

VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra aplikované matematiky

Multigridy v 1 dimenzi

Bakalářská práce

2008

Rostislav Hrtus

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě 8. května 2008

.....

Rád bych na tomto místě poděkoval panu Ing. Daliboru Lukášovi, Ph.D. za jeho velkou pomoc, trpělivost a motivaci při řešení této práce.

Abstrakt

Pro řešení soustav lineárních rovnic je známo mnoho postupů. Přímé řešiče, jako například Gaussova eliminační metoda, nejsou příliš vhodné pro řešení rozsáhlých soustav, z důvodu velkého počtu operací nutných k dosažení výsledku. Je proto snaha nalézt rychlejší metody, metody iterační. V práci je na 1D modelu průhybu struny popsáno několik metod. Jednou z nich je Richardsonova metoda, dále metoda TwoGrid (metoda dvou sítí), která je založena na Richardsonově iterační metodě a Gaussově eliminaci. Nejrychlejší z analyzovaných metod je pak Multigrid metoda, která kombinuje výhody Gaussovy eliminační metody a Richardsonovy metody.

Klíčová slova: Richardson, Multigrid, V-cyklus

Abstract

There are various procedures known to linear equations systems solving. Direct solvers, such as Gaussian elimination method, are not so suitable for large systems solving, because of large number of iterations that are necessary to obtain the result. There is a tendency to find faster methods, iteration methods. In this work, several methods are described. One of them is the Richardson method, the TwoGrid method follows. TwoGrid method is based on the Richardson iteration method and Gaussian elimination. The Multigrid method is the fastest from all analyzed methods. It combines advantages of Gaussian elimination and Richardson method.

Keywords: Richardson, Multigrid, V-cycle

Použité značení

H, h - kroky diskretizace

I - jednotková matice

M - iterační matice iterační metody

N - vektor upravené pravé strany iterační metody

P_{Hh} - matice prolongace

$R_{d+1}, d \in \mathbb{N}$ - Lagrangeův tvar zbytku Taylorova polynomu

$T_d(x), d \in \mathbb{N} \cup \{0\}$ - Taylorův polynom řádu d

$e_i, i \in \{1, \dots, n\}$ - vlastní vektory matice

$h_i, i \in \{1, \dots, n\}$ - hierarchie diskretizací

k - k -tá iterace daného parametru

$m, m \in \mathbb{N}$ - počet iterací iterační metody

$p, p \in \mathbb{N}$ - úroveň zanoření v-cyklu

$x_i, i \in \{1, \dots, n\}$ - bod diskretizovaného intervalu

α - tlumící parametr

β_i - vlastní číslo matice po aplikování iterační metody

β_i^k - vlastní číslo matice pro k -tý iteraci iterační metody

ϵ - přesnost

$\kappa(A) = \lambda_{max}/\lambda_{min}$ - číslo podmíněnosti pro matici symetrickou a pozitivně definitní

$\lambda_i, i \in \{1, \dots, n\}$ - vlastní čísla matice

λ_{max} - největší vlastní číslo

λ_{min} - nejmenší vlastní číslo

Obsah

1	Úvod	4
2	Použitý matematický aparát	5
2.1	Iterační metody řešení soustav rovnic	5
2.2	Konvergence iterační metody	5
3	Matematický model struny pro tuto úlohu	6
3.1	Rovnice průhybu struny	6
4	Metody řešení	10
4.1	Gaussova eliminační metoda (GEM)	10
4.2	Richardsonova metoda	11
5	Rychlejší metody řešení	16
5.1	Two Grid metoda	16
5.2	Multigrid metoda	20
5.3	Efektivita Multigridu- odvození složitosti	24
6	Porovnání metod přímých, iteračních a multigridu	26
6.1	Aplikace algoritmů na počítači	26
6.2	Efektivita jednotlivých metod	32
7	Závěr	34
8	Reference	35

Seznam obrázků

1	Průhyb struny	7
2	Diskretizace intervalu	7
3	Spektra matice A pro různé diskretizační kroky	9
4	Konvergence koeficientů	14
5	Spektra iterační matice Richardsonovy metody	14
6	Diskretizace pro metodu Two Grid	17
7	Spektra iterační matice metody Two Grid	19
8	Konvergence koeficientů metody Two Grid	20
9	Diskretizace pro metodu Multigrid	21
10	V-cyklus metody Multigrid	22
11	Konvergence koeficientů metody Multigrid (k= 100)	23
12	Konvergence koeficientů metody Multigrid (k= 10)	24
13	Pravá strana- zátěžová funkce	27
14	Řešení průhybu struny	27
15	Konvergence chyby řešení Richardsonovy metody	28
16	Divergence chyby řešení Richardsonovy metody- jiná volba α	28
17	Příspěvky jednotlivých diskretizací Two Grid (m= 3)	29
18	Příspěvky jednotlivých diskretizací Two Grid (m= 9)	29
19	Výsledek Multigridu pro různé zanoření V-cyklu	30
20	Divergence Multigridu vlivem tlumícího koeficientu	31

1 Úvod

V této bakalářské práci se budu věnovat metodám pro výpočet rozsáhlých soustav lineárních rovnic, které vznikají diskretizací jednodimenzionální úlohy průhybu struny. Tato oblast se začala rozvíjet pro potřebu inženýrských aplikací. Do aplikací multigridu spadá mnoho oblastí, převážně ty, kde máme nějaký objekt, a chceme zkoumat jeho vlastnosti v závislosti na silách na něj působících. Tyto zkoumané objekty mohou být z nehomogenního materiálu, mohou být velmi složité a problém nastává, když chceme takovýto objekt rozložit na menší, snáze řešitelné oblasti. Pokud jsou oblasti složité a chceme získat relevantní výsledek, plyne z toho potřeba řešit velmi velké soustavy. Aby byl výpočet proveditelný v reálném čase, je potřeba tyto soustavy řešit vhodnými metodami.

Práce ukáže základní značení a matematické nástroje, se kterými jsem pracoval. V kapitole 3 jsem odvodil matematický model této úlohy průhybu struny, kde se dozvíte, jak jsem postupoval při odvozování a jak vypadá celá soustava lineárních rovnic, kterou budu chtít řešit. Kapitola 4 je o možnostech řešení této soustavy, jako první metoda je uvedena Gaussova eliminace, což je nástroj značně neefektivní pro velké soustavy nerovnic. Dále se zmíním o iterativní metodě řešení, konkrétně půjde o Richardsonovu metodu. Kapitola 5 pak vysvětlí jak lze postupy z předchozí kapitoly kombinovat a tím docílit mnohem rychlejšího řešení. Bude se jednat o metodu Two Grid, která spojuje výhodu Richardsonovy metody a Gaussovy eliminační metody. Posléze vysvětlím, že se dá metodu Two Grid ještě zlepšit tím, že použiji hierarchii lineárních soustav pro výpočet ve více úrovních, čímž je možno získat velmi rychlý nástroj pro řešení těchto soustav a tím je Multigrid. V poslední kapitole 6 ukážu efektivitu jednotlivých metod a prokážu smysl této práce na praktických zkouškách algoritmů zde uvedených.

2 Použitý matematický aparát

Tato kapitola bude obsahovat základní matematický aparát použitý v této práci.

2.1 Iterační metody řešení soustav rovnic

Mějme dānu soustavu lineárních rovnic

$$Au = b; A \in \mathbb{R}^{n \times n}, u, b \in \mathbb{R}^n.$$

Je-li matice A regulární, pak

$$\exists! u : Au = b.$$

Iterační metodou budu nazývat zobrazení

$$u^{k+1} = \phi(u^k), k \in \mathbb{N} \cup 0.$$

Nechť je pak iterační metoda tvaru

$$u^{k+1} = Mu^k + N,$$

kde M je iterační matice a N je vektor.

2.2 Konvergence iterační metody

Iterační metoda ϕ se nazývá konvergentní, jestliže

$$\forall b \in \mathbb{R}^n : \lim_{k \rightarrow \infty} u^k = u$$

a je nezávislá na počáteční iteraci u^0 .

3 Matematický model struny pro tuto úlohu

3.1 Rovnice průhybu struny

Chceme řešit

$$-u''(x) = f(x), x \in (0, 1) \quad (3.1.1)$$

$$u(0) = u(1) = 0. \quad (3.1.2)$$

Na Obrázku ?? uvádím tuto úlohu v grafické podobě.

3.1.1 Analytické řešení

Analyticky v 1D lze často řešit za předpokladu pravé strany, kterou umíme integrovat. Necht' například $f(x) = 1, x \in (0, 1), u(0) = u(1) = 0$.

$$\begin{aligned} -u'' &= 1 \\ u' &= -\int 1 dx = -(x + C_1) \\ u &= -\int (x + C_1) dx = -\left(\frac{x^2}{2} + C_1 x + C_2\right) = -\frac{x^2}{2} - C_1 x - C_2 \end{aligned}$$

Dosazení počátečních podmínek pro zjištění C_1 a C_2

$$\begin{aligned} 0 &= 0 - C_2 \\ 0 &= \frac{1}{2} + C_1 - C_2 \\ C_1 &= -\frac{1}{2}, C_2 = 0. \end{aligned}$$

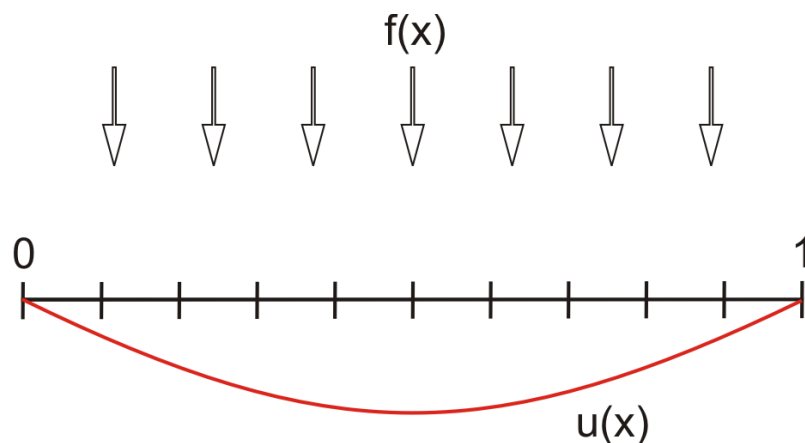
Řešení diferenciální rovnice (??) s podmínkami (??) je v tomto tvaru

$$u(x) = -\frac{x^2}{2} + \frac{1}{2}, x \in (0, 1).$$

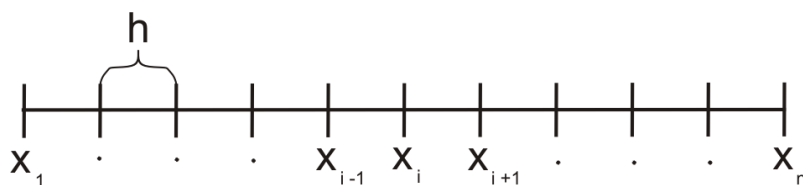
3.1.2 Numerické řešení

Mohou nastat případy, kdy nebudu znát primitivní funkci k pravé straně rovnice (??). Tato skutečnost může vést k tomu, že nebudeme schopni tuto diferenciální rovnici řešit a v tomto případě je nutno použít numerické metody. V tomto odstavci se budu věnovat odvození soustavy lineárních rovnic, která bude dobře aproximovat řešení (??) s podmínkami (??).

Pro numerické řešení je důležitá diskretizace. Diskretizací rozumíme dělení intervalu, na kterém řeším úlohu, s ekvidistantním krokem $h = x_i - x_{i+1}, \forall i \in \{1, \dots, n-1\}, h > 0$. V tomto případě použiji diskretizaci intervalu $\langle 0, 1 \rangle$. Na Obrázku ?? je vidět, jaké použiji



Obrázek 1: Průhyb struny



Obrázek 2: Diskretizace intervalu

značení u diskretizace.

Pro řešení numerickou metodou je nutné sestavit matici zachycující síť bodů na intervalu, na kterém úlohu řeším. Matice bude zachycovat tzv. stencil, který vyjadřuje souvislost mezi body struny, stencil bude velikosti 3×1 a pro jeho odvození diferenciální rovnici aproximují pomocí rozvoje Taylorovy řady pro x_{i+1} a x_{i-1} a z nich pak x_i vyjádřím. Taylorova řada, pro funkci mající derivaci řádu n je tvaru

$$T_n(x_{i+1}) = u(x_i) + h \frac{u'(x_i)}{1!} + h^2 \frac{u''(x_i)}{2!} + \dots + h^n \frac{u^{(n)}(x_i)}{n!} + R_{n+1}(\xi), \quad h > 0, \xi \in (x_i, x_{i+1}),$$

a tedy

$$T_n(x_{i+1}) = \sum_{j=0}^n h^j \frac{u^{(j)}(x_i)}{j!} + R_{n+1}(\xi),$$

kde

$$R_{n+1}(\xi) = h^{n+1} \frac{u^{(n+1)}(\xi)}{(n+1)!}$$

je Lagrangeův tvar zbytku.

V mém případě

$$u(x_{i+1}) = u(x_i) + hu'(x_i) + \frac{h^2}{2!}u''(x_i) + R_2(\xi_1), \quad (3.1.3)$$

$$u(x_{i-1}) = u(x_i) - hu'(x_i) + \frac{h^2}{2!}u''(x_i) + R_2(\xi_2). \quad (3.1.4)$$

Všimněme si nyní, že vzniká první chyba matematického modelu zanedbáním R_2 . Aproximujme tedy jen druhou derivaci.

Z rovnice (??) si vyjádřím

$$-u''(x_i) \approx \frac{2(u(x_{i+1}) - u(x_i) - hu'(x_i))}{h^2} \quad (3.1.5)$$

a z rovnice (??) pak

$$-u''(x_i) \approx \frac{2(u(x_{i-1}) - u(x_i) + hu'(x_i))}{h^2}. \quad (3.1.6)$$

Pokud pak rovnice (??) a (??) sečtu, získám

$$-2u''(x_i) \approx \frac{2(u(x_{i+1}) - u(x_i) - hu'(x_i) + u(x_{i-1}) - u(x_i) + hu'(x_i))}{h^2}$$

upravím

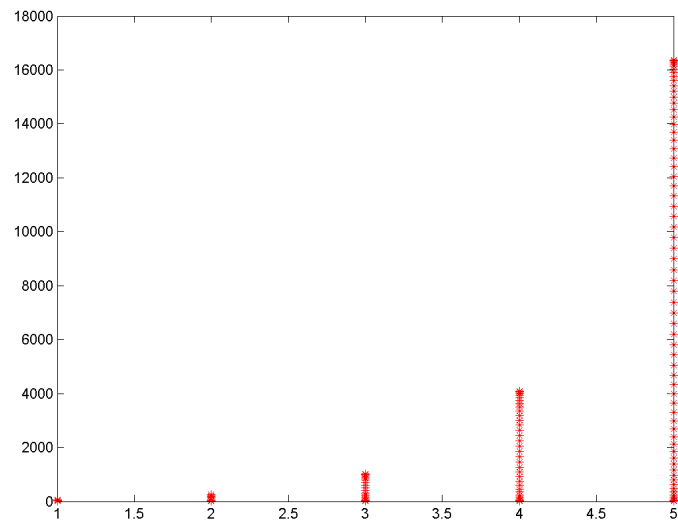
$$2u''(x_i) \approx \frac{-2u(x_{i-1}) + 4u(x_i) - 2u(x_{i+1}))}{h^2}$$

a po další úpravě dostanu výslednou rovnici průhybu struny

$$u''(x_i) \approx \frac{-u(x_{i-1}) + 2u(x_i) - u(x_{i+1}))}{h^2}. \quad (3.1.7)$$

Z této rovnice lze odvodit, jak bude vypadat matice (s ohledem na počáteční podmínky), se kterou budu pracovat. Celá soustava rovnic pro (??) s podmínkami (??) pak vypadá následovně

$$\underbrace{\frac{1}{h^2} \begin{pmatrix} 2 & 0 & & \cdots & 0 \\ 0 & 2 & -1 & & \vdots \\ & -1 & 2 & -1 & \\ & & \ddots & \ddots & \ddots \\ & & & -1 & 2 & -1 \\ \vdots & & & & -1 & 2 & 0 \\ 0 & \cdots & & & & 0 & 2 \end{pmatrix}}_A \underbrace{\begin{pmatrix} u_1(x_1) \\ \vdots \\ u_n(x_n) \end{pmatrix}}_u \approx \underbrace{\begin{pmatrix} b_1(x) \\ \vdots \\ b_n(x) \end{pmatrix}}_b. \quad (3.1.8)$$



Obrázek 3: Spektra matice A pro různé diskretizační kroky

Matice A je pásová, třídiagonální, což je pro řešení numerickými metodami velmi výhodné. V dalším postupu se bude hodit znalost spektra, proto jsem je na Obrázku ?? ukázal pro různé kroky diskretizace $h = 0.25, 0.125, 0.0625, 0.03125, 0.015625$. Spektrum matice je množina vlastních čísel $\lambda_i, i \in \{1, \dots, n\}$. Všimněme si, jak rychle rostou maximální vlastní čísla a minimální vlastní čísla matic zůstávají zhruba stejné. Tedy je potřeba vědět, jaké je číslo podmíněnosti, což je poměr největší hodnoty k té nejmenší z celého spektra matice. Půjde tedy o to, aby číslo podmíněnosti $\kappa(A) = \lambda_{max}/\lambda_{min}$, kde λ_{min} je nejmenší vlastní číslo a λ_{max} největší vlastní číslo, bylo co nejmenší- bude to stěžejní pro řešení dané soustavy lineárních rovnic. Toho dosáhnu později užitím iteračních metod, kterými se tyto úpravy provádějí a to nejčastěji pomocí předpodmiňovacích metod.

4 Metody řešení

4.1 Gaussova eliminační metoda (GEM)

Gaussova eliminační metoda slouží pro řešení soustavy rovnic, řadí se mezi přímé metody, tedy výsledku dosáhneme po konečném počtu kroků. GEM sestává ze dvou kroků, jsou jimi dopředná redukce a zpětná substituce. Dopředná redukce znamená úpravu matice soustavy na schodový tvar řádkovými úpravami. Zpětná substituce řeší již soustavu s maticí ve schodovém tvaru zpětným dosazováním. Jak víme, výpočetní náročnost GEM je $O(n^3)$ neboť dopředná redukce vyžaduje počet násobení

$$\sum_{k=1}^{n-1} (n-k)^2 \leq Cn^3, C = konst.$$

a zpětná substituce vyžaduje počet operací

$$\sum_{k=1}^{n^2} (n-1) \leq Cn^3, C = konst.$$

Pro matici (pásovou) A naší soustavy (??) však lze GEM implementovat tak, že platí $O(sn^2)$, kde s je šířka pásu (konstanta).

4.2 Richardsonova metoda

Richardsonova metoda je zástupcem metod iteračních a danou soustavu lineárních rovnic (??) řeší s určitou přesností. Po nalezení řešení, které vyhovuje dané přesnosti, výpočet končí.

Následuje odvození iterační matice Richardsonovy metody, přičemž chci dosáhnout tvaru $u^{k+1} = Mu^k + N$, kde M je iterační matice, kterou hledám a N je vektor závislý pouze na pravé straně a volbě tlumícího parametru α , viz níže.

$$\begin{aligned}
 u^0 &\in (0, \dots, 1) \\
 r &= b - Au^0 \\
 u^1 &= u^0 + \alpha r \\
 u^1 &= u^0 + \alpha(b - Au^0) \\
 u^1 &= u^0 + \alpha b - \alpha Au^0 \\
 u^1 &= \underbrace{(I - \alpha A)}_M u^0 + \underbrace{\alpha b}_N \\
 r^{k+1} &= b - Au^k \\
 u^{k+1} &= \underbrace{(I - \alpha A)}_M u^k + \underbrace{\alpha b}_N.
 \end{aligned} \tag{4.2.1}$$

Za řešení s požadovanou přesností pak budeme považovat $u = u^{k+1}$, při splnění podmínky

$$\bullet \quad \|Au^{k+1} - b\| \leq \epsilon \|b\|, \epsilon \rightarrow 0_+.$$

Pokud je zadán maximální počet iterací $k = k^{max}$, kde k^{max} je maximální počet iterací, pak za řešení považují $u^{k^{max}}$ a to nehledě na chybu řešení.

Pozastavme se u hledání vhodného koeficientu α .

Věta 4.1 *Je-li α takové, že*

$$0 < \alpha < \frac{1}{\lambda_{max}}, \tag{4.2.2}$$

pak Richardsonova metoda konverguje.

Důkaz. Za předpokladu, že A je symetrická, pozitivně definitní matice a $n \in \mathbb{N}$.

$$\begin{aligned}
 Au &= b; A \in \mathbb{R}^{n \times n}; u, b \in \mathbb{R}^n. \\
 u^{k+1} &= \underbrace{(I - \alpha A)}_M u^k + \underbrace{\alpha b}_N
 \end{aligned}$$

Chci dokázat, že přibližné řešení soustavy (??) pomocí Richardsonovy metody konverguje, t.j. $u^k \rightarrow u$ pro volbu

$$0 < \alpha < \frac{1}{\lambda_{max}}. \quad (4.2.3)$$

Jestliže matice A je symetrická a reálná, pak označím vlastní čísla a vlastní vektory (λ_i, e_i) , $i \in (1, \dots, n)$. Vlastní čísla řadím vzestupně

$$\lambda_i < \lambda_{i+1}, \quad i \in (1, \dots, n-1).$$

Pak mohu psát

$$\begin{aligned} u &= \sum_{i=1}^n \beta_i e_i, \\ u^k &= \sum_{i=1}^n \beta_i^k e_i. \end{aligned} \quad (4.2.4)$$

Platí

$$Au = A \sum_{i=1}^n \beta_i e_i = A \sum_{i=1}^n \beta_i \lambda_i e_i, \quad (4.2.5)$$

a

$$Au^k = A \sum_{i=1}^n \beta_i^k e_i = A \sum_{i=1}^n \beta_i^k \lambda_i e_i. \quad (4.2.6)$$

Korekce řešení v k -té iteraci je

$$\begin{aligned} u^{k+1} &= u^k + \alpha r^{k+1} = u^k + \alpha(b - Au^k) = \\ &= u^k + \alpha(Au - Au^k). \end{aligned} \quad (4.2.7)$$

Z (??) a (??) po dosazení do (??) získám rovnici

$$\sum_{i=1}^n \beta_i^{k+1} e_i = \sum_{i=1}^n \beta_i^k e_i + \alpha \left(\sum_{i=1}^n \beta_i \lambda_i e_i - \sum_{i=1}^n \beta_i^k \lambda_i e_i \right) \quad (4.2.8)$$

Matice A je pozitivně definitní, tedy vlastní čísla $\lambda_i, i \in (1, \dots, n)$ jsou kladná. Nyní přenásobím rovnici (??) vektorem e_j , přičemž $e_j : e_j \perp e_i, e_i \neq e_j, j \in (1, \dots, n)$ a upravím

$$\begin{aligned} \sum_{i=1}^n \beta_i^{k+1} \|e_j\|^2 &= \sum_{i=1}^n \beta_i^k \|e_j\|^2 + \alpha \left(\sum_{i=1}^n \beta_i \lambda_i \|e_j\|^2 - \sum_{i=1}^n \beta_i^k \lambda_i \|e_j\|^2 \right) \\ \beta_j^{k+1} &= \beta_j^k + \alpha \beta_j \lambda_j - \alpha \beta_j^k \lambda_j \\ \beta_j^{k+1} - \beta_j &= \beta_j^k + \alpha \beta_j \lambda_j - \alpha \beta_j^k \lambda_j - \beta_j \\ \beta_j^{k+1} - \beta_j &= \beta_j^k (1 - \alpha \lambda_j) + \beta_j (\alpha \lambda_j - 1) \\ \beta_j^{k+1} - \beta_j &= (1 - \alpha \lambda_j) (\beta_j^k - \beta_j). \end{aligned} \quad (4.2.9)$$

Zvolím-li $0 < \alpha < \frac{1}{\lambda_{max}}$, pak:

$$\begin{aligned} \left| \beta_j^{k+1} - \beta_j \right| &= \underbrace{(1 - \alpha \lambda_j)}_{>0} \left| \beta_j^k - \beta_j \right| < 1 \left| \beta_j^k - \beta_j \right|. \\ \Rightarrow 0 < \alpha \lambda_j < \frac{\lambda_j}{\lambda_{max}} &\leq 1. \end{aligned} \quad (4.2.10)$$

Posloupnosti koeficientů $(\beta_j^k)_{k=1}^{\infty}$ tedy konvergují

$$\lim_{k \rightarrow \infty} \beta_j^k = \beta_j. \quad (4.2.11)$$

Z čehož plyne

$$u^k \rightarrow u. \quad \blacksquare$$

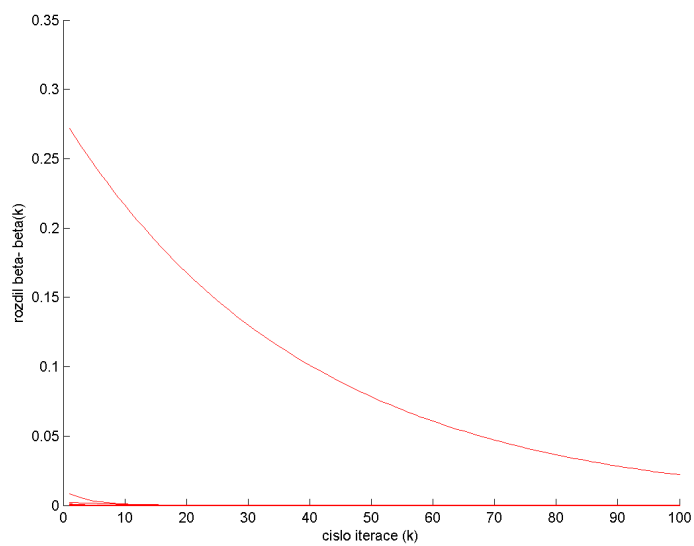
$$\begin{aligned} \left| \beta_j^{k+1} - \beta_j \right| &= \underbrace{(1 - \alpha \lambda_j)}_{\text{rychlost konvergence}} \left| \beta_j^k - \beta_j \right| < 1 \left| \beta_j^k - \beta_j \right| \\ 0 < (1 - \alpha \lambda_j) &< 1. \end{aligned} \quad (4.2.12)$$

Pak upravím nerovnici

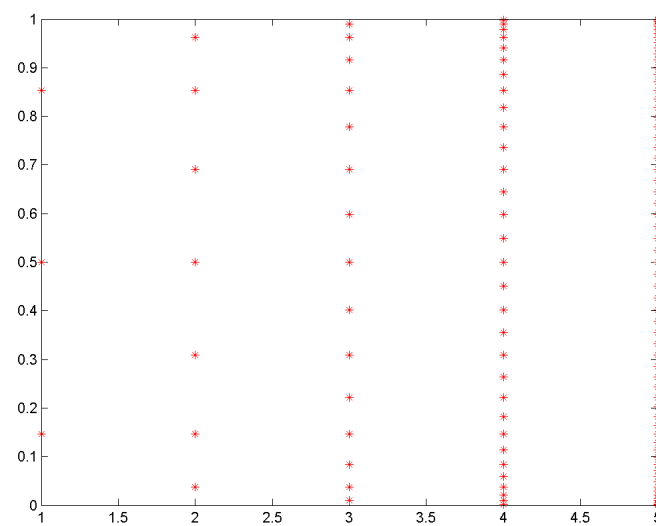
$$\begin{aligned} 0 < 1 - \alpha \lambda_j < 1. \\ 1 > \alpha \lambda_j > 0. \end{aligned} \quad (4.2.13)$$

Rychlost konvergence závisí na volbě α , jak vidíme z nerovnice (??), zaručená konvergence je pro volbu $\alpha < \frac{1}{\lambda_{max}}$, avšak bez důkazu uvedu, že optimální konvergence bych docílil volbou $\alpha = \frac{2}{\lambda_{max} + \lambda_{min}}$.

Rychlost konvergence ukazují na Obrázku ??, kde je zvolen krok $h = 0.1$ a pro počet iterací $k = 100$ konvergují koeficienty v jednotlivých iteracích k nule. Všimněme si, že hlavní obtížnost řešení tkví v minimalizaci koeficientu s nejvyšší hodnotou, samozřejmě, pokud bychom jej smazali, bude vyčnívat další koeficient, avšak ne až do takové míry. Čím větší máme soustavu, tím je tento koeficient větší, tedy metoda pak konverguje mnohem pomaleji. Na Obrázku ?? je jasně vidět, jak se zlepšilo číslo podmíněnosti vůči původní matici. Krok byl zvolen opět $h = 0.25, 0.125, 0.0625, 0.03125, 0.015625$. Podařilo se srazit maximální vlastní čísla iterační matice níže.



Obrázek 4: Konvergence koeficientů



Obrázek 5: Spektra iterační matice Richardsonovy metody

Výpis metody vytvořené v Matlabu vidíme níže.

```
function [u, it]= Richardson(A,b,epsilon,alfa,u0,itmax)
u=u0;
it = 0;
r= b- A* u;
normr0= norm(r);

if (itmax==0) itmax= inf;
end
hold on
while (norm(r)/ normr0 > epsilon) && (it < itmax)
    u= u+ alfa* r;
    r= b- A* u;
    it = it + 1;
end
```

Výpis 1: Richardsonova metoda

5 Rychlejší metody řešení

5.1 Two Grid metoda

Tato metoda, jak již název napovídá, bude řešit danou soustavu (??) pomocí dvou sítí. Celá procedura spočívá v použití Richardsonovy metody a GEM na různých diskretizacích. Mějme tedy dvě diskretizace popsané parametry

$$h, H; h = \frac{1}{2}H.$$

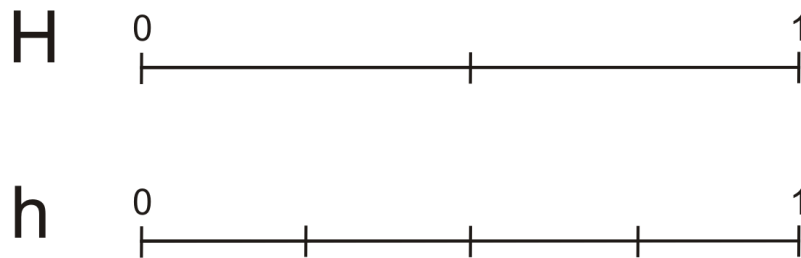
Tyto diskretizace jsem naznačil na Obrázku ?? . Necht' také matice A s diskretizací h má označení A_h a matice s diskretizací H má označení A_H . Stejně tak budu značit u_h, u_H, f_h, f_H pro příslušné diskretizace. Metoda Two Grid pracuje tak, že se provede Richardsonova metoda na jemnější síti, následně se spočítá reziduum tohoto řešení. Reziduum je třeba přepočítat na hrubší síť, což se zajistí tzv. maticí prolongace P_{Hh} . Matice prolongace má rozměry $(2n - 1) \times n$ a vypadá následovně

$$P_{Hh} = \begin{pmatrix} 0 & & & & & \dots & 0 \\ \frac{1}{2} & \frac{1}{2} & & & & \dots & 0 \\ 0 & 1 & 0 & & & \dots & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & & & \dots & 0 \\ 0 & 0 & 1 & 0 & & \dots & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 & \dots & 0 \\ 0 & \dots & 0 & \ddots & \ddots & 0 & \dots & 0 \\ 0 & \dots & & 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 0 & \dots & & & 0 & 1 & 0 & 0 \\ 0 & \dots & & & 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & \dots & & & & 0 & 1 & 0 \\ 0 & \dots & & & & 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & \dots & & & & & & 0 \end{pmatrix}. \quad (5.1.1)$$

Matice (??) zajistí přechod mezi diskretizací h a H , jinými slovy je to zobrazení z $\mathbb{R}^{\frac{n}{2}-1}$ do \mathbb{R}^{n-1} . Pakliže mám spočtené reziduum na obou sítích, mohu přistoupit k násobení matice A_H

$$A_h = P_{Hh}^T A_H P_{Hh}. \quad (5.1.2)$$

Řešení na hrubší síti je pak vypočteno pomocí GEM. Jelikož tato soustava má zhruba o polovinu méně neznámých, není výpočet tak náročný, jako v případě diskretizace h . Nyní



Obrázek 6: Diskretizace pro metodu Two Grid

odvodím iterační matici metody Two Grid. Již vím, že iterační matice Richardsonovy metody vypadá následovně

$$u^{k+1} = \underbrace{(I - \alpha A)}_M u^k + \underbrace{\alpha b}_N.$$

Iterační matici Two Grid metody pak odvodím takto

$$u_h = (I - \alpha A_h) u_h + \alpha b_h \quad (5.1.3)$$

$$r_h = b_h - A_h u_h. \quad (5.1.4)$$

Na základě (??) a (??) mohu psát

$$r_H = P_{Hh}^T r_h = P_{Hh}^T (b_h - A_h u_h) = P_{Hh}^T b_h - P_{Hh}^T A_h u_h \quad (5.1.5)$$

$$A_H = P_{Hh}^T A_h P_{Hh}. \quad (5.1.6)$$

Dále pak

$$A_H u_H = r_H$$

$$u_H = A_H^{-1} r_H$$

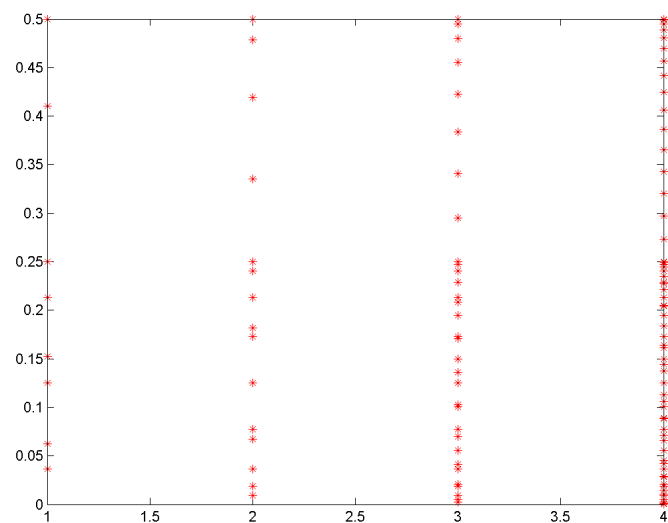
A tedy mohu odvodit iterační předpis

$$\begin{aligned}
u^1 &= u^0 + P_{Hh}u_H \\
u^1 &= u^0 + P_{Hh}A_H^{-1}r_H \\
u^1 &= u^0 + P_{Hh}A_H^{-1}\left(P'_{Hh}b_h - P'_{Hh}A_h u^0\right) \\
u^1 &= u^0 + P_{Hh}A_H^{-1}P'_{Hh}b_h - P_{Hh}A_H^{-1}P'_{Hh}A_h u^0 \\
u^1 &= u^0 - P_{Hh}A_H^{-1}P'_{Hh}A_h u^0 + P_{Hh}A_H^{-1}P'_{Hh}b_h \\
u^1 &= \left(I - P_{Hh}A_H^{-1}P'_{Hh}A_h\right)u^0 + P_{Hh}A_H^{-1}P'_{Hh}b_h \\
u^1 &= \underbrace{\left(I - P_{Hh}A_H^{-1}P'_{Hh}A_h\right)}_M u^0 + \underbrace{P_{Hh}A_H^{-1}P'_{Hh}b_h}_N \\
u^{k+1} &= \underbrace{\left(I - P_{Hh}A_H^{-1}P'_{Hh}A_h\right)}_M u^k + \underbrace{P_{Hh}A_H^{-1}P'_{Hh}b_h}_N
\end{aligned} \tag{5.1.7}$$

Po korekci na hrubší síti je provedeno ještě m iterací Richardsonovy metody

$$u^{k+1} = \underbrace{\left(I - \alpha A_h\right)^m \left(I - P_{Hh}A_H^{-1}P'_{Hh}A_h\right)}_M u^k + \underbrace{P_{Hh}A_H^{-1}P'_{Hh}b_h}_N \tag{5.1.8}$$

Na Obrázku ?? je vidět, jak iterační matice metody Two Grid ještě zmenšila maximální vlastní čísla. Hodnot je zde uvedeno méně kvůli kroku na hrubé síti pomocí GEM, tedy $h = 0.125, 0.0625, 0.03125, 0.015625$. Výpis (a zároveň algoritmus Two Grid) metody vytvořené v Matlabu vidíme níže.



Obrázek 7: Spektra iterační matice metody Two Grid

```
function [uh, iter]= twoGrid (Ah,bh,AH,PHh,u0,epsilon,alfa,m)
```

```
[uh, it]= Richardson(Ah,bh,epsilon,alfa,u0,m);
```

```
rh= bh- Ah*uh;
```

```
rH= PHh'*rh;
```

```
iter = it ;
```

```
AH=PHh'*Ah*PHh;
```

```
uH= AH \ rH;
```

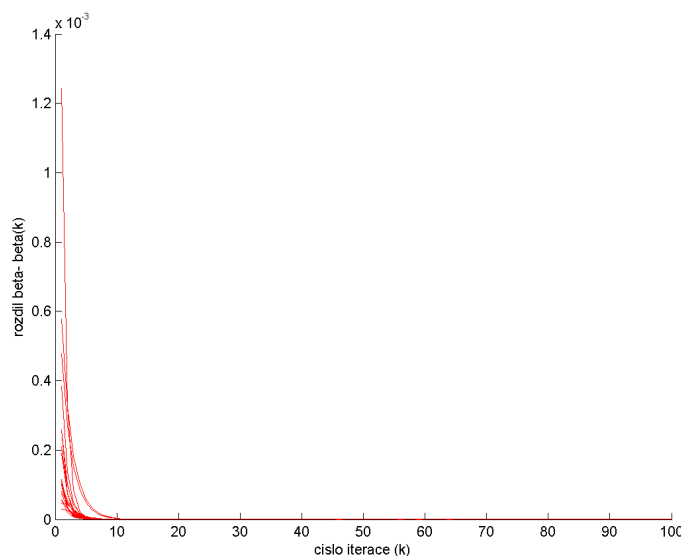
```
uh= uh+ PHh* uH;
```

```
[uh, it]= Richardson(Ah,bh,epsilon,alfa,uh,m);
```

```
iter = iter + it ;
```

Výpis 2: Two Grid metoda

Rychlost konvergence ukazují na Obrázku ??, kde je zvolen krok $h = 0.1$ a pro počet iterací $k = 100$ konvergují koeficienty v jednotlivých iteracích k nule. Je vidět jasné zlepšení oproti Richardsonově metodě, která nedovedla omezit dostatečnou mírou nejvyšší koeficient, což způsobovalo pomalou konvergenci.

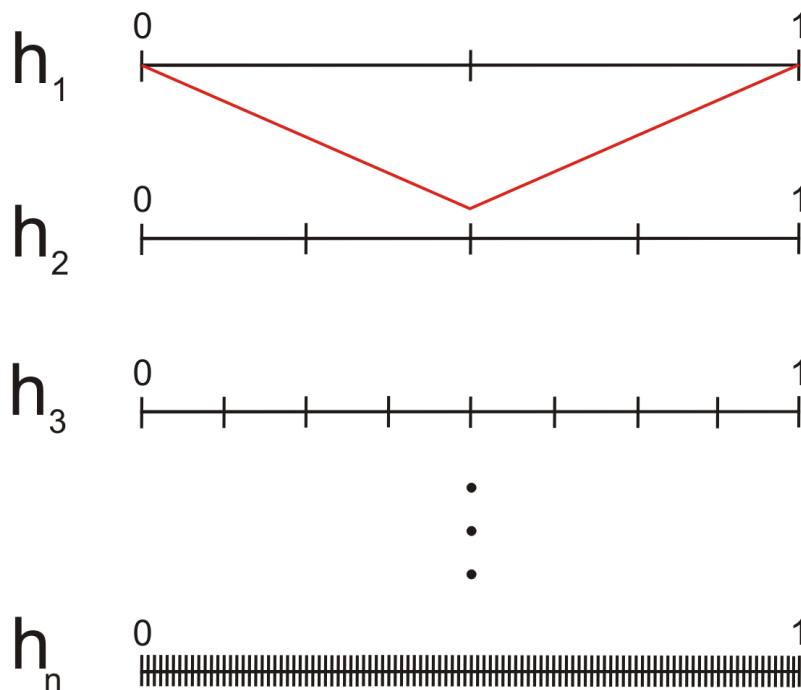


Obrázek 8: Konvergence koeficientů metody Two Grid

5.2 Multigrid metoda

Multigrid je silný nástroj pro řešení soustav rovnic. Jeho rekurzivní implementaci uvádím ve výpisu ???. Odstraňuje nevýhodu Two Grid, což je časově i paměťově náročná GEM ve druhé úrovni. Používá kombinaci kroků výše uvedených iteračních metod, a to tímto způsobem:

- inicializuje se s parametry level (počet zanoření v-cycle), pravou stranou a nultou iterací (obvykle nulový vektor odpovídající délky),
- sestaví si hierarchii potřebných matic, vektorů a parametrů, se kterými bude pracovat (dále jen globální pole),
- v každé iteraci si z globálního pole vybere matici, vektor a parametr k odpovídající úrovni zanoření v-cycle,
- provede vždy pevně daný počet m iterací Richardsonovy metody,
- spočítá reziduum, následně jej prolonguje a vloží jej jako nultou iteraci v rekurzivním volání,
- v první úrovni, na nejhrubší síti, je použito GEM, kdy časová náročnost je minimální a krok h je poměrně velký, kupříkladu $h_1 = 0.5$,
- při vracení se do vyšších úrovní se do konečné proměnné přičítají jednotlivé příspěvky přenásobené transponovanou prolongační maticí, čímž pak je dosaženo stejných dimenzí vektorů,



Obrázek 9: Diskretizace pro metodu Multigrid

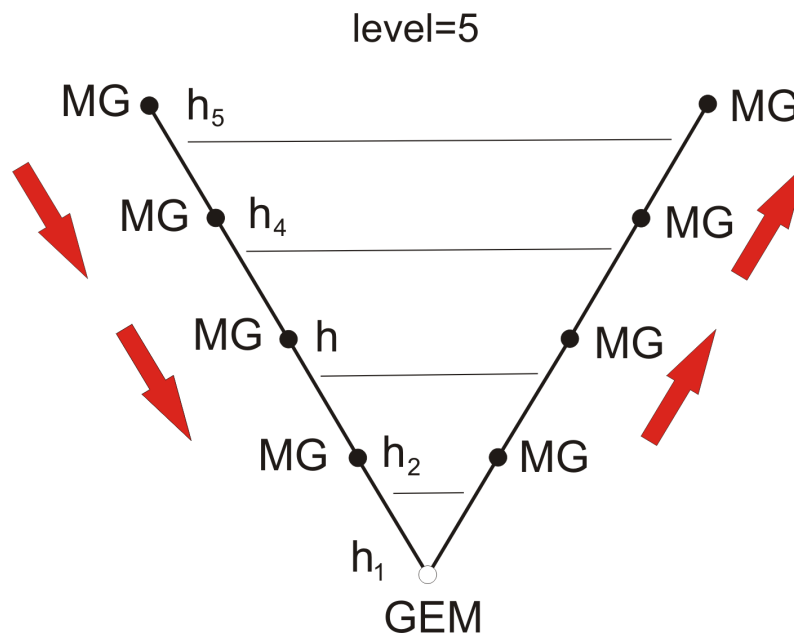
- na konci je možno ještě provést tzv. postsmoothing, což je ještě dalších m iterací Richardsonovy metody s výstupem z v -cycle v roli počátečního vektoru,
- po postsmoothingu je výstupem řešení úlohy (??).

Diskretizace známého intervalu $\langle 0, 1 \rangle$, kde řeším úlohu je na obrázku ?? a ukazuje tuto hierarchii sítí

$$h_i = \frac{1}{2} h_{i+1}, i \in (1, \dots, n-1).$$

Obrázek ?? ukazuje průběh V-cyklu s informací o směru rekurze a diskretizaci dané části zanoření. V algoritmu multigridu znamená p tvar zanoření, pokud je $p = 1$, jedná se o V-cyklus, pokud by bylo $p = 2$, pak bych hovořil o W-cyklu. Existují však i další tvary zanoření, o nichž se zde nebudu zmiňovat. Plné kolečka jsou rekurzivní volání, prázdné je řešení GEM na nejhrubší úrovni diskretizace.

Rychlost konvergence ukazují na Obrázku ??, kde je zvolen $level = 9$ (nejjemnější diskretizace v tomto případě je $h = 0.00195$) a v každém kroku bylo zadáno $m = 4$ iterace a pro počet iterací $k = 100$ konvergují koeficienty v jednotlivých iteracích k nule. Je vidět



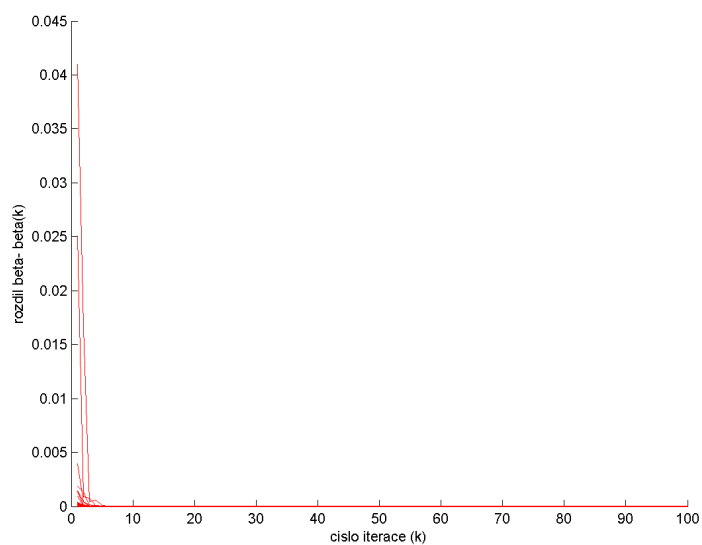
Obrázek 10: V-cyklus metody Multigrid

Algorithm 1 Algoritmus Multigridu

```

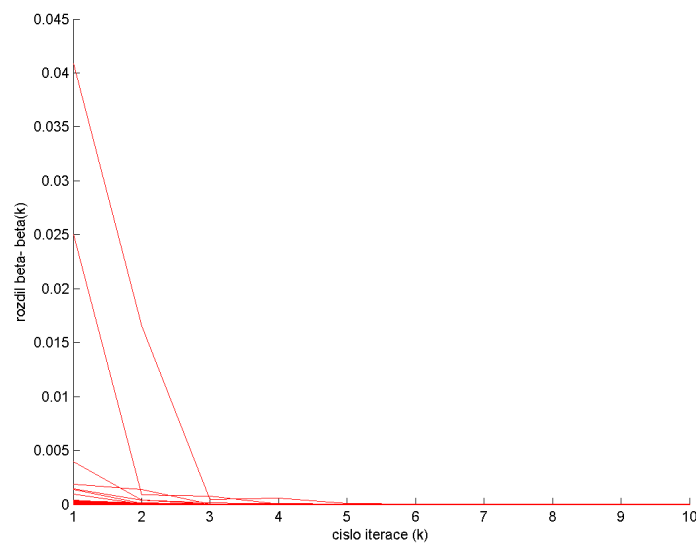
Multigrid ( $k, u^0, b$ )
if  $k=1$  then
     $u^0 = A_1^{-1}b$ 
end if
 $\Lambda_k \leq Ch_k^{-2}$ 
for  $l = 1$  to  $m_1$  do
     $u^l = u^{l-1} + \frac{1}{\Lambda_k} (b - A_k u^{l-1})$ 
end for
 $b = I_k^{k+1} (b - A_k u^{m_1})$ 
 $g_0 = 0$ 
for  $i = 1$  to  $p$  do
     $g_i = \text{Multigrid}(k-1, g_{i-1}, b)$ 
end for
 $u^{m_1+1} = u^{m_1} + I_{k-1}^k g_p$ 
for  $l = m_1 + 2$  to  $m_1 + m_2 + 1$  do
     $u^k = u^{l-1} + \frac{1}{\Lambda_k} (b - A_k u^{l-1})$ 
end for
Multigrid ( $k, u^0, b$ ) =  $u^{m_1+m_2+1}$ 

```



Obrázek 11: Konvergence koeficientů metody Multigrid ($k=100$)

enormní zlepšení oproti Two Grid metodě, která konvergovala pomaleji. Pro lepší přesnost uvádím ještě na Obrázku ?? stejnou situaci s počtem iterací $k = 10$.



Obrázek 12: Konvergence koeficientů metody Multigrid (k= 10)

5.3 Efektivita Multigridu- odvození složitosti

Nechť A_k je matice příslušné úrovně k , zde volím $p = 1$.

Věta 5.1 *Multigrid je třídy složitosti $O(n)$.*

Důkaz.

$$\text{Dim}A_k = n_k$$

Označím W_k jako práci potřebnou k vykonání k . úrovně obecně

$$W_k \leq C\bar{m}n_k + pW_{k-1}, \bar{m} = m_1 + m_2,$$

kde m_1 je práce při zanořování do v-cyklu a m_2 je složitost při vynořování z nižší úrovně v-cyklu a C je konstanta. $C' (n_1)^2$ je počet operací nutných k vyřešení GEM na nejhrubší diskretizaci

Pro počáteční iterace W_k dostanu

$$W_2 \leq C\bar{m}n_2 + C' (n_1)^2$$

$$W_3 \leq C\bar{m}n_3 + C\bar{m}n_2 + C' (n_1)^2$$

Jelikož vztah mezi dimenzemi soustav je

$$n_k = 2^{k-1}n_1 - 1$$

Pak mohu napsat

$$W_k \leq \sum_{i=2}^{k-1} C\bar{m} (2^{i-1}n_1 - i + 1) + C' (n_1)^2$$

Po rozepsání

$$W_k \leq 2C\bar{m} (n_1 2^{k+1} - 1) + C(k-3)\bar{m} - \sum_{i=2}^{k-1} C\bar{m}i + C' (n_1)^2$$

$$W_k \leq 2C\bar{m} (n_1 2^{k+1} - 1) + C(k-3)\bar{m} - C\bar{m} \frac{(k-3)[4 + (k-4)]}{2} = f(n_k)$$

$$f(n_k) = f(2^{k-1}n_1 - k)$$

Což je asymptotická složitost.

$$f(n_k) = O(n_k)$$

■

6 Porovnání metod přímých, iteračních a multigridu

6.1 Aplikace algoritmů na počítači

Zde chci ukázat grafy konvergence jednotlivých metod, jejich částečná řešení a experimenty. Jako pravou stranu pro všechny případy pokusů, tedy sílu, kterou jsem nechal zatěžovat strunu, jsem zvolil funkci

$$f(x) = 1 - |\sin(20x_i)| + |\cos(20x_i)|, i \in (2, \dots, n-1), n = \frac{1}{h} + 1. \quad (6.1.1)$$

Pro $h = 0.01$ je síla vyobrazena na Obrázku ?? a řešení pro tuto pravou stranu s diskretizací $h = 0.01$ je na Obrázku ??.

6.1.1 Richardsonova metoda- pozorování

Na Obrázku ?? je znázorněna Richardsonova metoda na síti $h = 0.01$ a je dobře vidět, jak rychle dokáže tlumit silnou amplitudu: červená barva v 1., 16. a 31. iteraci, ale celková chyba velmi pomalu konverguje a blíží se nule, to jsou modré křivky v iteracích 1550, 3100 a 4650. Zajímalo mě také, jak se bude chovat Richardsonova metoda, když nastavím jiný tlumící koeficient, než je $\alpha \in (0, \frac{1}{\lambda_{max}})$, $i \in (1, \dots, n)$. Zkusil jsem tedy volbu $\alpha = \frac{4}{\lambda_{max}}$. Nastavení byla totožná jako v předchozím případě, tedy $h = 0.01$ a souhlasí i barevné označení. Na Obrázku ?? je dobře vidět, že chyba se hodně zvětšila již při prvních iteracích a divergovala.

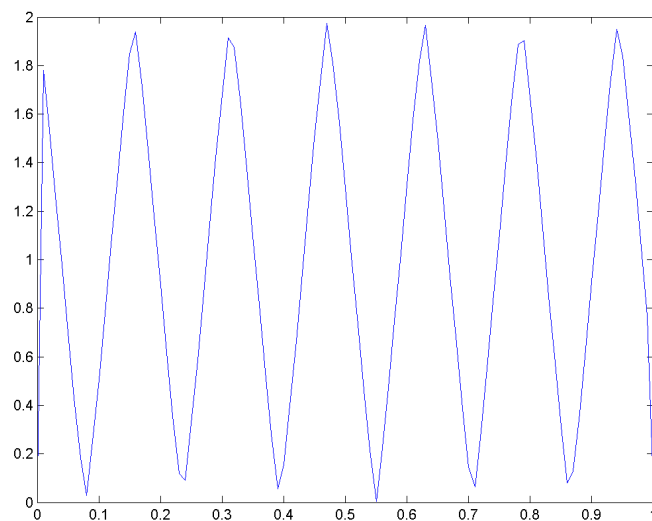
6.1.2 Two Grid metoda- pozorování

Nyní popíšu metodu Two grid. Na Obrázku ?? je vidět jednotlivé příspěvky- červenou barvou jsem označil část vypočtenou na hrubší síti a modrou barvou část řešenou na jemnější síti. V tomto případě bylo $h = 0.02$ a $H = 0.04$, počet iterací byl volen napevno- 3 iterací. Tímto obrázkem jsem chtěl ukázat, jak se od sebe jednotlivé příspěvky liší. Před GEM byl proveden stejný počet iterací Richardsonovy metody jako po výpočtu GEM- tento proces se nazývá presmoothing (předvyhlazení) a postsmoothing (povyhlazení). Výsledek, který reprezentuje červená křivka byl vstupem do Richardsonovy metody jejíž výsledek je modrá křivka.

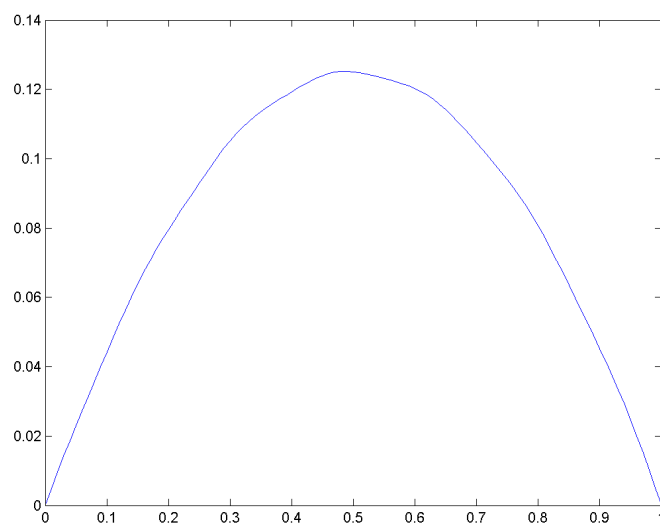
Na dalším Obrázku ?? je vidět tu samou situaci, avšak po 9 iteracích na jemné síti.

6.1.3 Multigrid metoda- pozorování

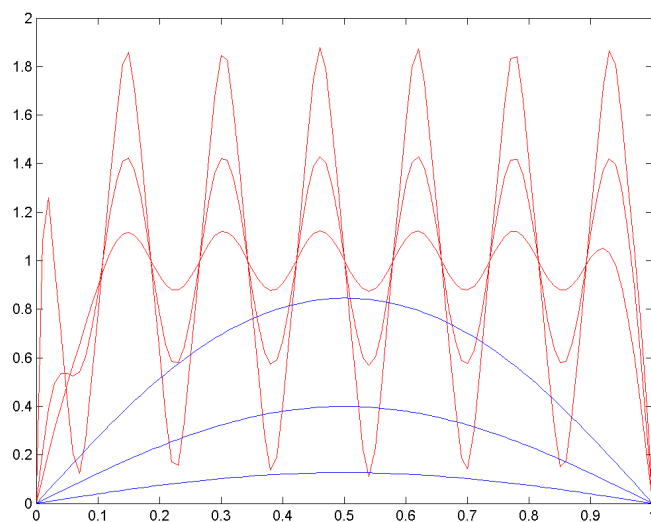
Obrázek ?? ukazuje, jak rychle se metoda Multigrid blíží k řešení, už rozdíl mezi 4. a 5. úrovní je neznatelný. V každé úrovni V-cyklu se provedly 4 iterace Richardsonovy metody, pak byla přesnost řádu $1e-5$. Opět jsem chtěl zkusit, jak bude vypadat výsledek Multigridu pro jinou volbu α , zkusil jsem, jako v předchozím případě $\alpha = \frac{4}{\lambda_{max}}$. Výsledek vidíte na Obrázku ?? . Je vidět, jakou má váhu zanoření dané úrovně, zpočátku není chyba tak enormní, později však vlivem více iterací dochází k vysoké chybě.



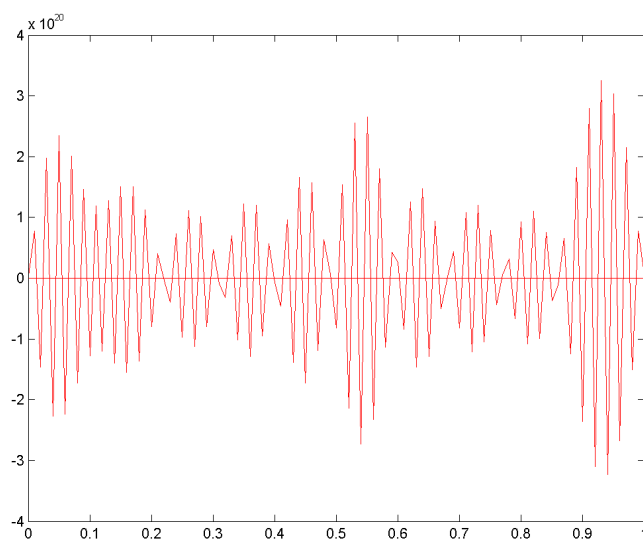
Obrázek 13: Pravá strana- zátěžová funkce



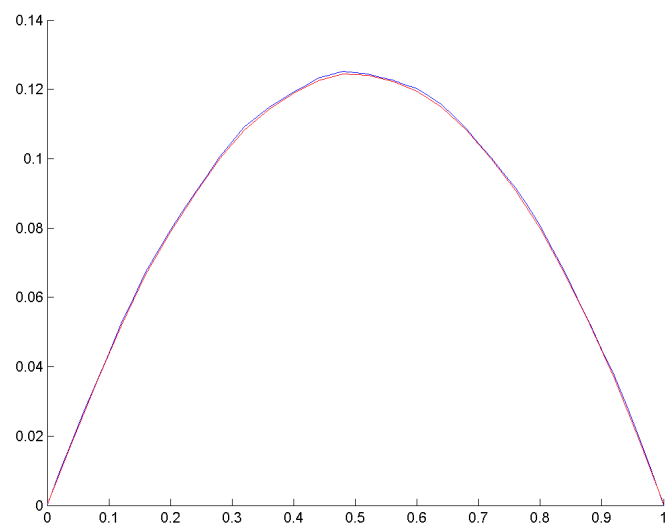
Obrázek 14: Řešení průhybu struny



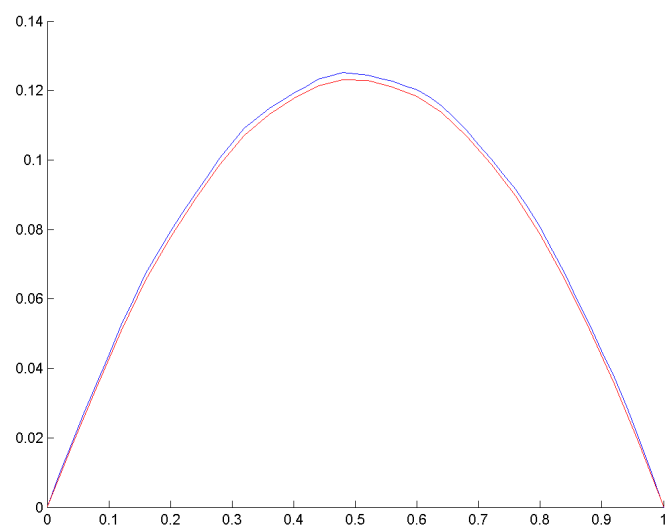
Obrázek 15: Konvergence chyby řešení Richardsonovy metody



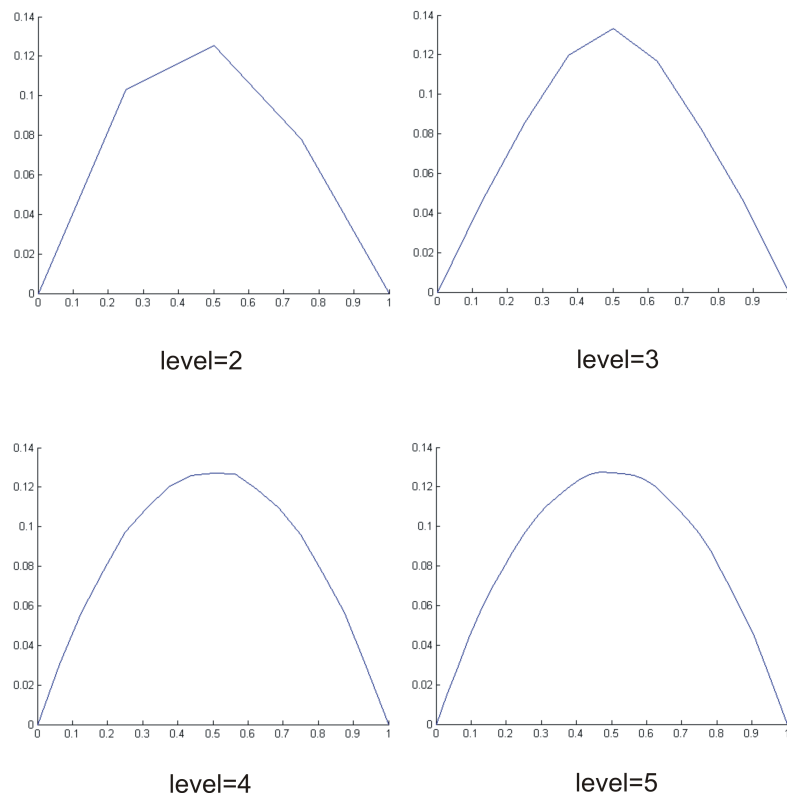
Obrázek 16: Divergence chyby řešení Richardsonovy metody- jiná volba α



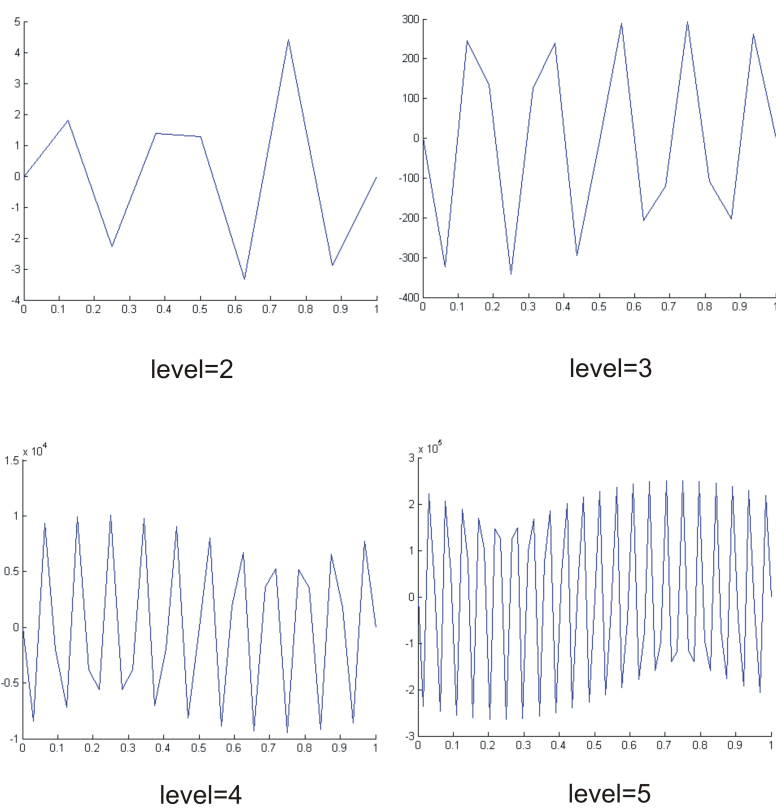
Obrázek 17: Příspěvky jednotlivých diskretizací Two Grid ($m=3$)



Obrázek 18: Příspěvky jednotlivých diskretizací Two Grid ($m=9$)



Obrázek 19: Výsledek Multigridu pro různé zanoření V-cyklu



Obrázek 20: Divergence Multigridu vlivem tlumícího koeficientu

6.2 Efektivita jednotlivých metod

Počty iterací pro konvergenci k výsledku v řádu $\epsilon = 1e - 5$ jsou uvedeny v Tabulce ?? . Uvedl jsem Richardsonovu metodu v konfrontaci s Two Grid a Multigrid metodou, ale jelikož Two grid metoda používá na hrubší síti GEM, výsledky jsou tím značně ovlivněny. Two Grid metoda je zde pouze pro přehled, protože v určité úrovni se chová velmi podobně jako GEM o úroveň výše. Vzhledem k tomu, že v souvislosti s přímou metodou nelze hovořit o počtu iterací, samotnou GEM zde neuvádím. Časové nároky na výpočet

h	rychlost řešení [it]		
	RM	TG	MG
0.25	61	1	8
0.125	282	4	8
0.0625	1158	4	8
0.03125	4664	4	8
0.0156	18667	4	8
0.0078	74679	4	8
0.0039	298666	3	8
0.00195	1194548	3	8

Tabulka 1: Počty iterací potřebné k výpočtu jednotlivých metod na různých sítích

jsou uvedeny v Tabulce ?? . Všimněme si hlavně časové náročnosti Richardsonovy metody na jemnější diskretizaci, na nejjemnější síti je poznat velmi vysoký nárůst potřebného času, až půl hodiny. Z obou tabulek mohu usoudit, že Richardsonova metoda stejně jako GEM není příliš vhodný nástroj pro řešení větších soustav (GEM implementovaný v Matlabu je zřejmě optimalizován, tedy rozdíly mezi jednotlivými diskretizacemi nejsou tak markantní). Ačkoliv podle počtu iterací vypadá Two Grid výhodně oproti Multigridu, v časové náročnosti je na tom o poznání hůře. Tyto tabulky tedy dokazují, jak je Multigrid vhodná metoda pro řešení rozsáhlých soustav.

h	rychlost řešení [s]			
	GEM	RM	TG	MG
0.25	0	0	0	0
0.125	0	0	0	0
0.0625	0	0	0	0
0.03125	0	0.047	0	0
0.0156	0	0.282	0	0
0.0078	0	3.063	0.015	0
0.0039	0.016	42.8	0.078	0.0039
0.00195	0.063	1851.2	0.56	0.032

Tabulka 2: Časy výpočtu jednotlivých metod na různých sítích

7 Závěr

Na základě teoretického rozboru, modelaci úlohy v Matlabu a mnoha experimentech bylo ověřeno, jak je Multigrid silný nástroj. Praktickými testy byla dokázána rychlost konvergence, praktická nezávislost algoritmu Multigrid na volené diskretizační síti, i rychlá konvergence v čase. Bylo dokázáno, že algoritmus Multigridu má lineární složitost. V práci byla řešena pouze úloha v 1D prostoru. Vzhledem k tomu, že 2D, resp. 3D, prostor obsahuje n^2 , resp. n^3 , prvků, je možno mezi metodami očekávat ještě markantnější rozdíl. V oblasti Multigridu je ještě mnoho metod a postupů, jak ještě efektivněji řešit úlohy tohoto typu, v dalším studiu bych se tedy rád ubíral hlouběji do této problematiky.

8 Reference

- [1] U. Trottenberg, C. W. Oosterlee a A. Schuller, *Multigrid*, Academic Press, 2001.
- [2] J. W. Ruge a K. Stüben, *Algebraic multigrid*, sborník Multigrid Methods 4, str. 73-130, SIAM, 1987.
- [3] Z.Dostál, *Lineární Algebra*, VŠB- Technická Univerzita Ostrava, 2004.
- [4] V.Vondrák, *Numerické metody I*, [http : //www.am.vsb.cz/vondrak/education/nm1/lectures](http://www.am.vsb.cz/vondrak/education/nm1/lectures).
- [5] J.Bouchala, *Matematická analýza I*, [http : //am.vsb.cz/bouchala/MA_pro_IT/9.pdf](http://am.vsb.cz/bouchala/MA_pro_IT/9.pdf).