20'

<u>smoothing property</u> (<u>vyhlazování</u> chyby)

jestliže $u^{(i+1)} = R u^{(i)} + g$, pro zjednodušení značení

pak pro chyby

$$e^{(i+1)} = R e^{(i)}$$

(což jsme viděli u iteračního zpřesnění s $R = I - M^{-1}A$)

$\Rightarrow \quad e^{(i)} = R^i e^{(0)}$ pokles chyby určený mocninou R

• a víme ZNM ~~...~~ $\lim_{i\to\infty} R^i = 0 \iff \rho(R) < 1$
spektrální rádius

neboli $|\lambda| < 1 \quad \forall \lambda \in sp(R)$

• nechť (λ, v) je vlastní pár R, pak

$$R^i v = R^{i-1} R v = \lambda R^{i-1} v = \dots = \lambda^i v$$

a $\|R^i v\| = |\lambda|^i \|v\|$

$\Rightarrow \quad \frac{\|R^i v\|}{\|v\|}$ konverguje k 0 rychleji pro $|\lambda|$ blízké nule
pomaleji pro $|\lambda|$ velká

• když $e^{(0)}$ rozložíme do jednotlivých vl. podprostorů matice R, tak některé složky chyby budou s $i \to \infty$ klesat rychle a některé pomalu

• ukazuje se, že malým vl. číslům matice R příslušejí oscilující vektory a ~~ale~~ velkým vl. číslům hladké vektory

$\Rightarrow \quad e^{(i)} = R^i e^{(0)}$ je pro velká i : <u>hladký vektor</u>

- 2 důsledky smoothing property (resp. vztahu $e^{(i)} = R^i e^{(0)}$)
  1) chyba bude po několika iteracích hladká (dominovaná vl. vektory přísl. ~~malým~~ velkým vl. číslům R)
  2) pro hladký počáteční vektor mohou stac. it. metody konvergovat rychle – takový vektor musí být dominován vl. vektory přísl. malým vl. číslům R

hladký/oscilující vektor – intuitivní termín, ale chceme to popsat formálněji (pro odvození MG, AMG, ...)

↳ založíme to na vlastnostech problému (nikoli metody řešení)

A_j je SPD $\Rightarrow$ generuje skalární součin $(v, A_j w) = v^T A_j w$

- řekneme, že hladké vektory mají malou energii, tedy $\|v\|_{A_j}^2 = v^T A_j v$
  – " – oscilující – velkou – " –

  (musí to kvantifikovat přesně, ale my to přesně nepotřebujeme)

- pro (λ_k, v_k) vlastní pár A_j platí

$$\|v_k\|_{A_j}^2 = v_k^T A_j v_k = v_k^T (\lambda_k v_k) = \lambda_k \|v_k\|^2$$

$\Rightarrow$ hladké jsou vl. vektory přísl. malým vl. číslům
oscilující velkým

- A_j je diagonalizovatelná $\Rightarrow$ vl. vektory tvoří bázi $\mathbb{R}^{N_j}$
  ↳ budeme se dívat, jak se smoothery a MG obecně chovají pro vl. vektory A_j

Vlastní vektory (weighted) Jacobiho

$$R_\omega = (1-\omega)I + \omega R_J = I + \omega\left(D^{-1}(L+U) - I\right) =$$
$$= I - \omega \cdot D^{-1}A$$

v našem 1D případě $D = 2 \cdot I \Rightarrow D^{-1} = \frac{1}{2} I$

$\Rightarrow R_\omega = I - \frac{\omega}{2} A$

vl. vektor A je vl. vektorem R_ω a pro (λ_k, v_k)

$$R_\omega v_k = \left(1 - \frac{\omega}{2}\lambda_k\right) v_k$$

vl. čísla a vl. vektory A (z 1D kon. diferencí)

lze ukázat, že $(n = \dim(A))$

$$\lambda_k(A) = 4 \sin^2\left(\frac{k\pi}{2n}\right) \qquad 1 \le k \le n-1$$

$$v_{k,j} = \sin\left(\frac{jk\pi}{n}\right) \qquad 1 \le k \le n-1, \quad 1 \le j \le n-1$$

j-tá složka vl. vektoru

$$\Rightarrow \lambda_k(R_\omega) = 1 - 2\omega \sin^2\left(\frac{k\pi}{2n}\right) \qquad 1 \le k \le n-1$$

(pozn. pro jemnější dělení roste $n \Rightarrow \lambda_1 \to 1$)

obrázky z MG Tutor.

Fig 2.6 (str. 20) – jak vypadají vl. vektory A

Fig 2.7 – vlastní čísla R_ω (pro různá ω)

↳ klasický Jacobi efektivně potlačuje vl. vektory "uprostřed spektra"

↳ $\omega = 2/3$ znamená, že potlačujeme i ty rychlejší

Fig 2.8 – potvrzení předchozího pozorování

kolik iterací potřebujeme pro potlačení chyby, pokud $e^{(0)} = v_k$

Fig 2.4, 2.5 (str. 15) – konvergence metody pro obecn. volbu chyby

Pro Gauss-Seidela je to trochu jiné, vl. vektory A nejsou vl. vektory R_{GS}. Smoothing property tak jen ilustrujeme numericky → Fig 2.11 (str. 26) zpět ke konvekci na hrubé síti

jak se numericky chová pro oscilující / hladké módy ?

MATLAB : smoothing_property(_1D) - grafy srovnávající, jak vypadá norma vl. matice A pro konvekci na hr. síti (a zhlazení!)

(zde pro 2-úrovňovou metodu)

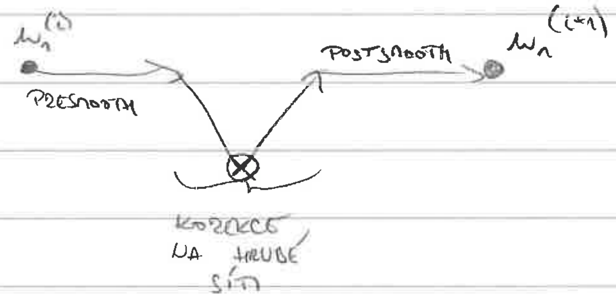
MULTIGRID = KOMBINACE ZHLAZOVÁNÍ A KOREKCE NA HRUBÉ SÍTI

vyjdeme z důsledků smoothing property, které jsme si popsali dříve

VAR I: počítáme korekci na hrubé síti, dostaneme oscilující vektor a ten vyhladíme

VAR II: počítáme vyhlazování a hladkou chybu opravíme na hrubé síti

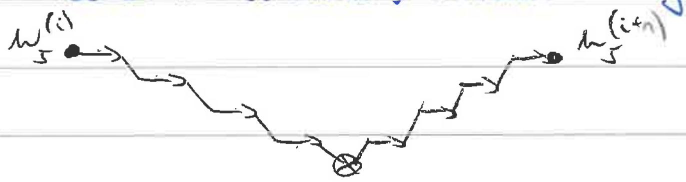
VAR III: kombinace I+II
→ schéma



zobecnění na více sítí

myšlenka: problém na hrubé síti (v rámci korekce na hr. síti) neřešíme přímou metodou, ale rekurzí, dokud není dostatečně malý

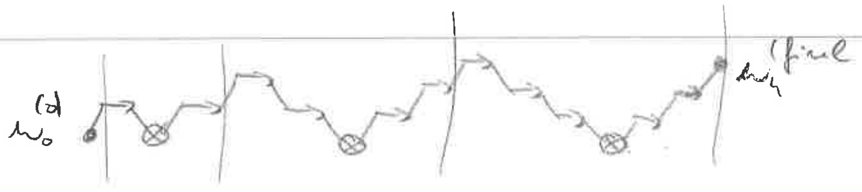
→ základní (iterační) schéma: "V-cyklus"



• rekurze může vypadat složitěji "W-cyklus"

lze zobecnit

• FMG – neiterační MG



PROJEKT pro studenty

k dispozici kódy pro (1D a) 2D multigrid

(v 2D mi nefungují variational property – proč?!)

co pro ně chci:

- zkusit vybrat si nějakou problematiku / téma
- experimentovat, otestovat připravit a sepsat výsledky → PROTOCOL PREZENTACE
- představit to pak na přednášce (mini: s plakátem)

  co dostali? proč se to děje? co s tím?

možná témata:

- závislost na počtu sítí
- volba zhlazovače a počtu zhlazení (včetně volby #presmooth #postsmooth)
- nepřesné řešení na hrubé síti
- různý interpolační stencil / různá volba hrubých sítí
- délka výpočtu v závislosti na ...
- ...

sledování ~~kon~~ konvergence

- maximální / asymptotický kontrakční faktor
- # iterací než poklesne residuum / chyba pod danou toleranci