega} w(\mathbf{x})\phi(\mathbf{x})dx$$

pro všechny $\phi \in C_0^\infty$. Budeme ji značit $D^{\boldsymbol{\alpha}}u$.

Sobolevův prostor $W^{k,p}(\Omega)$ je normovaný lineární prostor všech funkcí $u \in L^p$ jejichž zobecněná derivace $D^\alpha u$ patří do L^p pro všechna α taková, že $|\alpha| \leq k$. Na Sobolevově prostoru definujeme normu

$$\|u\|_{k,p} = \sum_{|\alpha| \leq k} \|D^\alpha u\|_p$$

1.2.2 Slabé řešení

Klasickým řešením okrajové úlohy (1.9, 1.10) je taková funkce $u(x, y)$, která má na Ω spojitě druhé parciální derivace a platí (1.9) na Ω , (1.10) na $\partial\Omega$. To nám však nedovoluje postihnout případ, kdy hledaná funkce nemá v není v některých bodech Ω spojitá. V takovém případě klasické řešení neexistuje a budeme hledat takzvané slabé řešení. Nejprve zavedeme prostor testovacích funkcí

$$V = \{v \in W^{1,2}(\Omega) : v(\partial\Omega) = 0\}.$$

Dále zavedeme prostor přípustných funkcí, který se však díky homogenním Dirichletovým okrajovým podmínkám shoduje s prostorem testovacích funkcí

$$V_g = \{u \in W^{1,2}(\Omega) : u(\partial\Omega) = 0\}.$$

Místo toho, aby řešení u splňovalo v každém bodě rovnost

$$-\varepsilon \Delta u = f$$

budeme požadovat aby byla splněna integrální rovnost

$$\iint_{\Omega} -\varepsilon \Delta u v d\Omega = \iint_{\Omega} f v d\Omega$$

pro všechna $v \in V$. Nyní použijeme Greenovu větu

$$\iint_{\Omega} \varepsilon \nabla u \nabla v d\Omega - \iint_{\Omega} \varepsilon \frac{\partial u}{\partial n} \nabla v dS = \iint_{\Omega} f v d\Omega$$

Díky okrajovým podmínkám (1.10) zřejmě platí $\iint_{\Omega} \varepsilon \frac{\partial u}{\partial n} \nabla v dS = 0$ a tak získáme tvar

$$\iint_{\Omega} \varepsilon \nabla u \nabla v d\Omega = \iint_{\Omega} f v d\Omega \tag{1.11}$$

Funkci $u(x, y) \in V_g$, která splňuje (1.11) nazveme *slabým řešením* úlohy (1.9), (1.10).

Definujeme bilineární formu

$$a(u, v) = - \iint_{\Omega} \varepsilon \nabla u \nabla v d\Omega \quad (1.12)$$

a funkcional

$$F(v) = \iint f v d\Omega. \quad (1.13)$$

Místo úlohy (1.9, 1.10) tedy hledáme $u \in V_g$ takové, aby platilo

$$a(u, v) = F(v) \quad \forall v \in V.$$

Poznamenejme že bilineární forma $a(u, v)$ je symetrická, neboť zřejmě platí rovnost $a(u, v) = a(v, u)$

1.2.3 Metoda konečných prvků

Metoda konečných prvků, dále jen MKP patří mezi *Galerkinovy metody*. Ty spočívají v tom, že hledáme řešení jako lineární kombinaci N *bázových funkcí* v_i

$$u_h = \sum_{i=1}^N c_i v_i.$$

Přibližným Galerkinovým řešením tedy rozumíme funkci u_h , pro kterou platí

$$a(u_h, v_h) = F(v_h) \quad \forall v_h \in V$$

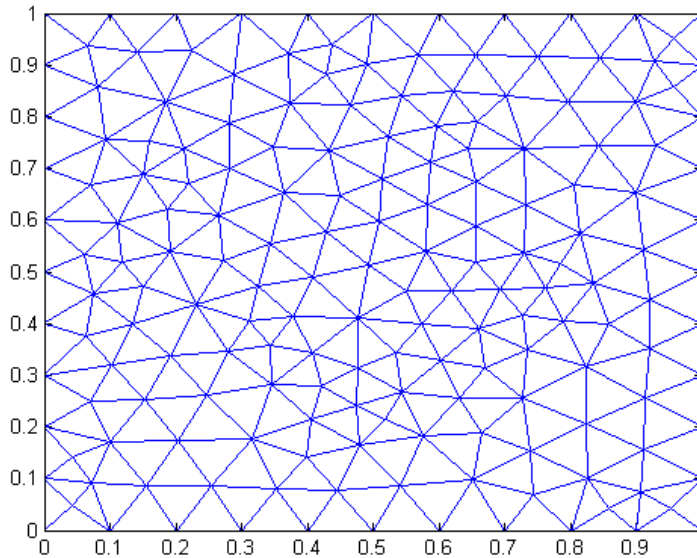
MKP je metoda se speciální volbou bázových funkcí. Nejprve však musíme množinu vyjádřit jako sjednocení konečných podmnožin. Ty nazveme *konečné prvky*. Pro dvou-dimenzionální úlohy se většinou používají buď trojúhelníky, nebo čtyřúhelníky. Poté na každém konečném prvku zkonstruujeme polynomiální bázové funkce. Tyto bázové funkce jsou nenulové pouze na konečném prvku, na zbytku Ω jsou nulové.

Triangulace

Pokud pro diskretizaci Ω použijeme konečné prvky typu trojúhelník, nazývá se tato *triangulací*. Předpokládejme, že hranicí oblasti Ω je polygon. Množinu trojúhelníků

$$\tau_h = \{T_1, T_2, \dots, T_N\}$$

nazveme přípustnou, budou-li splněny podmínky



Obrázek 1.2: Příklad triangulace

1. $\Omega = \bigcup_{s=1}^N T_s$
2. Pro dva různé trojúhelníky T_s, T_r platí $T_s \cap T_r = \emptyset$
3. Pro všechny trojúhelníky T_s platí, že každá strana trojúhelníka je buď součástí hranice $\partial\Omega$, ne stranou jiného trojúhelníka.

Takovou triangulaci nazveme síť. Pro každý trojúhelník definujeme h_s = délka nejdelší strany T_s . Každou síť potom můžeme charakterizovat pomocí jejího rozlišení

$$h = \max_{T_s \in \tau_h} h_s$$

Síť s nižším rozlišením potom nazveme jemnější síť a naopak síť s vyšším rozlišením nazveme hrubší.

Bázové funkce

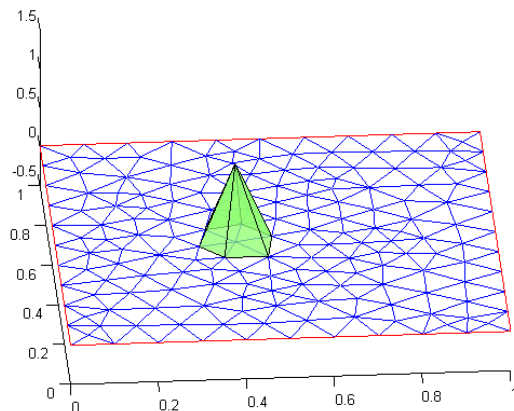
Označíme vrcholy trojúhelníka triangulace $M_1 M_2$ a M_3 . Každý uzel triangulace má *globální souřadnice*, které určují jeho zařazení v rámci všech uzlů. Má ale také takzvané *lokální souřadnice*, což jsou souřadnice v rámci daného elementu.

Nad každým uzlem triangulace M definujeme po částech lineární bázové funkce typu

$$N^s(x, y) = a^s + b^s x + c^s y,$$

kteře musí splňovat interpolační podmínky

$$v_n(M_n) = 1, v_n(M_m) = 0 \forall m \neq n.$$



Obrázek 1.3: Bázová funkce

Všechny bázové funkce jsou tedy nenulové pouze nad trojúhelníky, jejichž vrcholem je M_n . Tyto trojúhelníky tedy tvoří nosič bázové funkce.

Takové bázové funkce jsou spojité na Ω a tvoří lineárně nezávislý systém. Označíme ho tedy $V^h = v_1, v_2, \dots, v^N$. Platí, že $\dim(V^h) = N$ a $V^h \subset W^{1,2}$, neboť každá bázová funkce je spojitá a její derivace existuje skoro všude na Ω .

Pokud označíme souřadnice vrcholů trojúhelníka $M_1 = (x_1^s, y_1^s)$, $M_2 = (x_2^s, y_2^s)$ a $M_3 = (x_3^s, y_3^s)$ restrikce bázových funkcí na trojúhelníku označíme

$$\begin{aligned} N_1^s(x, y) &= a_1^s + b_1^s x + c_1^s y \\ N_2^s(x, y) &= a_2^s + b_2^s x + c_2^s y \\ N_3^s(x, y) &= a_3^s + b_3^s x + c_3^s y \end{aligned}$$

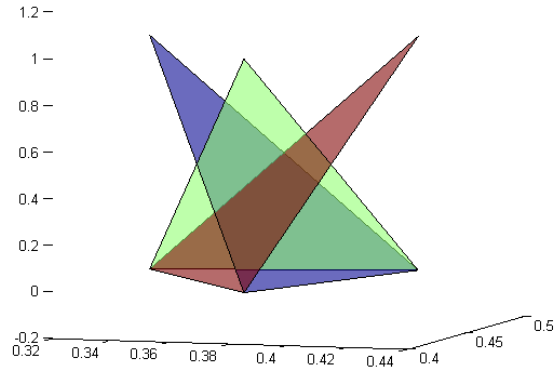
Ty jsou jednoznačně určeny interpolačními podmínkami

$$N_i^s(M_j^s) = \begin{cases} 1 & \text{pro } i = j \\ 0 & \text{pro } i \neq j \end{cases}$$

Odtud získáme koeficienty

$$\begin{aligned} a_1^s &= \frac{1}{D^s}(x_2^s y_3^s), & b_1^s &= \frac{1}{D^s}(y_2^s - y_3^s), & c_1^s &= \frac{1}{D^s}(x_3^s - x_2^s), \\ a_2^s &= \frac{1}{D^s}(x_2^s y_3^s), & b_2^s &= \frac{1}{D^s}(y_2^s - y_3^s), & c_2^s &= \frac{1}{D^s}(x_3^s - x_2^s), \\ a_3^s &= \frac{1}{D^s}(x_2^s y_3^s), & b_3^s &= \frac{1}{D^s}(y_2^s - y_3^s), & c_3^s &= \frac{1}{D^s}(x_3^s - x_2^s), \end{aligned}$$

$$D^s = \det \begin{pmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{pmatrix}$$



Obrázek 1.4: Restrikce bázových funkcí na jednom elementu

je dvojnásobek obsahu trojúhelníka.

Sestavení diskrétní úlohy

Každou funkci $u_h \in V$ můžeme tedy vyjádřit ve tvaru lineární kombinace

$$u_h = \sum_{i=1}^N U_i v_i$$

Hledané uzlové parametry $U_i = u_h(M_i) \approx u(M_i)$ nalezneme řešením soustavy lineárních algebraických rovnic

$$\mathbf{AU} = \mathbf{F} \tag{1.14}$$

Kde $\mathbf{U} = [U_1, U_2, \dots, U_N]$ a $\mathbf{F} = [F_1, F_2, \dots, F_N]$,

$$F_i = F(v_i) = \iint_{\Omega} f(x, y) v_i(x, y) dx dy$$

$$A_{ij} = a(v_i, v_j) = - \iint_{\Omega} (v_i - v_{ix}) v_{jx} + (v_i - v_{iy}) v_{jy} d\Omega$$

Matici A nazveme *globální maticí tuhosti*. Tu sestavíme tak, že pro každý konečný prvek nejprve sestavíme *lokální matici tuhosti*

$$A = \begin{bmatrix} a_{11}^s & a_{12}^s & a_{13}^s \\ a_{21}^s & a_{22}^s & a_{23}^s \\ a_{31}^s & a_{32}^s & a_{33}^s \end{bmatrix}$$

poté zařadíme do globální matice tuhosti

$$A(k, l) = A(k, l) + A^s(i, j)$$

Obdobně budeme postupovat i pro *vektor zatížení* F . Lokální vektor zatížení $F^s = \iint_{\Omega} f(M_i)v_i d\Omega$ zařadíme do globálního vektoru zatížení

$$F(k) = F(k) + F^s(i)$$

Pro výpočet integrálů je vhodné transformovat trojúhelník T_s na referenční trojúhelník T_0 o vrcholech $(0, 1), (0, 0), (1, 0)$. To zajistí následující transformace

$$\begin{aligned} x' &= x_1^s + (x_2^s - x_1^s)\xi + (x_3^s - x_1^s)\eta \\ y' &= y_1^s + (y_2^s - y_1^s)\xi + (y_3^s - y_1^s)\eta \end{aligned}$$

Pro integraci potom platí

$$\int_{T_s} \phi(x, y) dx dy = \int_{T_0} D^s \phi(x', y') dx' dy'$$

Takové integrály potom snadno můžeme přibližně vypočítat pomocí Gaussových kvadraturních vzorců ve tvaru

$$\int f(x, y) dx \approx \sum_{i=1}^n W_i f(r_i, s_i)$$

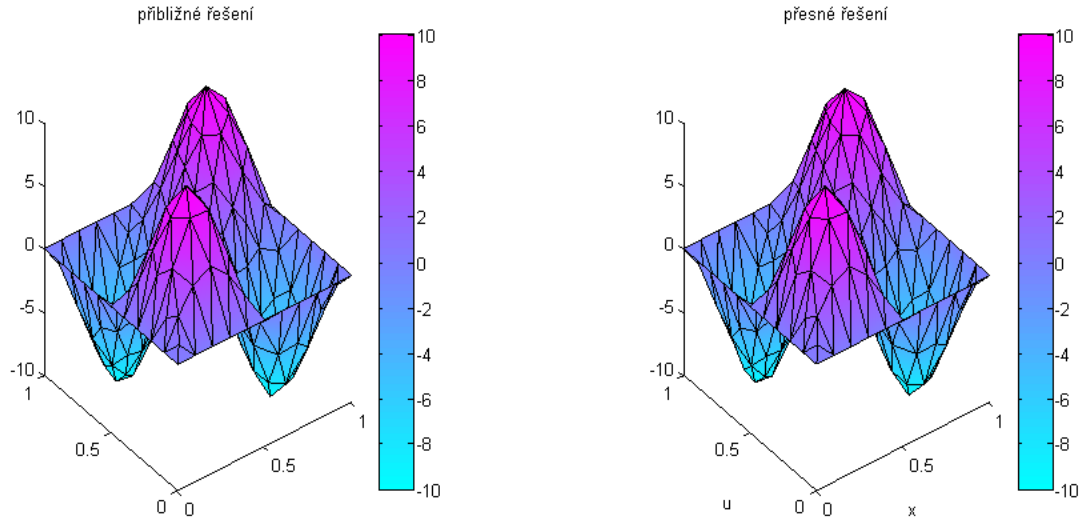
Zde r_i, s_i jsou souřadnice bodů a W_i je potom jejich váha. Více o tomto tématu lze nalézt například v [1]. Já zde nyní pouze uvedu příklad těchto hodnot

$$\begin{aligned} \mathbf{W} &= \left[\frac{1}{6}, \frac{1}{6}, \frac{1}{6} \right] \\ \mathbf{r} &= \left[\frac{1}{6}, \frac{2}{3}, \frac{1}{6} \right] \\ \mathbf{s} &= \left[\frac{1}{6}, \frac{1}{6}, \frac{2}{3} \right] \end{aligned}$$

Abychom získali hledanou aproximaci řešení, musíme vyřešit soustavu lineárních algebraických rovnic (1.14). Matice tuhosti je symetrická pozitivně definitní, což vyplývá z vlastností bilineární formy (1.12). Můžeme použít opět například metodu sdružených gradientů.

1.3 Metoda sdružených gradientů

Jednou z nejdůležitějších iteračních metod pro řešení soustav lineárně algebraických rovnic je *metoda sdružených gradientů*, dále také zkráceně CG. Předpokládejme $\mathbf{A} \in \mathbb{R}^{n \times n}$ symetrickou pozitivně definitní čtvercovou matici a vektor $\mathbf{b} \in \mathbb{R}^n$. Řešení úlohy



Obrázek 1.5: Příklad řešení

$$\mathbf{Ax} = \mathbf{b}$$

je potom ekvivalentní s úlohou nalezení minima funkcionálu

$$F(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T \mathbf{Ax} - \mathbf{x}^T \mathbf{b},$$

neboť $\nabla F(\mathbf{x}) = \mathbf{Ax} - \mathbf{b}$ a $F(\mathbf{x})$ nabývá globálního minima právě pro \mathbf{x} , které splňuje $\nabla F(\mathbf{x}) = 0$. Definujeme *energetickou normu*, nebo-li *\mathbf{A} -normu* vektoru \mathbf{z} následovně

$$\|\mathbf{z}\|_{\mathbf{A}}^2 = \mathbf{z}^T \mathbf{Az}.$$

Uvažujeme k -tou aproximaci \mathbf{x}_k . Platí tedy

$$F(\mathbf{x}_k) = \frac{1}{2} (\mathbf{x} - \mathbf{x}_k)^T \mathbf{A} (\mathbf{x} - \mathbf{x}_k) - \frac{1}{2} \mathbf{x}^T \mathbf{Ax} = \frac{1}{2} \|\mathbf{x} - \mathbf{x}_k\|_{\mathbf{A}}^2 - \frac{1}{2} \|\mathbf{x}\|_{\mathbf{A}}^2.$$

Z toho vyplývá, že úloha nalezení minima funkcionálu F je ekvivalentní s minimalizací \mathbf{A} -normy chyby $\|\mathbf{x} - \mathbf{x}_k\|_{\mathbf{A}}$.

Novou aproximaci vektoru \mathbf{x}_k získáme z předchozí pomocí předpisu

$$\mathbf{x}_k = \mathbf{x}_{k-1} + \gamma_{k-1} \mathbf{p}_{k-1}.$$

Vektor \mathbf{p}_{k-1} zde představuje směrový vektor. Skalár γ_{k-1} určíme tak, aby vektor \mathbf{x}_k byl na přímce $\mathbf{z}(\gamma) = \mathbf{x}_{k-1} + \gamma \mathbf{p}_{k-1}$ bodem minima $F(\mathbf{z}(\gamma))$. Tohoto minima je dosaženo pro

$$\gamma_{k-1} = \frac{\mathbf{p}_{k-1}^T \mathbf{r}_{k-1}}{\mathbf{p}_{k-1}^T \mathbf{A} \mathbf{p}_{k-1}}$$

Volbu směrů \mathbf{p}_k provedeme tak aby byly \mathbf{A} -ortogonální, čili aby platilo

$$\mathbf{p}_i^T \mathbf{A} \mathbf{p}_j = 0 \quad \forall i \neq j$$

Metoda CG volí nové směry jako lineární kombinaci $\mathbf{p}_k = \mathbf{r}_k + \delta_k \mathbf{p}_{k-1}$. Skalár δ_k zvolíme potom tak, aby byla dodržena \mathbf{A} -ortogonalita vektorů $\mathbf{p}_k, \mathbf{p}_{k-1}$.

1.3.1 Předpodmínění

Pro efektivitu algoritmu CG je zásadní použití vhodného *předpodmínění*. Myšlenka předpodmínění spočívá v použití metody CG na ekvivalentní soustavu

$$(\mathbf{P}^{-1} \mathbf{A} \mathbf{P}^{-T})(\mathbf{P}^T \mathbf{x}) = \mathbf{P}^{-1} \mathbf{b}.$$

Úkolem je najít vhodnou matici \mathbf{P} tak, aby aproximace rychle konvergovali a abychom mohli snadno získat řešení původní soustavy $\mathbf{A} \mathbf{x} = \mathbf{b}$. Jednou z možností je například Jacobiho předpodmínění, kde volíme $\mathbf{P} = \text{diag}(\mathbf{A})$.

Jinou, často používanou formou předpodmínění je neúplný Choleského rozklad. Pokud je matice \mathbf{A} symetrická a pozitivně definitní, existuje jednoznačně určený Choleského rozklad

$$\mathbf{A} = \mathbf{L} \mathbf{L}^T,$$

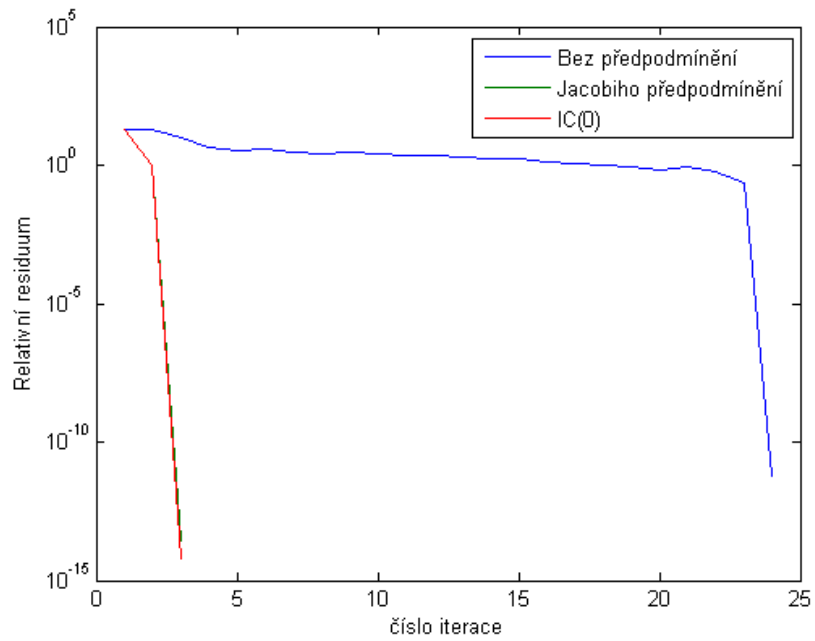
kde \mathbf{L} je dolní trojúhelníková matice. Jednou z vlastností matice tuhosti je to, že je řídká. Tuto vlastnost chceme zachovat a tak požadujeme totéž i po matici \mathbf{L} . Princip varianty neúplného Choleského rozkladu IC(0) spočívá v tom, že nenulové jsou pouze ty prvky matice l_{ij} , pro které je nenulový prvek matice \mathbf{A} a_{ij} . Neúplný Choleského rozklad tedy hledá matici \mathbf{L} tak, aby platilo

$$\mathbf{A} \approx \mathbf{L} \mathbf{L}^T$$

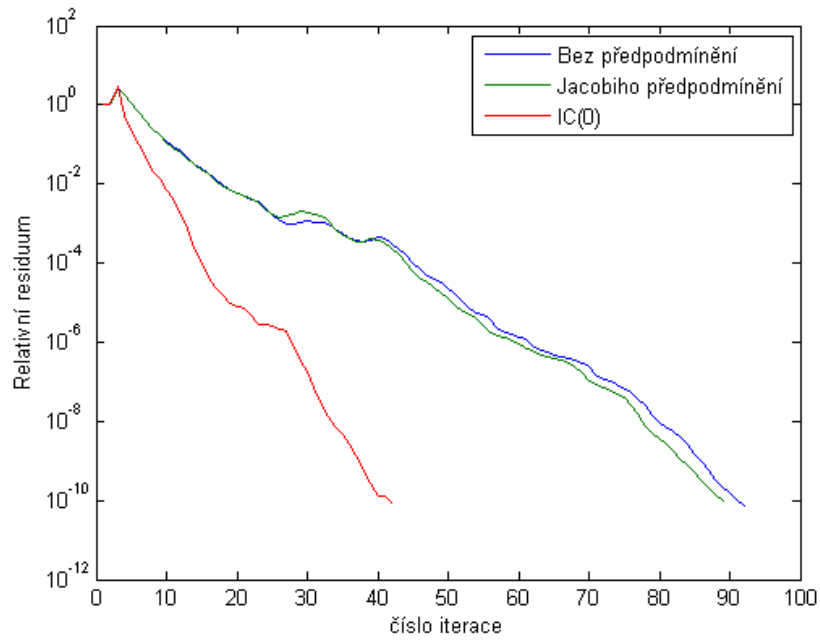
Abychom ukázali jak je předpodmínění důležité, budeme měřit rychlost konvergence CG. Tu můžeme měřit například pomocí *residua* $\mathbf{r}_k = \|\mathbf{A} \mathbf{x}_k - \mathbf{b}\|$. Častěji se však používá takzvané *relativní residuum* definované jako

$$\frac{\|\mathbf{A} \mathbf{x}_k - \mathbf{b}\|}{\|\mathbf{b}\|}.$$

Na grafech (1.6) a (1.7) je vynesené relativní residuum v jednotlivých iteracích. Ty ilustrují jak důležitou úlohu předpodmínění hraje.



Obrázek 1.6: Rychlost konvergence metody sdružených gradientů pro soustavu získanou diskretizací difuzní rovnice v 1D



Obrázek 1.7: Rychlost konvergence metody sdružených gradientů pro soustavu získanou diskretizací difuzní rovnice ve 2D

Kapitola 2

Advekčně difuzní rovnice

Mnoho modelů proudění, ale i dalších fyzikálních problémů, vede na rovnici advekčně-difuzního typu. Ta v sobě kombinuje jak difuzní člen stejný jako v již popsané difuzní rovnici, tak ještě advekční člen. V této kapitole nejprve použijeme metodu konečných diferencí na advekčně-difuzní rovnici v jedné dimenzi a metodu konečných prvků na problém ve dvou dimenzích. Ve většině praktických problémů je difuze méně podstatným jevem. Zaměříme se tedy na úskalí hledání přibližného řešení s dominantní advekcí. Důležité je zmínit, že pokud je dominantní advekce, má rovnice spíše hyperbolický charakter.

Tomuto tématu se věnují především kapitoly 3 a 4 v publikaci [1]

2.1 Advekčně difuzní rovnice v jedné dimenzi

V jedné dimenzi má tato rovnice tvar

$$u_t + \sigma u_x - \varepsilon u_{xx} = f(x),$$

kde $u(x, t)$ je hledaná funkce, $\sigma \in \mathbb{R}$ advekční parametr, $\varepsilon \in \mathbb{R}$ difuzní parametr a $f(x)$ opět zdrojový člen. Zaměříme se však na stacionární rovnici

$$\sigma u_x - \varepsilon u_{xx} = f(x), \tag{2.1}$$

Doplňme dále homogenní Dirichletovy okrajové podmínky na intervalu $(0, 1)$

$$u(0) = 0, u(1) = 0. \tag{2.2}$$

Tuto okrajovou úlohu budeme řešit metodou konečných diferencí.

2.1.1 Diskretizace metodou konečných diferencí

Přibližné řešení advekčně difuzní rovnice (2.1) s okrajovými podmínkami (2.2) můžeme v jedné dimenzi opět nalézt například pomocí metody konečných diferencí. Budeme postupovat obdobně jako v případě difuzní rovnice. Uvažujme opět ekvidistantní síť, po nahrazení derivací diferencemi zjistíme, že hledané přibližné řešení musí splňovat

$$\sigma \frac{U_{k+1} - U_{k-1}}{2h} - \varepsilon \frac{U_{k+1} - 2U_k + U_{k-1}}{h^2} = f(x_k).$$

Přibližné řešení okrajové úlohy (2.1), (2.2) v uzlových bodech tedy získáme vyřešením soustavy rovnic

$$\begin{aligned} 2\varepsilon U_1 - \left(\varepsilon + \frac{\sigma h}{2}\right)U_2 &= h^2 f(x_1) \\ -\left(\varepsilon - \frac{\sigma h}{2}\right)U_1 + 2\varepsilon U_2 - \left(\varepsilon + \frac{\sigma h}{2}\right)U_3 &= h^2 f(x_2) \\ &\vdots \\ -\left(\varepsilon - \frac{\sigma h}{2}\right)U_{N-4} + 2\varepsilon U_{N-3} - \left(\varepsilon + \frac{\sigma h}{2}\right)U_{N-2} &= h^2 f(x_{N-2}) \\ 2\varepsilon U_{N-2} - \left(\varepsilon + \frac{\sigma h}{2}\right)U_{N-1} &= h^2 f(x_{N-1}) \end{aligned}$$

Také tentokrát využijeme maticového tvaru soustavy

$$\mathbf{A}\mathbf{U} = \mathbf{F},$$

kde

$$\mathbf{F} = \begin{bmatrix} h^2 f(x_1) \\ h^2 f(x_2) \\ \vdots \\ h^2 f(x_{N-1}) \end{bmatrix},$$

je vektor pravé strany,

$$\mathbf{U} = \begin{bmatrix} U_1 \\ U_2 \\ \vdots \\ U_{N-1} \end{bmatrix},$$

značí vektor hledaných hodnot přibližného řešení v uzlových bodech a

$$\mathbf{A} = \begin{bmatrix} a & b & 0 & 0 & 0 & 0 \\ c & a & b & 0 & 0 & 0 \\ 0 & c & a & \ddots & 0 & 0 \\ 0 & 0 & \ddots & \ddots & b & 0 \\ 0 & 0 & 0 & c & a & b \\ 0 & 0 & 0 & 0 & c & a \end{bmatrix}$$

je čtvercová matice velikosti $N-1$, kde $a = 2\varepsilon$, $b = -\varepsilon + \frac{\sigma h}{2}$, $c = -\varepsilon - \frac{\sigma h}{2}$. Tato matice však není symetrická ani pozitivně definitní a tak nelze použít metodu sdružených gradientů.

2.2 Advektivně difuzní rovnice ve dvou dimenzích

Ve dvou dimenzích má advektivně-difuzní rovnice tvar

$$\boldsymbol{\sigma} \nabla u - \varepsilon \Delta u = f(x, y), \quad (2.3)$$

kde $\boldsymbol{\sigma} = [\sigma_1, \sigma_2] \in \mathbb{R}^2$ a $\nabla u(x, y) = \text{grad}(u(x, y))$

Její splnění budeme požadovat uvnitř omezené oblasti Ω . Úlohu doplníme o homogenní Dirichletovy okrajové podmínky

$$u(x, y) = 0 \text{ na } \partial\Omega. \quad (2.4)$$

Okrajovou úlohu budeme řešit opět metodou konečných prvků.

2.2.1 Slabé řešení

Při odvození slabého řešení advektivně difuzní rovnice využijeme částečně poznatků z první kapitoly. Požadujeme tentokrát splnění

$$\iint_{\Omega} \varepsilon \nabla u \nabla v d\Omega + \iint_{\Omega} \boldsymbol{\sigma} \nabla u v d\Omega = \iint_{\Omega} f v d\Omega$$

pro všechna $v \in V$. Označíme konvekční člen $c(u, v)$ a aplikujeme na něj Greenovu formuli

$$c(u, v) = \iint_{\Omega} \boldsymbol{\sigma} \nabla u v d\Omega = - \iint_{\Omega} \text{div}(v \boldsymbol{\sigma}) u d\Omega + \iint_{\partial\Omega} v \boldsymbol{\sigma} \mathbf{n} u d\Omega = - \iint_{\Omega} (\boldsymbol{\sigma} \nabla v) u d\Omega$$

Slabým řešením okrajové úlohy (2.3, 2.4) tedy nazveme takovou funkci $u(x, y) \in V_g$, která splňuje rovnost

$$\iint_{\Omega} \varepsilon \nabla u \nabla v d\Omega - \iint_{\Omega} (\boldsymbol{\sigma} \nabla v) u d\Omega = \iint_{\Omega} f v d\Omega \quad (2.5)$$

Definujeme bilineární formu

$$a(u, v) = \iint_{\Omega} \varepsilon \nabla u \nabla v d\Omega - \iint_{\Omega} (\boldsymbol{\sigma} \nabla v) u d\Omega, \quad (2.6)$$

funkcionál

$$F(v) = \iint_{\Omega} f v d\Omega. \quad (2.7)$$

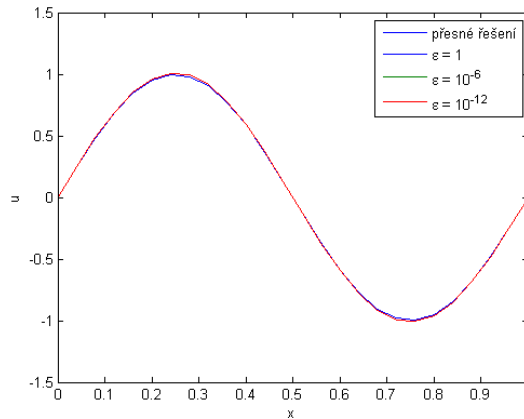
Hledáme $u \in V_g$ takové, aby platilo

$$a(u, v) = F(v) \quad \forall v \in V.$$

Poznamenejme že bilineární forma $a(u, v)$ v tomto případě již není symetrická. To způsobuje konvekční člen, který je naopak antisymetrický, neboť zřejmě platí $c(u, v) = -c(v, u)$. Aproximaci metodou konečných prvků potom provedeme stejně jako v případě difuzní rovnice.

2.3 Výpočet s dominantní advekcí

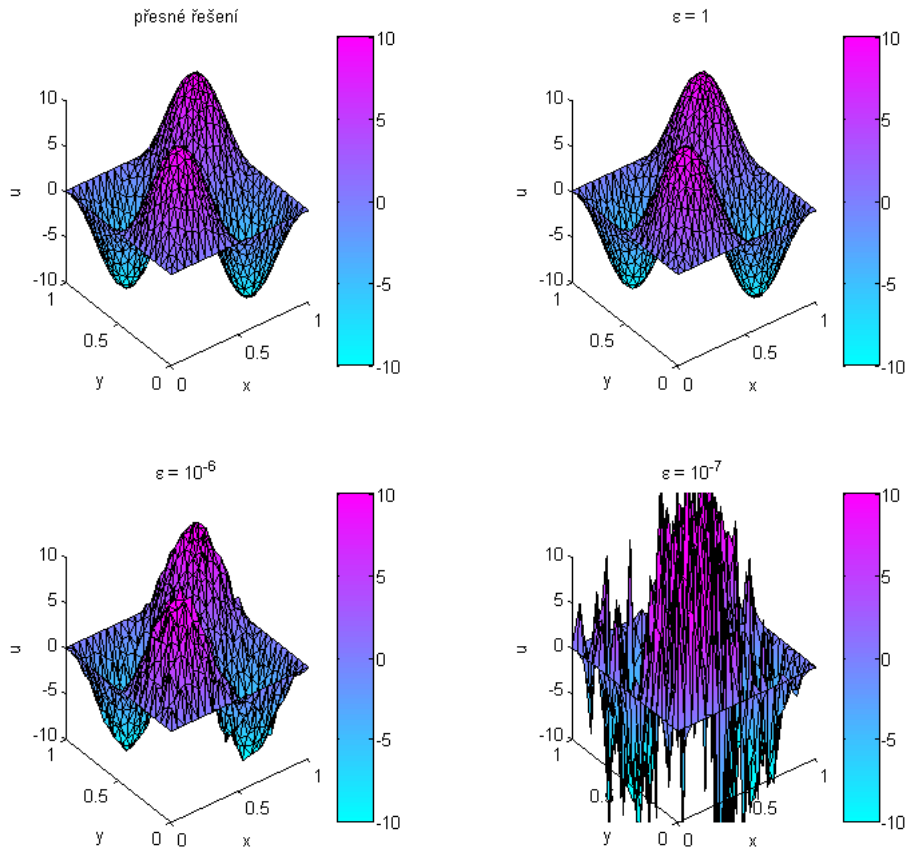
V této části se pokusíme ukázat, že výpočet přibližného řešení advektivně difuzní rovnice s dominantní advekcí může být problematický. Nejprve se zaměříme na jednodimenzionální případ. Do jednoho obrázku jsme vykreslili řešení rovnice s různými parametry ε . Druhý parametr jsme zvolili pevně $\sigma = 1$ a zdrojový člen má takový tvar, aby řešením byla funkce $u(x) = \sin(x)$.



Obrázek 2.1: Přesné a přibližné řešení advektivně-difuzní rovnice pro několik parametrů ε v 1D

V jednodimenzionální případě k žádným problémům nedochází. Stejný numerický experiment tedy provedeme i ve dvou dimenzích. Zvolíme pevně $\boldsymbol{\sigma} = [1, 1]$ a zdrojový člen tak, by řešení byla funkce $u(x, y) = 10 \sin(2\pi x) \sin(2\pi y)$. Obrázek (2.2) potom ukazuje přesné a přibližné řešení pro několik hodnot parametru ε . Pokud je difuze příliš malá v

řešení se objevují oscilace, které ho mohou zcela znehodnotit. Jak ukážeme později, tento problém lze zmírnit použitím vhodných metod pro řešení soustavy lineárně algebraických rovnic. Nelze ho však zcela eliminovat.



Obrázek 2.2: Přesné a přibližné řešení advekčně-difuzní rovnice pro několik parametrů ε v 2D

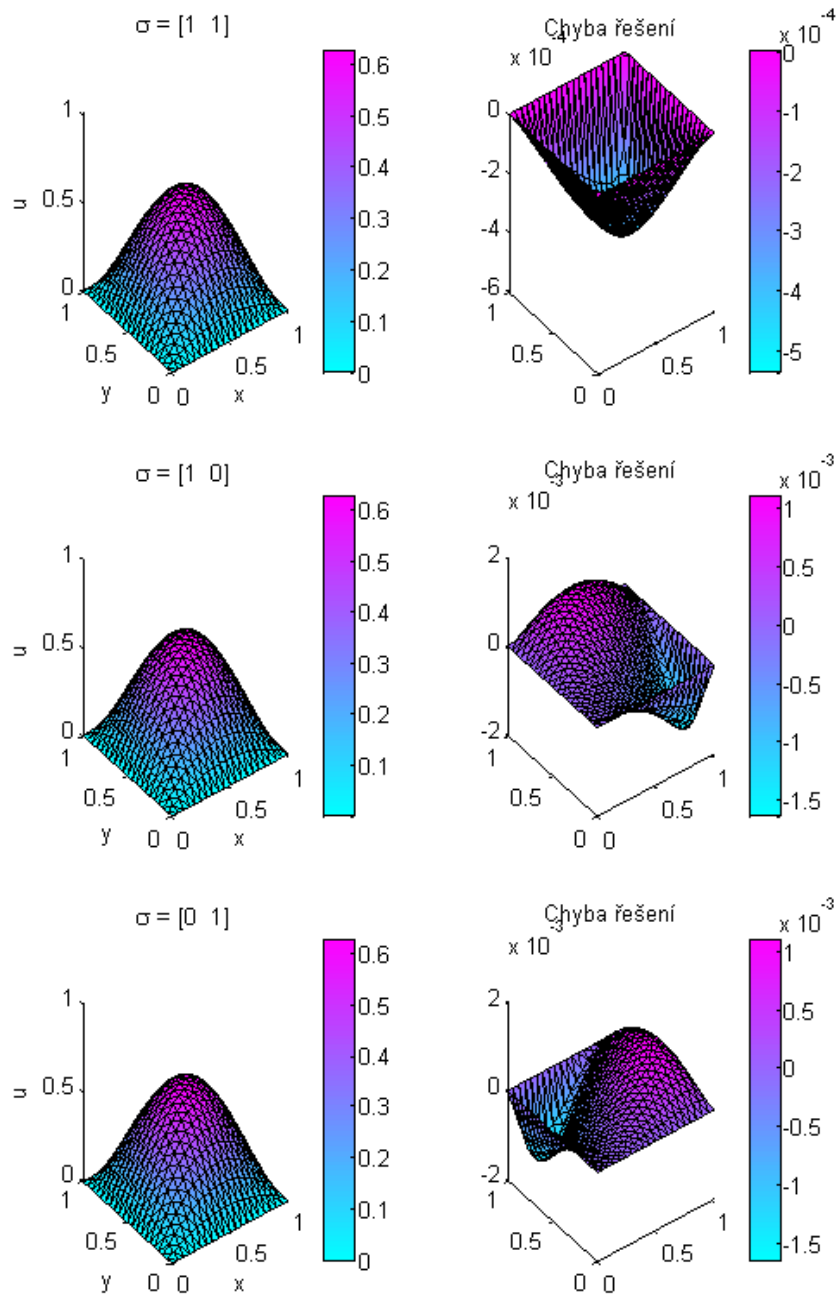
Čím je difuze slabší, tím se řešení advekčně-difuzní rovnice $u(x, y)$ blíží řešení hyperbolické rovnice

$$\sigma \nabla \hat{u} = f(x, y).$$

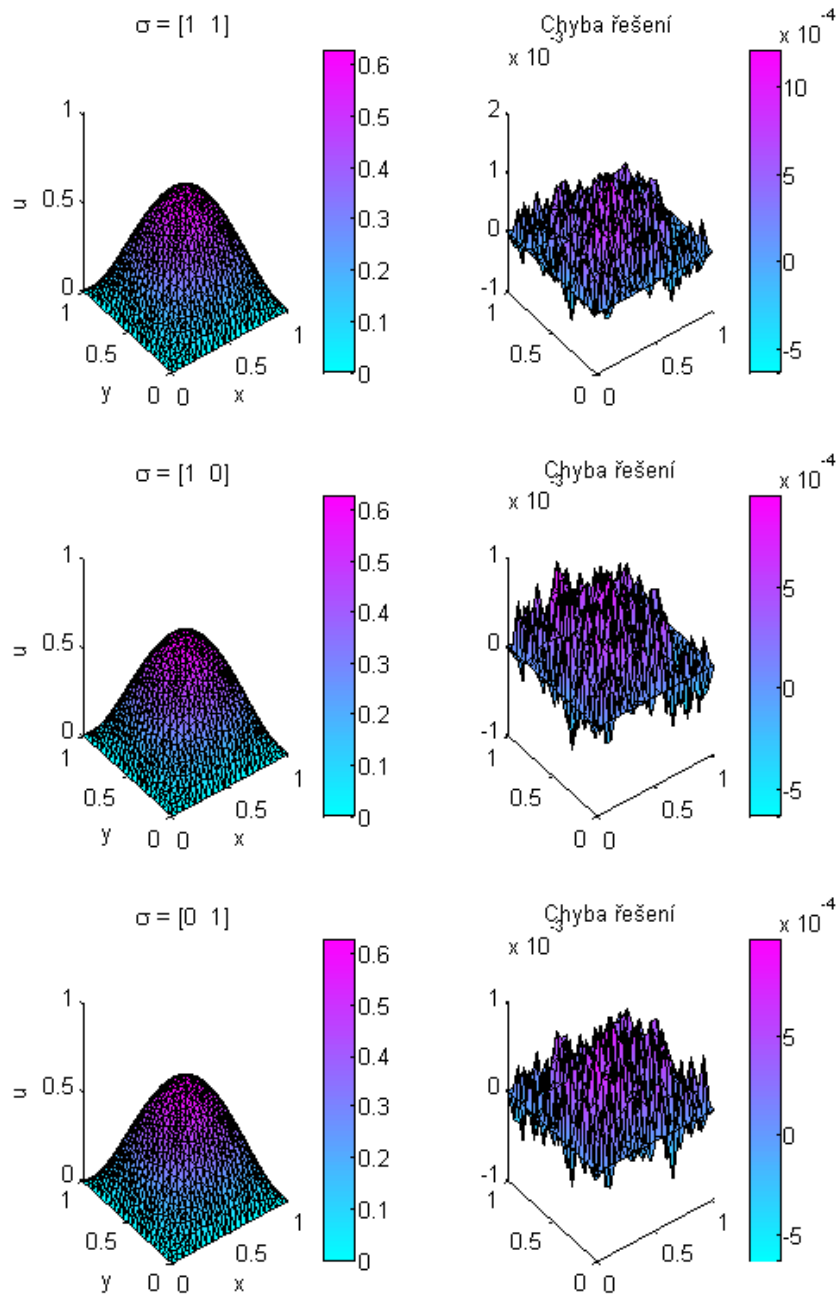
Platí tedy, že čím více je advekce dominantní, tím se advekčně-difuzní rovnice chová více jako hyperbolická.

2.4 Závislost chyby řešení na tvaru sítě

Jedním z podstatných aspektů metody konečných prvků je nejen rozlišení sítě, na které výpočet provádíme. Pokud použijeme pravidelnou síť zjistíme, že chyba přibližného řešení závisí také na směru vektoru σ . To ukazují obrázky (2.3). Tomuto efektu lze však snadno zabránit použitím nepravidelné sítě, jak je zřejmé na obrázku (2.4). Je nutné zde poznamenat, že při použití nepravidelná síť obsahuje trojúhelníky různých velikostí. Proto tato síť obsahuje více uzlů, i když je její rozlišení stejné jako rozlišení pravidelné sítě.



Obrázek 2.3:



Obrázek 2.4:

Kapitola 3

Metody řešení soustav lineárně algebraických rovnic

V předchozím textu jsme představili dvě metody řešení okrajových úloh. Obě tyto metody potřebovaly vyřešit soustavu lineárních algebraických rovnic. To bychom mohli samozřejmě udělat přímou metodou. Pokud se však podíváme například na výpočetní složitost Gaussovy eliminační metody, tak ta je například dle [6] řádově n^3 . Pokud však budeme chtít řešit praktické úlohy, potřebujeme vyřešit i soustavy s maticemi o velikosti v rádech milionů a větších. Proto se používají metody iterační. Také víme, že matice se kterými pracujeme jsou řídké. Proto budeme popisovat metody, které se často používají pro řešení takových problémů. Již popsaly metodu sdružených gradientů. Tu však lze použít pouze za předpokladu, že matice je symetrická a pozitivně definitní. Proto musíme hledat jiné metody.

Cílem této části textu je přiblížit principy některých používaných metod řešení soustav lineárních algebraických rovnic. Nebudeme však většinou popisovat přesnou algoritmizaci, ale zaměříme se na základní popis metod a některých jejich vlastností tak, abychom se později mohli pokusit vysvětlit jevy ke kterým dochází při použití těchto metod k řešení soustav získaných metodou konečných prvků, nebo metodou konečných diferencí. Proto také budeme předpokládat pouze reálné matice a vektory.

Tento přehled iteračních metod se opírá zejména o literaturu [6], [7], [3], [4].

3.1 Metody Krylovova podprostoru

Pro řešení problémů s řídkou maticí se často používají takzvané *projekční metody*. Ty spočívají v myšlence nehledat řešení na celém prostoru \mathbb{R}^n , ale pouze na nějakém vhodném podprostoru. Obecná projekční metoda hledá n -tou iteraci \mathbf{x}_n v prostoru aproximace \mathcal{S}_n , resp. na varietě

$$\mathbf{x}_n \in \mathbf{x}_0 + \mathcal{S}_n$$

Přesně je \mathbf{x}_n určena podmínkou, kterou klademe na residuový vektor $\mathbf{r}_n = \mathbf{b} - \mathbf{A}\mathbf{x}$

$$\mathbf{r}_n \perp \mathcal{L}_n.$$

Prostor \mathcal{L}_n se nazývá *levý prostor*.

Krylovovské metody volí prostor aproximace \mathcal{S}_n jako *Krylovův prostor*

$$\mathcal{K}_k(\mathbf{A}, \mathbf{r}_0) = \text{span}\{\mathbf{r}_0, \mathbf{A}\mathbf{r}_0, \dots, \mathbf{A}^{k-1}\mathbf{r}_0\}$$

Další důležitou volbou je volba levého prostoru. Tou se definují jednotlivé metody.

3.1.1 Báze Krylovova prostoru

Pro výpočet báze Krylovova prostoru můžeme využít *Arnoldiho algoritmu*. Ten počítá bázi $\mathbf{V}_k = \mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_k$ Krylovova prostoru $\mathcal{K}_k(\mathbf{A}, \mathbf{q})$ pomocí Gram-Schmidtova ortogonalizačního procesu. Volí $\mathbf{v}_1 = \frac{\mathbf{q}}{\|\mathbf{q}\|}$ a další vektory počítá ortogonalizací vektoru $\mathbf{A}\mathbf{v}_{n-1}$ vůči předchozím $k-1$ vektorům. Maticově lze potom Arnoldiho algoritmus vyjádřit vztahem

$$\mathbf{A}\mathbf{V}_k = \mathbf{V}_k\mathbf{H}_k + h_{k+1}\mathbf{v}_{k+1},$$

kde \mathbf{H}_k je horní Hessenbergova matice

$$\mathbf{H} = \begin{pmatrix} h_{11} & h_{12} & h_{13} & \cdots & h_{1k} \\ h_{21} & h_{22} & h_{23} & \cdots & h_{2k} \\ 0 & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \ddots & \ddots & h_{k-1,k} \\ 0 & 0 & 0 & h_{k,k-1} & h_{kk} \end{pmatrix}$$

Pokud je matice \mathbf{A} symetrická, musí být Hessenbergova matice také symetrická, neboť platí

$$\mathbf{H}_k = \mathbf{V}_k^T \mathbf{A} \mathbf{V}_k = \mathbf{V}_k^T \mathbf{A}^T \mathbf{V}_k = (\mathbf{V}_k^T \mathbf{A} \mathbf{V}_k)^T = \mathbf{H}_k^T$$

V tom případě budeme tuto matici značit \mathbf{T}_k

$$\mathbf{T}_k = \begin{pmatrix} \alpha_1 & \beta_1 & 0 & 0 & 0 \\ \beta_1 & \alpha_2 & \beta_2 & 0 & 0 \\ 0 & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & \beta_{n-2} & \alpha_{n-1} & \beta_{n-1} \\ 0 & 0 & 0 & \beta_{n-1} & \alpha_n \end{pmatrix}$$

Arnoldiho algoritmus potom lze snadno zjednodušit. Tento zjednodušený algoritmus se nazývá *Lanczosův algoritmus*. Pokud maticové vyjádření tohoto algoritmu

$$\mathbf{A}\mathbf{V}_k = \mathbf{V}_k\mathbf{T}_k + \beta_{k+1}\mathbf{v}_{k+1}$$

přepíšeme po řádcích, získáme tříčlenou rekurenci

$$\beta_{k+1}\mathbf{v}_{k+1} = \mathbf{A}\mathbf{v}_k - \beta_k\mathbf{v}_{k-1} - \alpha_k\mathbf{v}_k.$$

Lanczosův algoritmus tak má oproti Arnoldiho velkou výhodu. Zatímco v Arnoldiho algoritmu je zapotřebí k výpočtu k -tého prvku všech $k - 1$ předchozích, Lanczosův algoritmus potřebuje pouze dva předchozí, je však zapotřebí, aby matice \mathbf{A} byla symetrická. Pro výpočet lze však využít také *Lanczosův algoritmus pro nesymetrické matice*. Ten počítá hned báze dvou Krylovových prostorů. Označíme $\mathbf{v}_1, \dots, \mathbf{v}_k$ bázi prostoru $\mathcal{K}(\mathbf{A}, \mathbf{v})$ a $\mathbf{w}_1, \dots, \mathbf{w}_k$ bázi prostoru $\mathcal{K}(\mathbf{A}^T, \mathbf{w})$. Vektory \mathbf{v}_k a \mathbf{w}_k musí splňovat podmínky

$$\mathbf{v}_i^T \mathbf{w}_j = \begin{cases} 0 & i \neq j \\ 1 & i = j \end{cases}.$$

Takové vektory které tuto podmínku splňují nazveme *biortogonální*. Maticově lze tento algoritmus vyjádřit následovně

$$\begin{aligned} \mathbf{A}\mathbf{V}_k &= \mathbf{V}_k\mathbf{T}_k + \omega_{k+1}\mathbf{v}_{k+1}\mathbf{e}_k \\ \mathbf{A}^T\mathbf{W}_k &= \mathbf{W}_k\mathbf{T}_k^T + \omega_{k+1}\mathbf{w}_{k+1}\mathbf{e}_k \\ \mathbf{W}_k^T\mathbf{V}_k &= \mathbf{I}, \end{aligned}$$

kde $\mathbf{V}_k = [\mathbf{v}_1, \dots, \mathbf{v}_k]$, $\mathbf{W}_k = [\mathbf{w}_1, \dots, \mathbf{w}_k]$ a \mathbf{T}_k je opět třídiagonální matice (která však tentokrát není symetrická). Tento algoritmus tak konstruuje báze Krylova pro obecnou nesymetrickou matici \mathbf{A} . Nevýhodou však je, že může dojít k situaci, kdy nově vznikající vektor \mathbf{w} je kolmý na vektor \mathbf{v}_{k+1} a algoritmus se zastaví. V tom případě říkáme, že došlo k *Lanczosovu ukončení*. Problémy nastanou také, pokud je $\mathbf{w}^T\mathbf{v}_{k+1}$ blízké nule. Abychom nedošlo k Lanczosovu ukončení používají se tzv. *look-ahead* strategie. Jejich základní myšlenkou je, že se konstruují pouze ty vektory, jejichž výpočet je stabilní. Podrobněji se tomuto tématu věnuje například práce [7].

3.1.2 CG jako projekční metoda

Metodu sdružených gradientů lze pochopit také jako metodu projektivní. Hledá totiž řešení na prostoru $\mathcal{K}_k(\mathbf{A}, \mathbf{r}_0)$ a lze ho vyjádřit ve tvaru

$$\mathbf{x}_k = \mathbf{x}_0 + \mathbf{V}_k\mathbf{y}_k,$$

k -té residuum potom má tvar

$$\mathbf{r}_k = \mathbf{b} - \mathbf{A}\mathbf{x}_k = \mathbf{b} - \mathbf{A}(\mathbf{x}_0 + \mathbf{V}_k\mathbf{y}_k) = \mathbf{b} - \mathbf{A}\mathbf{x}_0 - \mathbf{A}\mathbf{V}_k\mathbf{y}_k = \mathbf{r}_0 - \mathbf{A}\mathbf{V}_k\mathbf{y}_k,$$

protože residuum musí být ortogonální ke všem bázovým vektorům platí

$$0 = \mathbf{V}_k^T \mathbf{r}_k = \mathbf{V}_k^T \mathbf{r}_0 + \mathbf{V}_k^T \mathbf{A} \mathbf{V}_k \mathbf{y}_k = \|\mathbf{r}_0\| \mathbf{e}_1 + \mathbf{T}_k \mathbf{y}_k$$

kde $\mathbf{V}_k = \mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_k$ a \mathbf{y}_k je vektor souřadnic.

3.1.3 Zobecněná metoda minima residuí (GMRES)

Tato metoda volí levý prostor $\mathcal{L}_k = \mathbf{A}\mathcal{K}_k(\mathbf{A}, \mathbf{r}_0)$. Vektor \mathbf{x}_k se potom zachází na varietě

$$\mathbf{x}_k \in \mathbf{x}_0 + \mathcal{K}_k(\mathbf{A}, \mathbf{r}_0)$$

a platí

$$\mathbf{r}_k \perp \mathbf{A}\mathcal{K}_k(\mathbf{A}, \mathbf{r}_0)$$

Bázi Krylovova prostoru $\mathbf{A}\mathcal{K}_k(\mathbf{A}, \mathbf{r}_0)$ hledáme pomocí Arnoldiho algoritmu. Vektor \mathbf{x}_k chceme vyjádřit jako lineární kombinaci

$$\mathbf{x}_k = \mathbf{x}_0 + \mathbf{V}_k\mathbf{y}_k.$$

Residuum v k -tém kroku potom můžeme vyjádřit ve tvaru

$$\begin{aligned} \mathbf{r}_k &= \mathbf{b} - \mathbf{A}\mathbf{x}_k = \mathbf{b} - \mathbf{A}(\mathbf{x}_0 + \mathbf{V}_k\mathbf{y}_k) = \mathbf{r}_0 - \mathbf{A}\mathbf{V}_k\mathbf{y}_k = \\ &= \|\mathbf{r}_0\| \mathbf{v}_1 - \mathbf{V}_{k+1} \mathbf{H}_{k+1,k} \mathbf{y}_k = \mathbf{V}_{k+1} (\|\mathbf{r}_0\| \mathbf{e}_1 - \mathbf{H}_{k+1,k} \mathbf{y}_k) \end{aligned}$$

Protože residuum má být minimální, hledané souřadnice \mathbf{y}_k nalezneme jako

$$\mathbf{y}_k = \arg \min_{\mathbf{y} \in \mathbb{R}^k} (\|\mathbf{r}_0\| \mathbf{e}_1 - \mathbf{H}_{k+1,k} \mathbf{y}) \quad (3.1)$$

Pro symetrické matice lze metodu zjednodušit. Pro určení báze Krylovova prostoru potom lze totiž využít Lanczosova algoritmu, taková metoda se nazývá MINRES. Oproti tomu metoda QMR využívá nesymetrického lanczosova algoritmu.

3.1.4 Metoda bikonjugovaných gradientů

Metoda bikonjugovaných gradientů, podobně jako nesymetrický Lanczosův algoritmus, konstruuje biortogonální vektory. Aproximace řešení má následující vlastnosti

$$\mathbf{x}_k \in \mathbf{x}_0 + \mathcal{K}_k(\mathbf{A}, \mathbf{r}_0) \quad \mathbf{r}_k \perp \mathcal{K}_k(\mathbf{A}^T, \mathbf{s}_0),$$

kde \mathbf{s}_0 je pomocný počáteční vektor, tzv. *stínový vektor*. Ten musí splňovat podmínku $\mathbf{s}_0^T \mathbf{r}_0 \neq 0$. Bázi prostoru $\mathcal{K}_k(\mathbf{A}^T, \mathbf{s}_0)$ můžeme využít k řešení soustavy $\mathbf{A}^T \mathbf{y} = \mathbf{c}$. Aproximace \mathbf{y}_k potom splňují

$$\mathbf{y}_k \in \mathbf{y}_0 + \mathcal{K}_k(\mathbf{A}, \mathbf{s}_0) \quad \mathbf{s}_k \perp \mathcal{K}_k(\mathbf{A}, \mathbf{r}_0).$$

Vektor $\mathbf{s}_k = \mathbf{c} - \mathbf{A}^T \mathbf{y}_k$ je residuum aproximace \mathbf{y}_k . Metoda bikonjugovaných gradientů tak řeší současně systémy $\mathbf{A}\mathbf{x} = \mathbf{b}$ a $\mathbf{A}^T \mathbf{y} = \mathbf{c}$, k čemuž využívá biortogonální báze. Tuto metodu lze chápat také jako další zobecnění metody CG, neboť pokud je \mathbf{A} symetrická poz. definitní lze algoritmus BiCG redukovat na CG. Z této metody se dále vyvinuly metody BiCGstab, BiCGstab(l) a CGS. Metody BiCGstab a BiCGstab(l) by měli být stabilnější. Oproti tomu pro metodu CGS potom platí, že pokud konverguje metoda BiCG, CGS konverguje rychleji. Může u ní však častěji dojít k zastavení.

Metoda LSQR

Metoda LSQR využívá pro generování Krylovových podprostorů takzvanou Golub-Kahanovu bidiagonalizaci. Ta vytváří posloupnost Krylovových podprostorů $\mathcal{K}_k(\mathbf{A}^T \mathbf{A}, \mathbf{A}^T \mathbf{b})$ pro obecnou obdélníkovou matici. Toto metodu je tedy možné využít také pro řešení problému

$$\mathbf{A}\mathbf{x} \approx \mathbf{b}$$

ve smyslu nejmenších čtverců. Tato metoda je velmi numericky stabilní a je vhodná pro výpočet špatně podmíněných úloh.

3.1.5 Metoda pro antisymetrickou matici

Podívejme se nyní na případ, kdy bude matice \mathbf{A} antisymetrická. Potom Hessenbergova matice musí být také antisymetrická, neboť platí

$$\mathbf{H}_k = \mathbf{V}_k^T \mathbf{A} \mathbf{V}_k = -\mathbf{V}_k^T \mathbf{A}^T \mathbf{V}_k = -(\mathbf{V}_k^T \mathbf{A} \mathbf{V}_k)^T = -\mathbf{H}_k^T.$$

Struktura této matice tedy musí být následující

$$\mathbf{H} = \begin{pmatrix} 0 & \beta_1 & 0 & 0 & 0 \\ -\beta_1 & 0 & \beta_2 & 0 & 0 \\ 0 & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & -\beta_{k-1} & 0 & \beta_k \\ 0 & 0 & 0 & -\beta_k & 0 \end{pmatrix} \quad (3.2)$$

Nyní můžeme postupovat podobně jako je popsána metoda sdružených gradientů z pohledu projekčních metod (3.1.2).

Je však potřeba si uvědomit, že pro k liché je matice (3.2) singulární. Musíme tedy spočítat vždy sudý počet prvků báze Krylovova prostoru. Získali jsme tak metodu pro řešení soustavy lineárních algebraických rovnic a antisymetrickou maticí. Tuto metodu budeme dále značit ACG.

3.1.6 Konvergence projekčních metod

Mějme obecnou projekční metodu, prostor aproximace \mathcal{S}^k , levý prostor \mathcal{L}^k . Aproximace řešení tedy splňují

$$\mathbf{x}_k \in \mathbf{x}_0 + \mathcal{S}^k \quad \mathbf{r}_k \perp \mathcal{L}^k.$$

S rostoucím počtem iterací k roste také dimenze prostorů \mathcal{S}^k a \mathcal{L}^k . Pro $k = n$ tedy platí $\mathcal{L}^k = \mathbb{R}^n$ a z podmínky $\mathbf{r}_k \perp \mathbb{R}^n$ plyne $\mathbf{r} = 0$. Přesné řešení tedy získáme maximálně po n krocích.

V případě metody sdružených gradientů potom rychlost konvergence záleží na poměru největšího a nejmenšího vlastního čísla matice $\mathbf{A} \frac{\lambda_{max}}{\lambda_{min}}$. Obecně je však velmi těžké předpovědět rychlost konvergence Krylovovských metod. Ta totiž v případě obecné nesymetrické matice již na vlastních číslech nezáleží.

3.1.7 Předpodmínění

Důležitou součástí výše popisovaných metod je použití vhodného *předpodmínění*. Myšlenka předpodmínění spočívá v použití iterační metody na ekvivalentní soustavu

$$(\mathbf{P}_L \mathbf{A} \mathbf{P}_R)(\mathbf{P}_R^{-1} x) = \mathbf{P}_L \mathbf{b}.$$

Úkolem je najít vhodné matice \mathbf{P}_L a \mathbf{P}_R tak, aby aproximace rychle konvergovali a abychom mohli snadno získat řešení původní soustavy $\mathbf{A} \mathbf{x} = \mathbf{b}$. Opět lze využít například Jacobiho předpodmínění. Účinnější metodou předpodmínění je však neúplný LU rozklad. My budeme využívat variantu ILU(0). Ta hledá horní trojúhelníkovou matici \mathbf{U} a dolní trojúhelníkovou matici \mathbf{L} tak, aby platilo

$$\mathbf{A} \approx \mathbf{L} \mathbf{U}$$

Zaplnění matic \mathbf{L} a \mathbf{U} potom odpovídá zaplnění matice \mathbf{A} .

3.1.8 Metody v prostředí MATLAB

Tuto práci jsme implementovali v prostředí Matlab. V tomto prostředí je několik dostupných metod pro řešení soustav lineárně algebraických rovnic implementováno. V následující tabulce jsou uvedeny metody, které v naší práci použijeme. Mimo nich použijeme také námi navrženou metodu ACG, kterou jsme také implementovali v prostředí Matlab.

Název metody	Příkaz pro spuštění metody	Požadavky na matici \mathbf{A}
Metoda bi-sdružených gradientů	bicg	čtvercová
Metoda BiCGstab	bicgstab	čtvercová
Metoda BiCGstab(1)	bicgstabl	čtvercová
Metoda CGS	cgls	čtvercová
Zobecněná metoda minimálních residuí	gmres	čtvercová
Metoda LSQR	lsqr	—
Metoda minimálních residuí	minres	čtvercová, symetrická
Metoda sdružených gradientů	pcg	čtvercová, SPD ¹
Metoda minimálních kvazi-reziduí	qmr	čtvercová
Metoda SYMMLQ	symmlq	čtvercová, symetrická
Metoda TFQMR	tfqmr	čtvercová

Obrázek 3.1: Tabulka metod implementovaných v prostředí Matlab

Kapitola 4

Numerické experimenty

Seznámili s problematikou výpočtu advekčně-difuzní rovnice s dominantní advekcí. V této kapitole provedeme několik numerických experimentů, jejichž účelem bude především ukázat, jak ovlivňuje poměr paramaterů rychlost konvergence iteračních metod při různé jemnosti sítě.

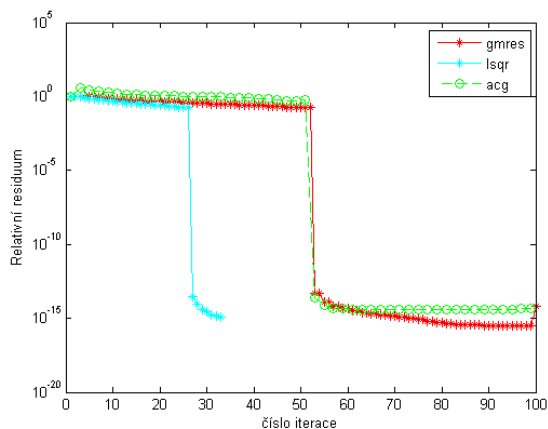
4.1 Problém dominantní advekce

Nejprve se však vrátíme k problematice výpočtu přibližného řešení advekčně-difuzní rovnice s dominantní advekcí. Již jsme se tomuto tématu věnovali v Kapitole 3, tentokrát ho však uchopíme z pohledu iteračních metod pro řešení soustav lineárně algebraických rovnic. Zaměříme se nyní na advekční rovnici

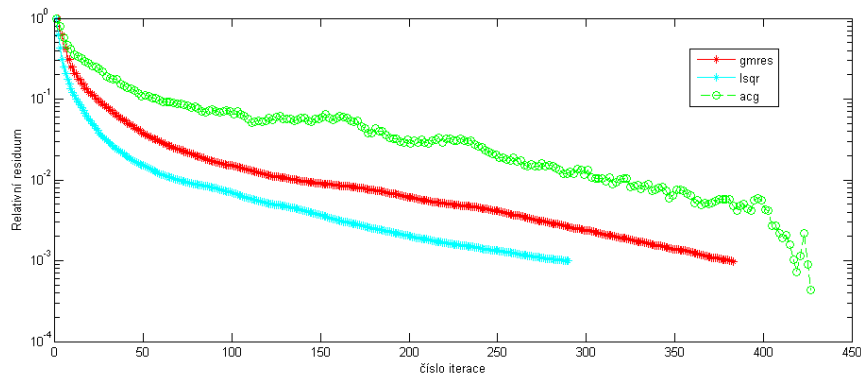
$$\sigma \nabla u = f(x, y).$$

Budeme opět uvažovat homogenní Dirichletovi okrajové podmínky. Jak matice soustavy lineárně algebraických rovnic získané metodou konečný diferencí pro rovnici v jedné dimenzi, tak i matice soustavy získané metodou konečných prvků pro rovnici ve dvou dimenzích jsou antisymetrické. Důsledkem toho mají obě matice na diagonále nuly. To má zásadní vliv na metody předpodmínění, které jsme v této práci uvedli.

Zřejmě nemá v tomto případě žádný smysl uvažovat Jacobiho předpodmínění. Ze stejného důvodu však nelze použít ani algoritmus ILU(0). Proto jsme provedli následující výpočty bez předpodmínění. Na obrázcích (4.1,4.2) lze pozorovat závislost relativního residua na počtu iterací zvolených metod.

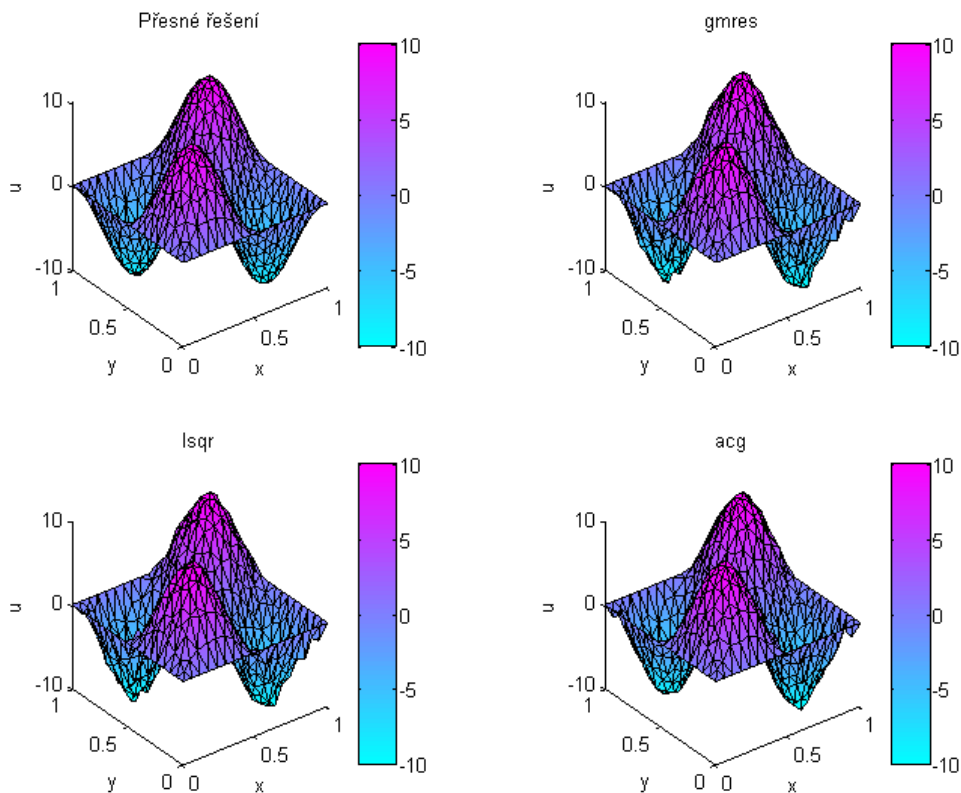


Obrázek 4.1: Rychlost konvergence metod v 1D případě. Pro výpočet okrajové úlohy jsme rozdělili interval $(0, 1)$ na 100 dílů a zvolili $\sigma = 1$.



Obrázek 4.2: Rychlost konvergence metod v 2D případě. Pro výpočet jsme použili nepravidelnou síť s rozlišením $h = 0.06$. Parametr rovnice jsme zvolili $\sigma = [1, 1]$ a zdrojový člen byl volen tak abychom znali přesné řešení.

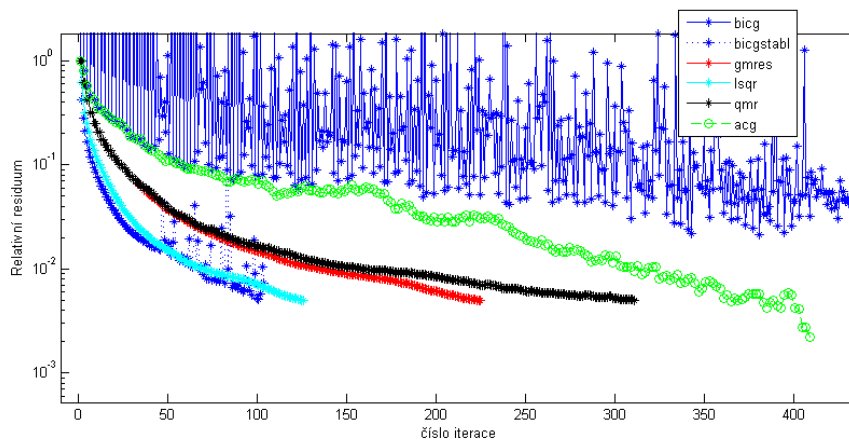
Z testovaných metod konvergují pouze metody GMRES, LSQR a ACG. Ostatní metody buď nekonvergují, nebo dojde brzy k jejich zastavení. Podíváme-li se blíže na konvergenci metody GMRES, zjistíme, že v lichých iteracích se aproximace řešení nemění. To je dáno tvarem Hessenbergovy matice. Ta je, jak jsme již uvedli v předešlé kapitole, pro k liché singulární, což se projeví při hledání k -tých souřadnic předpisem (3.1). Podobné chování ale můžeme očekávat vždy v případě $\varepsilon \gg \sigma$, neboť potom bude matice A velmi blízka antisymetrické matici. Obrázek (4.3) ilustruje, jak je v tom případě řešení dvoudimenzionálního problému zatíženo chybou.



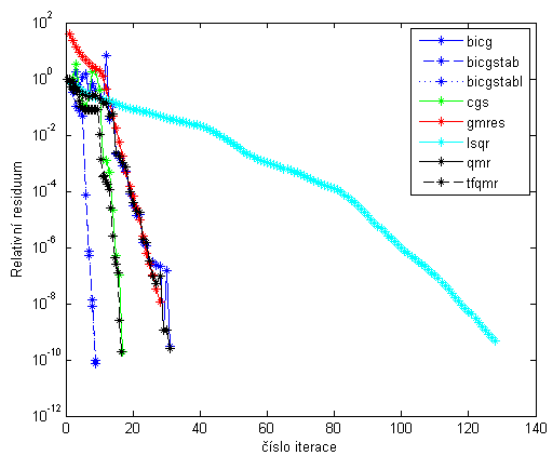
Obrázek 4.3: Nalezené přibližné řešení. Pro výpočet jsme použili nepravidelnou síť s rozlišením $h = 0.06$. Parametr rovnice jsme zvolili $\sigma = [1, 1]$ a zdrojový člen byl volen tak abychom znali přesné řešení.

Vraťme se nyní k advekčně-difuzní rovnici. Zvolíme-li difuzní parametr příliš malý, chování metod se velmi nezmění. Pro ilustraci jsme zvolili $\varepsilon = 10^{-9}$. V tom případě stále není možné použít Jacobiho předpodmínění, ani ILU(0). Jediným rozdílem je, že konvergují také metody BiCG a BiCGstab(1). Ovšem řešení je stále velmi poškozeno. Obrázku (4.3) ukazuje, že rychlost konvergence je stále velmi nízká.

Ukazuje se, že aby dobře pracoval algoritmus ILU(0), je zapotřebí difuzní parametr alespoň řádu 10^{-3} . Potom se již v řešení neobjevují nežádoucí oscilace a všechny metody naleznou řešení. Obrázek (4.5) ukazuje, že se také rychlost konvergence metod díky předpodmínění značně urychlí.



Obrázek 4.4: Rychlost konvergence metod pro advekčně-difuzní rovnici ve 2D. Pro výpočet jsme použili nepravidelnou síť s rozlišením $h = 0.06$. Parametry rovnice jsme zvolili $\sigma = [1, 1], \varepsilon = 10^{-9}$.



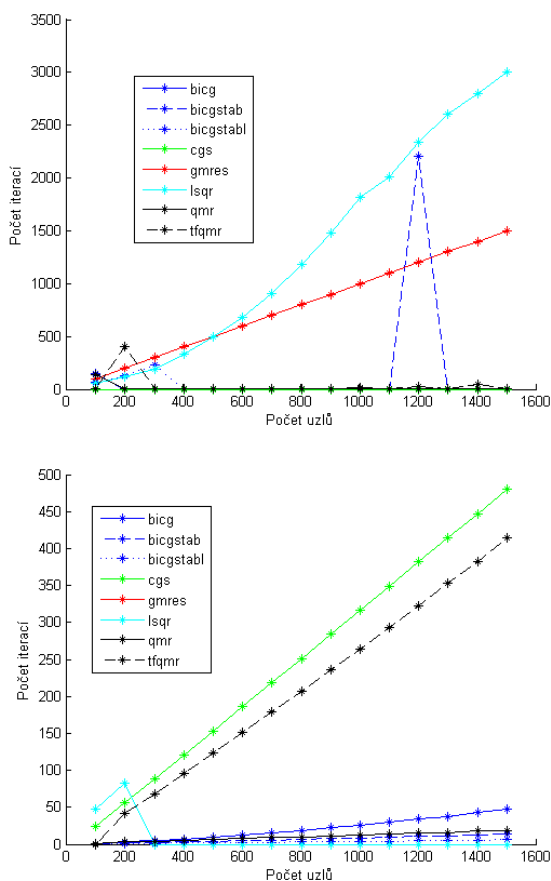
Obrázek 4.5: Rychlost konvergence metod pro advekčně-difuzní rovnici ve 2D. Pro výpočet jsme použili nepravidelnou síť s rozlišením $h = 0.06$. Parametry rovnice jsme zvolili $\sigma = [1, 1], \varepsilon = 10^{-3}$. Bylo použito předpodmínění ILU(0)

4.2 Počet iterací potřebný k dosažení dané přesnosti

Cílem tohoto experimentu je ukázat, jaká je závislost počtu iterací metod řešení soustav lineárních algebraických rovnic na jemnosti použité sítě pro různé hodnoty paramerů advektivně difuzní rovnice. Sledovali jsme kolik iterací je zapotřebí k dosažení předem dané přesnosti.

4.2.1 Rovnice v jedné dimenzi

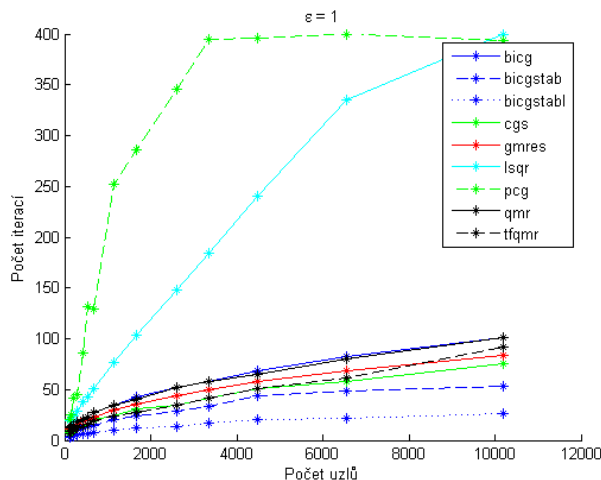
Pro tento experiment byla zvoleny parametr $\sigma = 1$ a provedli jsme výpočet pro několik parametrů ε . Ukazuje se, že v jednodimenzionálním případě roste počet potřebných iterací lineárně.



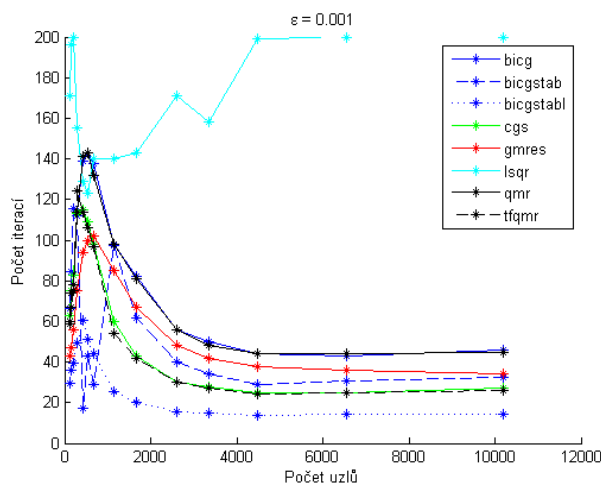
Obrázek 4.6: Počet potřebných iterací testovaných metod při výpočtu řešení advektivně-difuzní rovnice v 1D. Parametr $\sigma = 1$, nahoře $\varepsilon = 10^{-3}$, dole $\varepsilon = 10^3$.

4.2.2 Rovnice ve dvou dimenzích

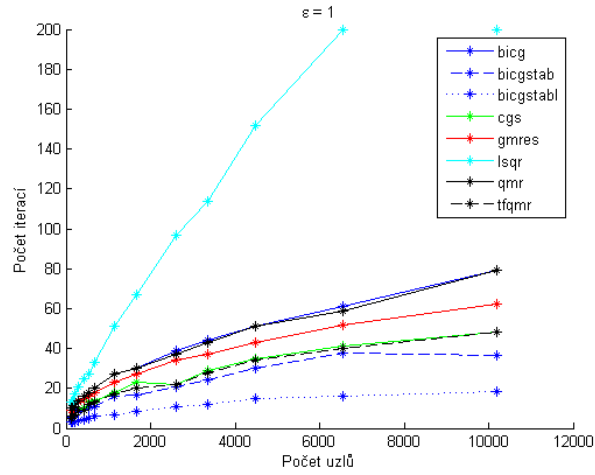
Experiment jsme provedli také s advekčně difuzní rovnicí ve dvou dimenzích. Zvolili jsme pevně σ a také tentokrát provedli výpočet pro několik parametrů ε . V tomto případě jsou však výsledky experimentu daleko zajímavější. Ukazuje se totiž, že u většiny metod počet iterací potřebných k dosažení zvolené přesnosti v případě dvourozměrné rovnice neroste, pokud platí $\varepsilon \ll \sigma$. Výjimkou je pouze metoda LSQR, která se však ukazuje jako nepřilíš vhodná pro tento problém. Uvádíme také počet iterací, potřebný pro výpočet difuzní rovnice. V případě difuzní rovnice roste počet potřebných iterací nejrychleji.



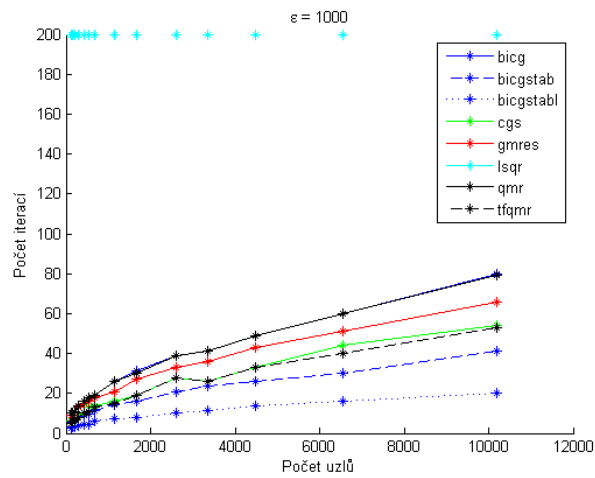
Obrázek 4.7: Počet potřebných iterací testovaných metod při výpočtu řešení difuzní rovnice ve 2d.



Obrázek 4.8: Počet potřebných iterací testovaných metod při výpočtu řešení advekčně-difuzní rovnice ve 2D. Parametry $\varepsilon = 10^{-3}$ a $\sigma = [1, 1]$.



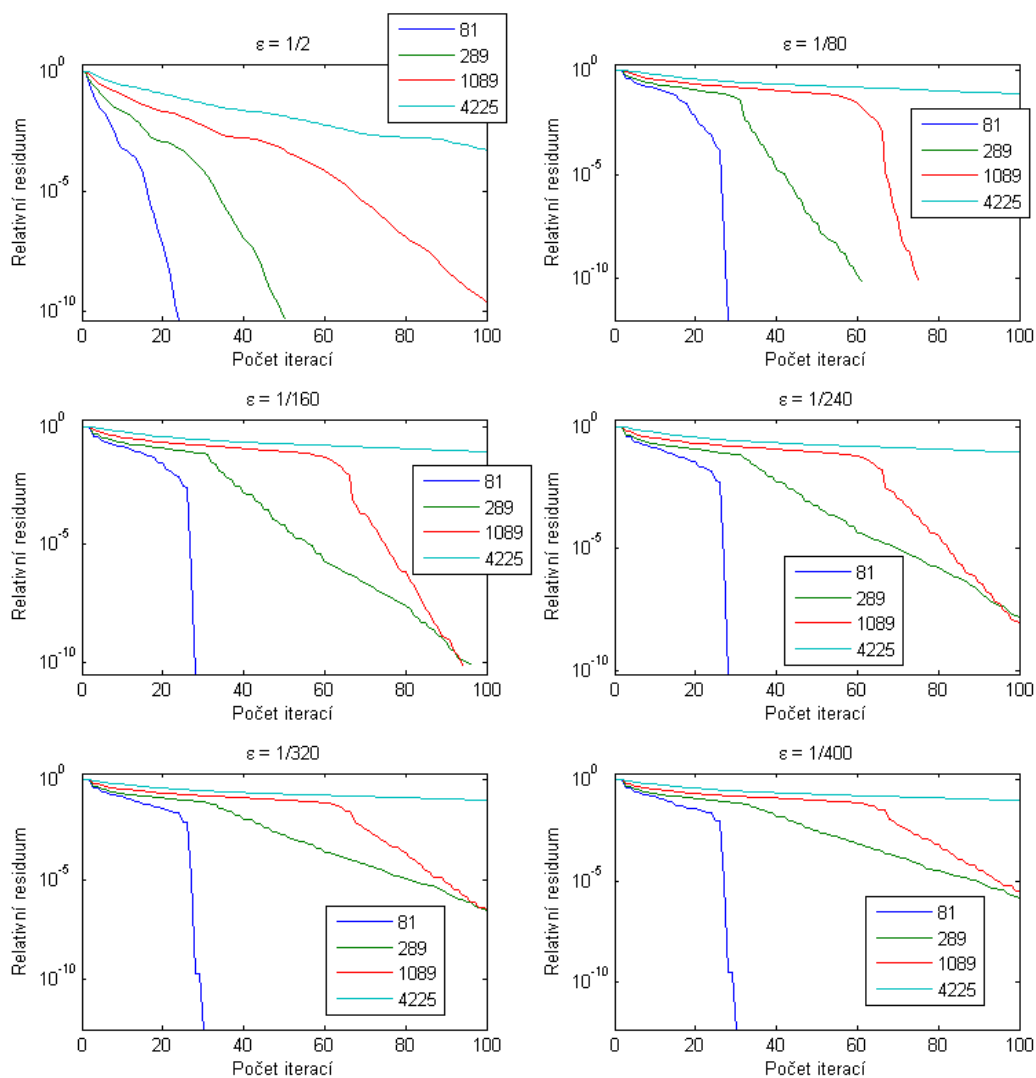
Obrázek 4.9: Počet potřebných iterací testovaných metod při výpočtu řešení advekčně-difuzní rovnice ve 2D. Parametry $\varepsilon = 1$ a $\sigma = [1, 1]$.



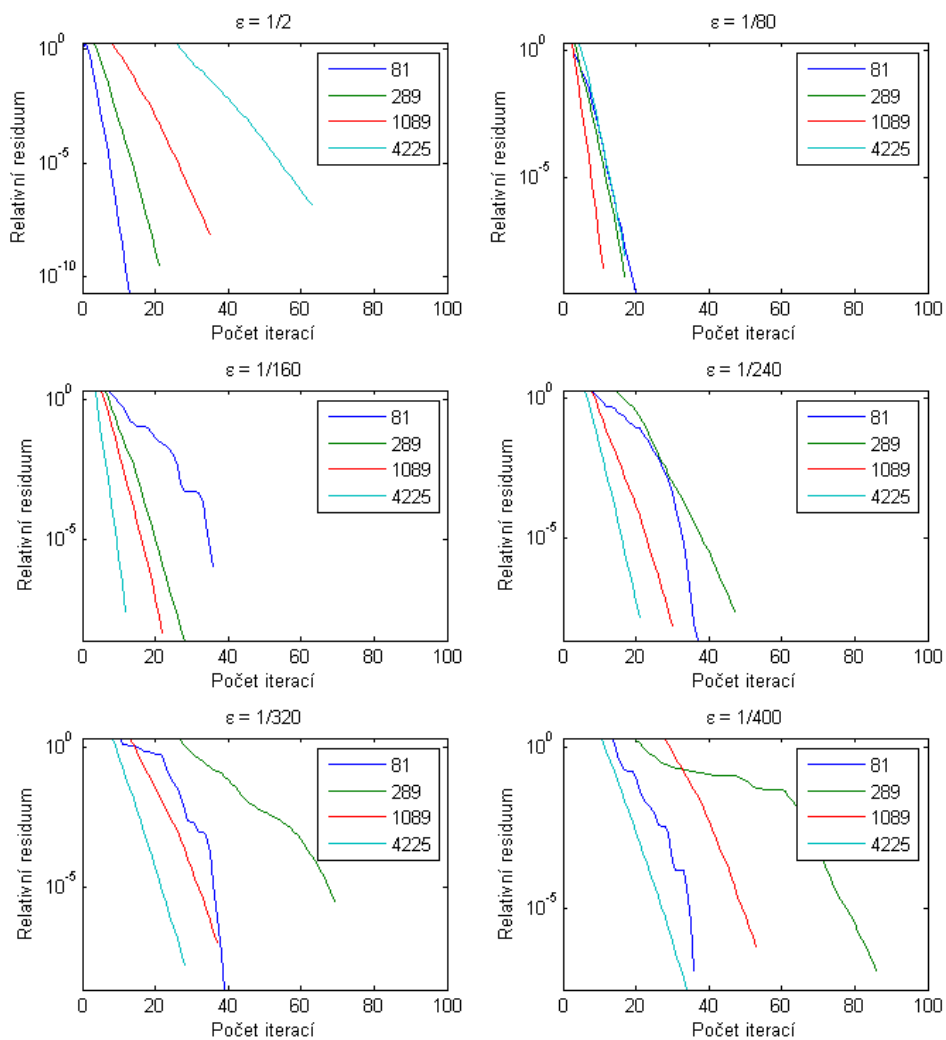
Obrázek 4.10: Počet potřebných iterací testovaných metod při výpočtu řešení advekčně-difuzní rovnice ve 2D. Parametry $\varepsilon = 10^3$ a $\sigma = [1, 1]$.

4.3 Rychlost konvergence GMRES

Podívejme se blíže na rychlost konvergence metody GMRES. Grafy (4.11, 4.12) ukazují relativní residuum v i -té iteraci metody. Pro srovnání uvádíme rychlost konvergence s předpodmíněním a bez předpodmínění. Zajímavý je především ten efekt, že s tím jak klesá velikost difuze se zrychluje konvergence soustavy pro jemnější síť. Tento efekt je patrný především pro předpodmíněnou metodu. Tam dochází dokonce k tomu, že při výpočtu na jemnější síti soustava konverguje značně rychleji než při výpočtu na hrubší síti.



Obrázek 4.11: Rychlost konvergence metody GMRES bez předpodmínění.



Obrázek 4.12: Rychlost konvergence metody GMRES s předpokmínění ILU(0).

4.4 Shrnutí numerický experimentů

Provedené numerické experimenty ukazují, že pokud při výpočtu řešení advekčně-difuzní rovnice konverguje, potom metoda BiCGstab(1) konverguje nejrychleji. Oproti tomu metoda GMRES se ukazuje jako nejvíce numericky stabilní. To je však jen jeden závěr.

Důležitější je zjištění, že pokud snižujeme difuzi, potom s tím jak zjemňujeme síť roste rychlost konvergence všech iteračních metod. Připomeň nyní, že s tím jak klesá difuzní parametr se advekčně-difuzní rovnice chová více jako hyperbolická. Toho se pokusíme využít v následující kapitole.

Kapitola 5

Hyperbolická soustava parciálních diferenciálních rovnic

Uvažujme nyní soustavu parciálních diferenciálních rovnic

$$\begin{aligned}u_t + \sigma_1 u_x + \sigma_2 u_y - \varepsilon p_x - \varepsilon q_y &= f(x, y) \\u_t + \frac{1}{T} p - \frac{1}{T} u_x &= 0 \\u_t + \frac{1}{T} q - \frac{1}{T} u_y &= 0\end{aligned}$$

kde $\sigma_1, \sigma_2, \varepsilon, T \in \mathbb{R}$ jsou kladné konstanty, $u(x, y), p(x, y), q(x, y)$ jsou hledané funkce a funkce $f(x, y)$ zdrojový člen. Její stacionární řešení zřejmě musí splňovat soustavu rovnic

$$\begin{aligned}\sigma_1 u_x + \sigma_2 u_y - \varepsilon p_x - \varepsilon q_y &= f(x, y) \\ \frac{1}{T} p - \frac{1}{T} u_x &= 0 \\ \frac{1}{T} q - \frac{1}{T} u_y &= 0\end{aligned}\tag{5.1}$$

Doplňme také okrajové podmínky. Opět budeme uvažovat pouze Dirichletovy homogenní okrajové podmínky

$$u(x, y) = 0 \quad \forall (x, y) \in \partial\Omega\tag{5.2}$$

Lze snadno ověřit, že taková soustava je ekvivalentní se rovnicemi (2.3, 1.9). Je však nezbytné zde zdůraznit, že evoluční rovnice se chová jinak, než odpovídající evoluční advekčně difuzní, resp. difuzní rovnice.

Jak jsme ukázali v předchozí kapitole, rychlost konvergence metod pro řešení soustav lineárně algebraických rovnic se zvyšuje čím více se advekčně-difuzní rovnice chová jako hyperbolická. Nyní budeme snažit ukázat, že obdobně se chová i tento systém. Díky tomu by se dal snížit počet iterací metod pro řešení soustav algebraických rovnic u soustav získaných metodou konečných prvků pro difuzní, resp. advekčně-difuzní rovnice tím, že bychom je převedli právě na tento systém. K tomuto postupu jsme se inspirovali v článku [9] a [10].

5.1 Metoda konečných prvků

Soustavu (5.1) s okrajovou podmínkou bychom mohli řešit opět metodou konečných prvků. Při použití lineárních prvků, tak jako v předchozích případech, by však tato metoda byla nestabilní. Proto musíme zvolit vhodnou metodu stabilizace. To můžeme provést například tak, že přidáme difuzní člen - takzvaná metoda přidané difuze (nebo přidané vazkosti). Budeme tedy místo soustavy (5.1) řešit tuto soustavu

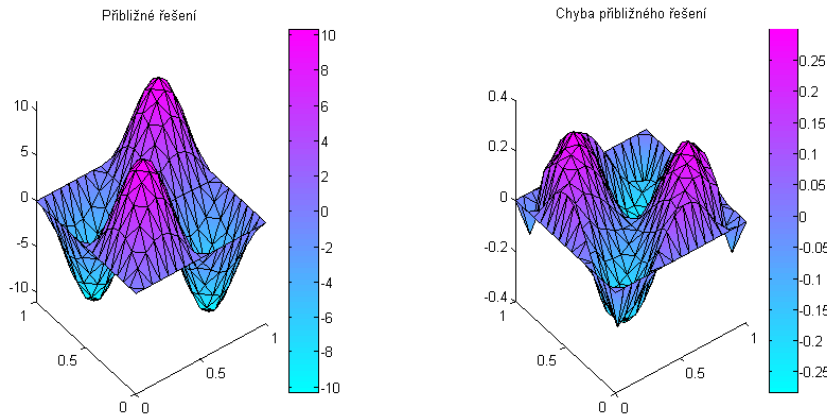
$$\begin{aligned} u_t + \sigma u_x + \sigma u_y - \varepsilon p x_x - \varepsilon p y_y &= f + \delta_1(u_{xx} + u_{yy}) \\ u_t + \frac{1}{T} p - \frac{1}{T} u_x &= \delta_2(p_{xx} + p_{yy}) \\ u_t + \frac{1}{T} q - \frac{1}{T} u_y &= \delta_3(u_{xx} + u_{yy}) \end{aligned} \quad (5.3)$$

Zde je δ další parametr rovnice, který budeme volit tak, aby metoda konečných prvků byla stabilní. Slabým řešením této soustavy nazveme funkce $u, p, q \in V_g$ takové, že splňují rovnost

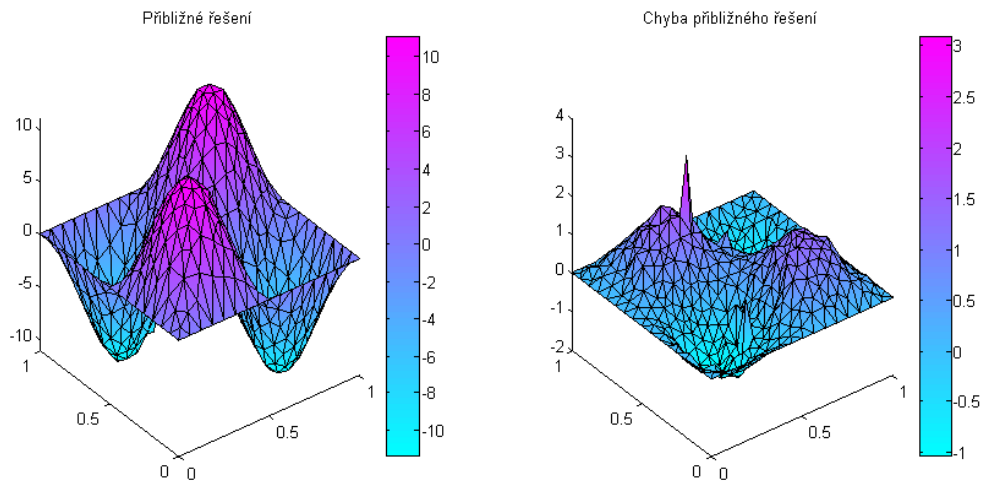
$$\begin{aligned} - \int_{\Omega} u(\sigma v_x + \sigma v_y) d\Omega - \int_{\Omega} \varepsilon p_x v d\Omega - \int_{\Omega} \varepsilon q_y v d\Omega + \int_{\Omega} \delta_1(u_x v_x - u_y v_y) d\Omega &= \int f v d\Omega \\ \int_{\Omega} p v d\Omega + \int_{\Omega} u v_y d\Omega + \int_{\Omega} \delta_2(p_x v_x - p_y v_y) d\Omega &= 0 \\ \int_{\Omega} q v d\Omega + \int_{\Omega} u v_x d\Omega + \int_{\Omega} \delta_3(q_x v_x - q_y v_y) d\Omega &= 0 \end{aligned}$$

Dále použijeme k diskretizaci metodu konečných prvků obdobně jako v předchozích případech. Na tomto místě však poznamenejme, že vzniklá soustava lineárně algebraických rovnic bude obsahovat třikrát více neznámých a také třikrát více rovnic.

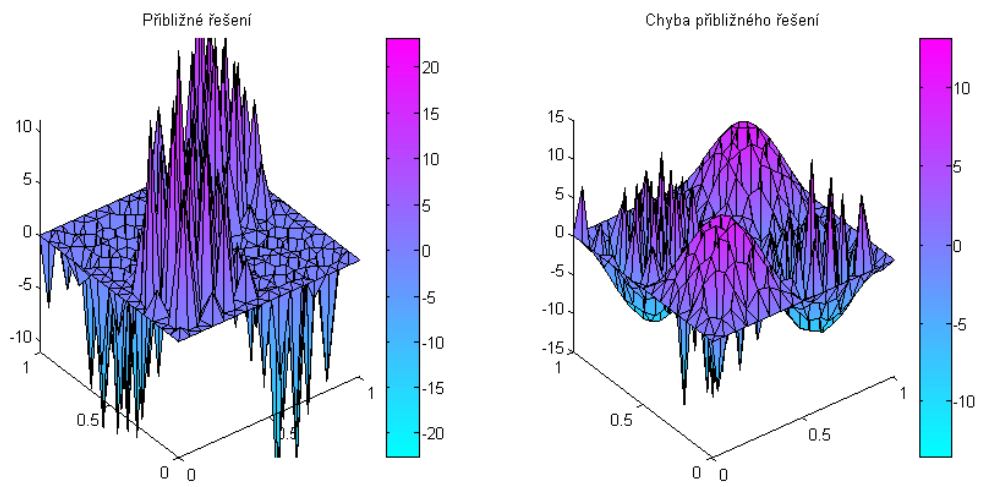
Důležitou otázkou je, jak volit parametry $\delta_1, \delta_2, \delta_3$. Obrázky (5.1), (5.2), (5.3) ukazují nalezené přibližné řešení pro různé hodnoty těchto parametrů. Ty musí být voleny tak, aby dostatečně stabilizovaly soustavu rovnic, ale aby nezanesly do řešení příliš velkou chybu. Obecně však není možné určit optimální hodnotu této přidané difuze.



Obrázek 5.1: Řešení hyperbolického systému s parametrem $\delta_{1,2,3} = 0.25$.



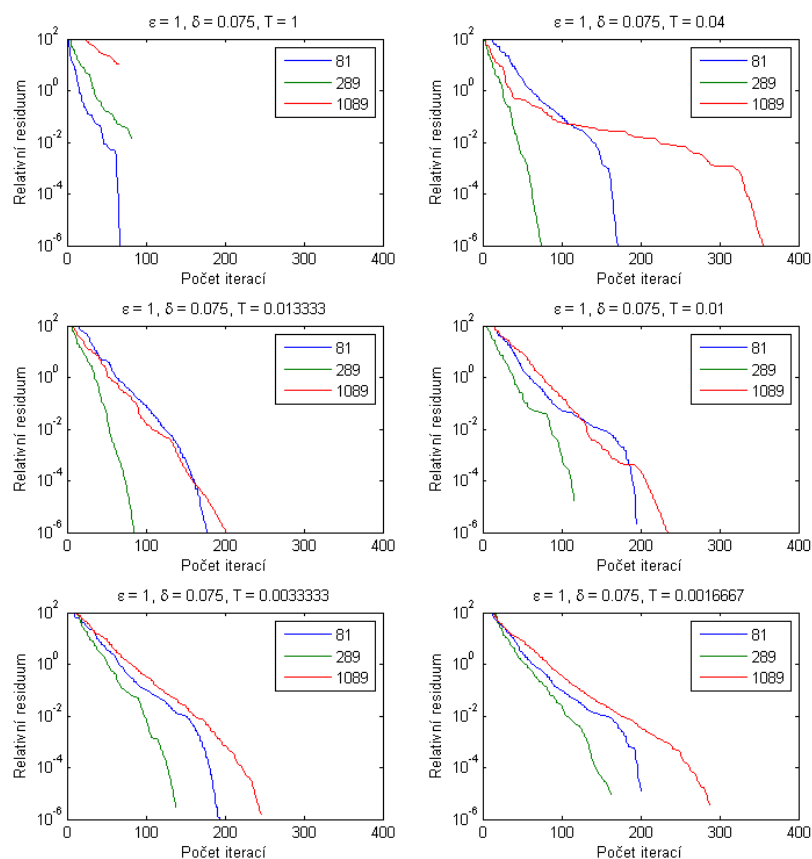
Obrázek 5.2: Řešení hyperbolického systému s parametrem $\delta_{1,2,3} = 0.1$.



Obrázek 5.3: Řešení hyperbolického systému s parametrem $\delta_{1,2,3} = 10^{-5}$.

5.2 Rychlost konvergence metody GMRES

Obrázek (5.4) ukazuje, jak se mění rychlost konvergence metody GMRES pro různé hodnoty parametru T . Z tohoto numerického experimentu je zřejmé, že pro vhodné parametry možné urychlit rychlost konvergence metody GMRES pro jemnější síť.



Obrázek 5.4: Rychlost konvergence metody GMRES pro hyperbolickou soustavu parciálních diferenciálních rovnic. Parametry byli zvoleny $\sigma = [1, 1]$, $\varepsilon = 1$, $\delta_{1,2,3} = 0.075$. V legendě jsou uvedeny počty uzlů sítě.

Kapitola 6

Závěr

V této práci jsme předvedli metody pro řešení difuzní a advekčně-difuzní rovnice v jedné a ve dvou dimenzích. Jedním ze zásadních bodů těchto metod je výpočet soustavy lineárně algebraických rovnic.

Pomocí několika numerických experimentů jsme ukázali, že pokud řešíme advekčně-difuzní rovnici, potom se rychlost konvergence iteračních metod řešení soustav lineárně algebraických rovnic urychluje s tím, jak se zjemňuje síť. To byl jeden z cílů práce.

Dalším cílem práce bylo ukázat, jak lze tento efekt využít. Představili jsme tedy hyperbolickou soustavu parciálních diferenciálních rovnic, na kterou lze převést jak difuzní, tak advekčně-difuzní rovnici. Ukazuje se, že pro vhodné parametry se také u této soustavy urychluje konvergence iteračních metod pro řešení soustav lineárně algebraických rovnic. Otázkou však zůstává, zda existují takové parametry, pro které by se rychlost konvergence urychlila natolik aby se vyplatilo pracovat se vzniklými třikrát většími maticemi. To je však již nad rámec této práce.

Literatura

- [1] H. Elman, D. Silvester, A. Wathen: *Finite Elements and Fast Iterative Solvers*. Oxford University Press, 2005.
- [2] H. Elman, D. Silvester, A. Wathen: *Iterative Methods for Problems in Computational Fluid Dynamics*. Report CS-TR-3675, UMIACS-TR-96-58, 1996.
- [3] Elman, M. Chernesky: *Ordering Effects on Relaxation Methods Applied to Discrete One-Dimensional Convection-Diffusion Equation*. SIAM Journal on Numerical Analysis, Vol. 30, No. 5, 1993, pp. 1268-1290.
- [4] N. Nachtigal, S. Reddy, L. Trefethen: *How Fast Are Nonsymmetric Matrix Iterations?* SIAM Journal of Matrix Analysis and Applications, Vol. 13, No. 3, 1992, pp. 778-795.
- [5] J.E. Akin: *Finite Element Analysis with Error Estimators*, Butterworth Heinemann, 2005
- [6] J. D. Tebbens, I. Hnětynková, M. Plešinger, Z. Strakoš, P. Tichý : *Analýza metod pro maticové výpočty. Základní metody*, Matfyzpress, 2012
- [7] P. Tichý: *O některých otevřených problémech v krylovovských metodách*, Disertační práce, Matematicko - fyzikální fakulta Univerzity Karlovy, Praha, 2002
- [8] Ir. A. Segal: *Finite element methods for the incompressible Navier-Stokes equations*, Delft University of Technology, 2012
- [9] H. Nishikawa: *A first-order system approach for diffusion equation. I: Second-order residual-distribution schemes*, Journal of Computational Physics 227, 2007, p. 315–352
- [10] H. Nishikawa: *A First-Order System Approach for Diffusion Equation. II: Unification of Advection and Diffusion*, Journal of Computational Physics, 2009
- [11] S. Míka, P. Příkryl, M. Brandner,: *Speciální numerické metody. Numerické metody řešení okrajových úloh pro diferenciální rovnice*, Vydavatelský servis, Plzeň, 2006