Gradientní iterační metody

- základní myšlenka: pro symetrické a pozitivně definitní matice

  je nalezení řešení $Ax=b$ ekvivalentní

  nalezení minima kvadratické formy

$$\boxed{\phi(x) = \frac{1}{2} x^T A x - x^T b}, \quad x \in \mathbb{R}^n$$

neboť

$$\nabla\phi(x) = \frac{1}{2}\left[Ax + (x^TA)^T\right] - b \underset{\uparrow}{=} Ax - b = 0$$

pro $A = A^T$ (symetrické matice)

- hledáme přibližné řešení iteračně

  volbou poč. x_0 (např. $x_0 = 0$) a pak vhodného směru p_k v každém

  kroku s tím, že minimalizujeme $\phi(x_k + \alpha p_k)$ vzhledem k α

  neboli $\frac{\partial\phi(x_k+\alpha p_k)}{\partial\alpha} = 0$

protože

$$\phi(x_k+\alpha p_k) = \frac{1}{2}(x_k+\alpha p_k)^T A (x_k+\alpha p_k) - (x_k+\alpha p_k)^T b$$

$$= \underbrace{\frac{1}{2} x_k^T A x_k - x_k^T b}_{\text{konst.}} + \frac{\alpha}{2}(\underbrace{p_k^T A x_k + x_k^T A p_k}_{2p_k^T A x_k \text{ díky symetričnosti}}) - \alpha p_k^T b + \frac{\alpha^2}{2} p_k^T A p_k$$

$$= \frac{1}{2} x_k^T A x_k - x_k^T b + \alpha\left[p_k^T (Ax_k - b)\right] + \frac{\alpha^2}{2} p_k^T A p_k$$

a derivováním

$$0 = \frac{\partial\phi}{\partial\alpha} = -p_k^T r_k + \alpha\, p_k^T A p_k \implies \boxed{\alpha_k = \frac{p_k^T r_k}{p_k^T A p_k}}$$

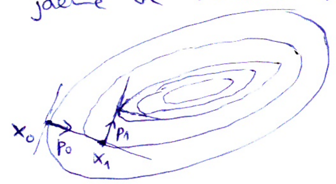
kde jsme označili zbytkový (reziduální) vektor v k-té iteraci

$$r_k = b - Ax_k \qquad \text{(reziduum)}$$

- volba směru p_k:

metoda největšího spádu

jdeme ve směru $\nabla\phi$



neboli $\nabla\phi(x_k) = Ax_k - b = -r_k$

takže volíme $p_k = r_k$

a tedy

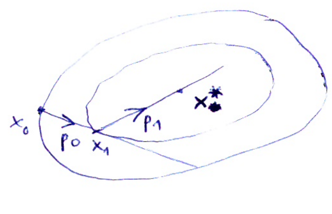
$$\alpha_k = \frac{r_k^T r_k}{r_k^T A r_k}$$

metoda sdružených gradientů (conjugate gradient - CG)

jdeme v tzv. A-sdruženém směru

tj. volíme p_k tak, aby

$$\boxed{p_k^T A p_i = 0 \quad \forall i = 0,\dots,k-1}$$



ve 2D jdeme přímo správným směrem

a) <u>metoda největšího spádu</u>

přímočaře:

zvolíme x_0

pro $k = 1, 2 \dots$

$r_{k-1} = b - A x_{k-1}$ (pokud malé konec)

$$\alpha_{k-1} = \frac{r_{k-1}^T r_{k-1}}{r_{k-1}^T A r_{k-1}}$$

$$x_k = x_{k-1} + \alpha_{k-1} r_{k-1}$$

ne moc dobré kvůli dvěma matic. násobením

lépe: $r_k = b - A x_k = b - A(x_{k-1} + \alpha_{k-1} r_{k-1})$
$= r_{k-1} - \alpha_{k-1} A r_{k-1}$

a tedy

zvolíme x_0

$r_0 = b - A x_0$

pro $k = 1, 2, \dots$

$w_{k-1} = A r_{k-1}$

$$\alpha_{k-1} = \frac{r_{k-1}^T r_{k-1}}{r_{k-1}^T w_{k-1}}$$

$x_k = x_{k-1} + \alpha_{k-1} r_{k-1}$

$r_k = r_{k-1} - \alpha_{k-1} w_{k-1}$

pokud $\|r_k\| < \varepsilon$ konec

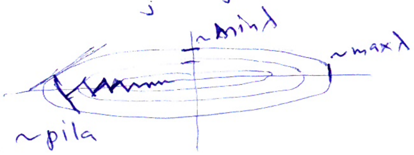
• pro hůře podmíněné matice A pomalá konvergence

$$\varkappa(A) = \frac{\max_j |\lambda_j|}{\min_j |\lambda_j|} \gg 1 \Rightarrow$$ protáhlé elipsoidy

vždy jdeme kolmo na předchozí směr

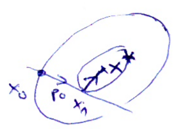
$$r_{k-1}^T r_k = r_{k-1}^T r_{k-1} - \alpha_{k-1} r_{k-1}^T A r_{k-1} = 0$$

což není dobré



b) <u>metoda sdružených gradientů</u>

motivace volby $p_1^T A p_0 = 0$ ve 2D:



protože $\nabla\phi(x_1) = -r_1$ je kolmé na p_0, dostáváme

$$0 = p_0^T r_1 = p_0^T (b - A x_1) = p_0^T (A x^* - A x_1) = p_0^T A \underbrace{(x^* - x_1)}_{\alpha p_1 \text{ pro jisté } \alpha} = \alpha\, p_0^T A p_1 = 0$$

a díky symetričnosti A máme

$p_1^T A p_0 = 0$ ← přímočará volba ve 2D

ovšem pro $m > 2$ nemáme jednoznačnou volbu, protože neznáme x^*

pokud budeme volit p_k tak, aby bylo A-sdružené s $p_0, \dots, p_{k-1}$

tak budeme minimalizovat $\phi(x)$ v prostoru $x_0 + c_0 p_0 + \dots + c_k p_k$

a optimum bude $x_k = x_0 + \alpha_0 p_0 + \dots + \alpha_k p_k$

[p_0 a p_1 jsou lin. nezávislé → tvoří rovinu, která řeže elipsoidy
→ soustředné elipsy v rovině → střed bude v x_2 atd.]

[pokud některé z elips je kružnice, pak se dostaneme hned do středu, což je pokud $\lambda_i = \lambda_j$ pro nějaké $i \neq j$, λ_i vlastní čísla A
→ rychlejší konvergence]

algoritmus metody sdružených gradientů (bez předpodmínění)

zvol x_0 (např. 0)

$r_0 = b - A x_0$

$p_0 = r_0$, $\gamma_0 = r_0^T r_0$

pro $k = 1, 2, \dots$

toto jako u metody největšího spádu:

$$\left\{\begin{aligned} w_{k-1} &= A p_{k-1} \\ \alpha_{k-1} &= \gamma_{k-1} / (p_{k-1}^T w_{k-1}) \\ x_k &= x_{k-1} + \alpha_{k-1} p_{k-1} \\ r_k &= r_{k-1} - \alpha_{k-1} w_{k-1} \end{aligned}\right.$$

pokud $\|r_k\| < \varepsilon$ konec

určení nového „optimálního" směru:

$$\left\{\begin{aligned} \gamma_k &= r_k^T r_k \\ \beta_{k-1} &= \gamma_k / \gamma_{k-1} \\ p_k &= r_k + \beta_{k-1} p_{k-1} \end{aligned}\right.$$

Pozn.:

1) nejde v pravém slova smyslu o iterační metodu, v přesné aritmetice skončíme nejvýše po n krocích

avšak a) díky zaokr. chybám není konvergence zaručena

b) často konverguje mnohem rychleji (k dostatečně přesnému řešení), zvláště pokud matici předpodmíníme

2) opět pouze jedno násobení $A \cdot v$ (obvykle jako subroutina od uživatele) a 2 skalární součiny, neboť místo $p_{k-1}^T r_{k-1}$ je použito $r_{k-1}^T r_{k-1}$, protože

$$p_{k-2}^T r_{k-1} = 0$$

~techu $\perp$ ~PØ

nebo přímo výpočtem

$$p_{k-2}^T r_{k-1} = p_{k-2}^T (r_{k-2} - \alpha_{k-2} A p_{k-2}) = 0 \text{ díky volbě } \alpha_k = \frac{p_k^T r_k}{p_k^T A p_k}$$

- Lze ukázat (viz např. Trefethen, Bau: Numerical Linear Algebra) že vektory generované metodou sdruž. grad. splňují (pokud $r_k \neq 0$, pro $r_k = 0$ již máme řešení)

1) $p_k^T A p_j = 0$ pro $\forall j = 0, \dots k-1$

2) $r_k^T r_j = 0$ pro $\forall j = 0, \dots, k-1$

3) podprostory $\mathcal{L}(p_0, \dots p_{k-1})$, $\mathcal{L}(r_0, A r_0, \dots A^{k-1} r_0)$ a $\mathcal{L}(A e_0, \dots, A^k e_0)$ jsou stejné ($e_0 = x^* - x_0$) a tento podprostor se nazývá k-tý Krylovův prostor (A, r_0) (pokud volíme $x_0 = 0$, pak $r_0 = b$ a jde o (A, b))

Náznak důkazu: indukcí v k:

3) z $p_k = r_k + \beta_{k-1} p_{k-1}$ plyne, že $\mathcal{L}(p_0, \dots, p_{k-1}) = \mathcal{L}(r_0, r_1, \dots, r_{k-1})$

a z $r_k = r_{k-1} - \alpha_{k-1} A p_{k-1}$, přičemž $p_0 = r_0$ plyne $= \mathcal{L}(r_0, A r_0, \dots A^{k-1} r_0)$

a díky $e_0 = x^* - x_0$ máme $A e_0 = A x^* - A x_0 = b - A x_0 = r_0$

2) spočteme $r_k^T r_j = (r_{k-1} - \alpha_{k-1} A p_{k-1})^T r_j = r_{k-1}^T r_j - \alpha_{k-1} p_{k-1}^T A r_j = \begin{cases} 0 & \text{pro } j < k-1 \text{ dle indukce} \\ 0 & \text{pro } j = k-1 \text{ díky } \alpha_{k-1} = \dfrac{r_{k-1}^T r_{k-1}}{p_{k-1}^T A r_{k-1}} \end{cases}$

což plyne z $p_{k-1}^T A p_{k-1} = p_{k-1}^T A r_{k-1} + \underbrace{\beta_{k-1} p_{k-1}^T A p_{k-2}}_{0 \text{ dle indukce}}$

1) $p_k^T A p_j = r_k^T A p_j + \beta_{k-1} p_{k-1}^T A p_j = \begin{cases} 0 & \text{pro } j < k-1 \text{ dle indukce a díky 2), neboť } A p_j = \frac{1}{\alpha_j}(r_j - r_{j+1}) \\ 0 & \text{pro } j = k-1 \text{ neboť } \beta_{k-1} = \dfrac{r_k^T r_k}{r_{k-1}^T r_{k-1}} = -\dfrac{r_k^T A p_{k-1}}{p_{k-1}^T A p_{k-1}} \end{cases}$

$\Leftarrow r_k^T r_k = \cancel{r_k^T r_{k-1}} - \alpha_{k-1} r_k^T A p_{k-1}$, kde $r_k^T r_{k-1} = 0$

$p_{k-1}^T A p_{k-1} = p_{k-1}^T A r_{k-1} = \frac{1}{\alpha_{k-1}} (r_{k-1}^T r_{k-1} - \cancel{r_k^T r_{k-1}})$, kde $r_k^T r_{k-1} = 0$

Předpodmínění (preconditioning)

- zvláště u metod jako je metoda sdruž. gradientů apod. je předpodm. zcela zásadní pro rychlou konvergenci, neboť rychlost konvergence závisí na čísle podmíněnosti matice $\varkappa(A) = \|A\| \|A^{-1}\| = \frac{|\lambda_{max}|}{|\lambda_{min}|}$ ← pro normální matice $AA^T = A^TA$, což platí pro SPD matice
  a předpodmínění číslo podmíněnosti zmenšuje.

- zákl. myšlenka: místo $Ax = b$ řešme $M^{-1}Ax = M^{-1}b$ (což je ekvivalentní problém), kde M^{-1} je „snadné“
  ovšem i když M a A jsou SPD matice, tak $M^{-1}A$ už nemusí být SPD
  navíc konstruovat $M^{-1}A$ přímo může narušit řídkost matice $\Rightarrow$ A a M^{-1} se aplikují odděleně

- existují obecné postupy, např. $M = \mathrm{diag}(A)$, pokud se změní číslo podmíněnosti (nefunguje např. pro $T_N u = h^2 f$, neboť $\mathrm{diag}\, T_N = 2I$, což nezmění číslo podmín., jak λ_{min}, tak λ_{max} se vydělí 2)
  ale velmi často a účinněji se používají matice M šité na míru daného fyz. problému (např. vyřeší se problém bez interakce a pak se s interakcí iteruje apod.)

- pokud $M^{-1}A$ není SPD (ale A je SPD), lze řešit $\underbrace{((C^{-1})^T A C^{-1})}(Cx) = (C^{-1})^T b$ pro nesingulární matici C
  $$\tilde{A} \tilde{x} = \tilde{b}$$
  protože nyní je $\tilde{A}$ také symetrická (pro lib. C) a pozitivně definitní
  neboť $\tilde{A}^T = ((C^{-1})^T A C^{-1})^T = (C^{-1})^T A ((C^{-1})^T)^T = \tilde{A}$
  a $v^T \tilde{A} v = v^T (C^{-1})^T A C^{-1} v = (C^{-1}v)^T A (C^{-1} v) > 0$ pro lib. v
  jak je to s číslem podmíněnosti $\tilde{A}$? je dáno vlastními čísly $\tilde{A}$ (je SPD)
  a ty jsou stejné jako pro matici
  $$C^{-1} \tilde{A} C = C^{-1} (C^{-1})^T A = (C^T C)^{-1} A$$
  $\Rightarrow$ pokud bychom měli rozumnou SPD matici M, pak lze brát $C^T C$ jako její Choleského rozklad

ovšem lze ukázat, že tento rozklad není nutno dělat, a lze přímo použít algoritmus

$r_0 = b - Ax_0$
vyřeš $Mz_0 = r_0 \to z_0$
$p_0 = z_0$
pro $k = 1, 2, \dots$
  $w_{k-1} = Ap_{k-1}$
  $\alpha_{k-1} = (z_{k-1}^T r_{k-1}) / (p_{k-1}^T w_{k-1})$
  $x_k = x_{k-1} + \alpha_{k-1} p_{k-1}$
  $r_k = r_{k-1} - \alpha_{k-1} w_{k-1}$ $\to$ pokud $\|r_k\| < \varepsilon$ konec
  vyřeš $Mz_k = r_k \to z_k$
  $\beta_{k-1} = (z_k^T r_k) / (z_{k-1}^T r_{k-1})$
  $p_k = z_k + \beta_{k-1} p_{k-1}$

kde
$x_k = C^{-1} \tilde{x}_k$
$p_k = C^{-1} \tilde{p}_k$
$w_k = C^{-1} \tilde{w}_k$
$r_k = C^T \tilde{r}_k$

Konvergence metody sdružených gradientů (CG)

- protože pracujeme se symetrickými, pozitivně definitními maticemi, můžeme pomocí těchto matic definovat normu

$$\|v\|_A = \sqrt{v^T A v} \quad , \text{neboť } v^T A v \text{ je nula pouze pro } v = 0$$

  ta se používá při rozboru metody CG (conjugated gradients)

- pro chybu $e_k = x_k - x^*$ přibližného řešení x_k v k-tém kroku, x^* je přesné řešení $Ax^* = b$, dostaneme

$$\|e_k\|_A^2 = (x_k - x^*)^T A (x_k - x^*) =$$

$$= x_k^T A x_k - 2 x_k^T \underbrace{A x^*}_{b} + (x^*)^T A x^* =$$

$$= 2\phi(x_k) + (x^*)^T A x^* \quad , \text{kde } \phi(x) = \tfrac{1}{2} x^T A x - x^T b$$

  a tedy pokud minimalizujeme $\phi(x)$, minimalizujeme též chybu danou A-normou $\|e\|_A$.

- řešení hledáme na prostoru $x_0 + K_k$, kde

$$K_k = \mathcal{L}(p_0, p_1, \dots, p_{k-1}) = \underbrace{\mathcal{L}(r_0, Ar_0, \dots, A^{k-1} r_0)}_{\text{Krylovův prostor } K_k(r_0, A)} = \mathcal{L}(Ae_0, \dots, A^k e_0)$$

  neboli

$$x_k = x_0 + \alpha_0 p_0 + \alpha_1 p_1 + \dots + \alpha_{k-1} p_{k-1}$$

  z čehož odečtením x^*

$$e_k = e_0 + \alpha_0 p_0 + \dots + \alpha_{k-1} p_{k-1} = e_0 + c_1 A e_0 + \dots + c_k A^k e_0$$

  pro jisté koeficienty $c_1, \dots, c_k$

  neboli

$$e_k = P_k(A) e_0 \quad , \text{kde } P_k(z) = 1 + c_1 z + \dots + c_k z^k \text{ je}$$

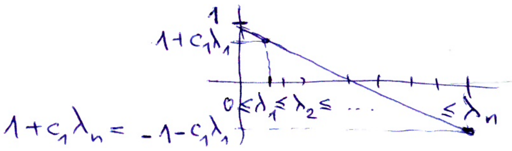
  polynom z prostoru $\mathcal{P}_k = \{ P(z)$ stupně nejvýše k pro nej $P(0) = 1 \}$

- pomocí metody CG konstruujeme přímo polynom P_k, který minimalizuje $\|e_k\|_A = \|P_k(A) e_0\|_A$ na prostoru $\mathcal{P}_k$

• chceme-li najít <u>odhad chyby</u> v k-tém kroku CG, stačí najít polynom v $\mathcal{P}_k$, pro který umíme $P(A)$ nějak rozumně odhadnout, čímž dostaneme odhad i pro $\|e_k\|_A$ neboť pro lib. $\tilde{P}_k \in \mathcal{P}_k$ platí $\|e_k\|_A \leq \|\tilde{P}_k(A) e_0\|_A$

- protože A je diagonalizovatelná (díky SPD), tj. $A = V \Lambda V^T$, kde $\Lambda = \mathrm{diag}(\lambda_j)$ a λ_j, $j = 1, \dots, n$ jsou vlastní čísla A, můžeme psát $$P(A) = V P(\Lambda) V^T \text{, pro lib. polynom}$$
a tedy
$$\|P(A) e_0\|_A^2 = e_0^T P(A)^T A P(A) e_0 =$$
$$= e_0^T V P(\Lambda)^T \underbrace{V^T A V}_{\Lambda} P(\Lambda) V^T e_0 =$$
$$= e_0^T V \,\mathrm{diag}\left(\lambda_j P(\lambda_j)^2\right) V^T e_0 \leq$$
$$\leq \max_j P(\lambda_j)^2 \cdot e_0^T V \Lambda V^T e_0 = \max_j P(\lambda_j)^2 \, \|e_0\|_A^2$$

- odhad ~~můžeme~~ uděláme tak, že vybereme polynom $P(z)$, abychom snadno odhadli jeho velikost ve vlastních číslech λ_j matice A: pro $k=1$ volíme např. $\tilde{P}_1(z) = 1 + c_1 z$ tak, aby nabíval maximálních hodnot na krajích intervalu $\langle \lambda_1, \lambda_n \rangle$, kde leží všechna λ_j



neboli musí být $\tilde{P}_1(z) = 1 - \frac{2z}{\lambda_1 + \lambda_n}$

a pro $\max_j |\tilde{P}_1(\lambda_j)| = 1 - \frac{2\lambda_1}{\lambda_1 + \lambda_n} = \frac{\lambda_n / \lambda_1 - 1}{\lambda_n / \lambda_1 + 1} = \frac{\varkappa(A) - 1}{\varkappa(A) + 1}$

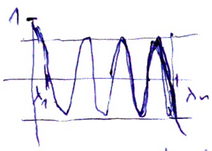
kde $\varkappa(A)$ je číslo podmíněnosti A

pro obecné k lze takový polynom zkonstruovat pomocí Čebyševova polynomu stupně k

$$\tilde{P}_k(z) = \frac{T_k\left(\frac{\lambda_n + \lambda_1 - 2z}{\lambda_n - \lambda_1}\right)}{T_k\left(\frac{\lambda_n + \lambda_1}{\lambda_n - \lambda_1}\right)}$$

{ posunuto a škálováno na interval $\langle \lambda_1, \lambda_n \rangle$

{ „normalizace" aby v $z=0$ bylo $\tilde{P}_k(0) = 1$



lze pak ukázat (viz např. Trefethen), že

$$\max_j |\tilde{P}_k(\lambda_j)| = \tilde{P}_k(\lambda_1) = \frac{\overset{1}{\overbrace{T_k(1)}}}{T_k\left(\frac{\lambda_n + \lambda_1}{\lambda_n - \lambda_1}\right)} = 2\left[\left(\frac{\sqrt{\varkappa}+1}{\sqrt{\varkappa}-1}\right)^k + \left(\frac{\sqrt{\varkappa}+1}{\sqrt{\varkappa}-1}\right)^{-k}\right]^{-1} \leq 2\left(\frac{\sqrt{\varkappa}-1}{\sqrt{\varkappa}+1}\right)^k$$

a tedy
$$\|e_k\|_A \leq \|\tilde{P}_k(A) e_0\|_A \leq 2\left(\frac{\sqrt{\varkappa}-1}{\sqrt{\varkappa}+1}\right)^k \|e_0\|_A = 2\left(1 - \frac{2}{\sqrt{\varkappa}+1}\right)^k \|e_0\|_A$$

a pro určitou přesnost a rozumné $\varkappa$ potřebujeme $\mathcal{O}(\sqrt{\varkappa})$ iterací. „Jde jen o odhad, může konvergovat rychleji"

Modifikace metody sdružených gradientů

- standardní verze CG, povětšinou s předpodmíněním, funguje pro symetrické, pozitivně definitní matice
- přímočaré je zobecnění na hermitovské, pozitivně definitní matice, kdy jen transpozici nahradíme hermitovským sdružením tj. bude $\gamma_k = r_k^+ r_k$ a $\alpha_{k-1} = \gamma_{k-1}/(p_{k-1}^+ w_{k-1})$
- v principu by šlo použít CG na libovolný systém $Ax = b$ když ho vynásobíme A^T, případně A^+, tj. budeme řešit normální rovnice (měli jsme je u metody nejmenších čtverců)

$$A^T A x = A^T b, \quad \text{nebo} \quad A^+ A x = A^+ b \text{ pro komplexní matice}$$

  neboť $A^T A$ a $A^+ A$ jsou již symetrické, pozitivně definitní matice
  - problém však je, že číslo podmíněnosti $\kappa(A^T A) = \kappa^2(A)$, a tedy pro špatně podmíněnou matici A jde o ještě hůře podmíněný systém ⇒ má smysl používat jen pro rozumně podmíněné úlohy
  - pro tuto metodu se používá zkratka CGNR a při její implementaci se $A^T A$ či $A^+ A$ přímo nekonstruují, ale volají se podprogramy na výpočet $A \cdot x$ a $A^T x$ či $A^+ x$.
  - v kombinaci s dobrým předpodmiňovačem může ale jít o rychlou metodu
- pro komplexní symetrické matice $A^T = A$ (ne $A^+ = A$) se používá varianta conjugate orthogonal CG = COCG metoda která nahrazuje skalární součin $x^+ y$ symetrickým součinem $x \cdot y$ bez komplexního sdružení
  - velmi často, zvláště s předpodmíněním, konverguje, ale konvergence není zaručena!
  - velká výhoda (pokud konverguje) je její rychlost, oproti jiným obecnějším metodám

- pro obecný systém $Ax=b$ lze též použít metodu bisdružených gradientů (BiCG), případně její stabilnější verzi BiCGStab, neboť BiCG ne vždy konverguje a není obecně stabilní (neminimalizuje žádné $\phi(x)$)
  - oproti CG metodě potřebujeme kromě Ax též A^Tx (či A^+x) a v průběhu výpočtu se vedle r_k a p_k generují ještě posloupnosti $\tilde{r}_k$ a $\tilde{p}_k$ a je třeba volit kromě x_0 a $p_0=r_0$ ještě $\tilde{p}_0=\tilde{r}_0$ (často se volí $\tilde{r}_0=r_0$, ale obecně to tak být nemusí)
  - rezidua nejsou nyní ortogonální navzájem, ale platí
    $$\tilde{r}_i^T r_j = r_i^T \tilde{r}_j = 0 \quad \text{pro } j<i, \text{ tzv. biortogonalita}$$
    $$\tilde{p}_i^T A p_j = p_i^T A^T \tilde{p}_j = 0 \quad \text{pro } j<i, \text{ tzv. bisdruženost}$$
    a navíc $\quad \tilde{r}_i^T p_j = r_i^T p_j \quad$ pro $j<i$
  - algoritmus včetně předpodmínění maticí M:

    spočti $r_0 = b - Ax_0$ pro zvolené x_0
    zvol $\tilde{r}_0$ (např. $=r_0$)
    pro $i=1,2,\dots$
      vyřeš $Mz_{i-1}=r_{i-1}$ a $M^T\tilde{z}_{i-1}=\tilde{r}_{i-1}$
      $\gamma_{i-1} = z_{i-1}^T \tilde{r}_{i-1}$
      pokud $\gamma_{i-1}=0$, tak konec (metoda selhala, dělení nulou)
      pokud $i=1$ — někdy se tomu lze vyhnout např. restartováním metody apod.
        $p_i = z_{i-1}$, $\tilde{p}_i = \tilde{z}_{i-1}$
      jinak
        $\beta_{i-1} = \gamma_{i-1}/\gamma_{i-2}$
        $p_i = z_{i-1} + \beta_{i-1} p_{i-1}$
        $\tilde{p}_i = \tilde{z}_{i-1} + \beta_{i-1}\tilde{p}_{i-1}$
      $q_i = Ap_i$, $\tilde{q}_i = A^T\tilde{p}_i$
      $\alpha_i = \gamma_{i-1}/(\tilde{p}_i^T q_i)$
      $x_i = x_{i-1} + \alpha_i p_i$
      $r_i = r_{i-1} - \alpha_i q_i$
      $\tilde{r}_i = \tilde{r}_{i-1} - \alpha_i \tilde{q}_i$
      test konvergence ($\|r_i\|_2 < \varepsilon$ apod.)

  - pro symetrické, pozitivně definitní matice totéž, co CG, ale dvakrát náročnější