# Preconditioning Conjugate Gradient method

**Tsung-Ming Huang**

Matrix Computation, 2016, NTNU

# Plan

- Gradient method
- Conjugate gradient method
- Preconditioner

# Gradient method

# Theorem $Ax = b, \quad A:$ s.p.d

**Definition**

$A:$ symmetric positive definite if

$A^\top = A$

$x^\top A x > 0, \forall\, x \neq 0$

**Inner product**

$$< x, y > = x^\top y \text{ for any } x, y \in \mathbb{R}^n$$

**Define**

$$g(x) = < x, Ax > -2 < x, b > = x^\top A x - 2x^\top b$$

**Theorem** $A:$ s.p.d

$x^*$ is the sol. of $Ax = b$ ⟷ $g(x^*) = \min_{x \in \mathbb{R}^n} g(x)$

# Proof

**Assume** $x^*$ is the sol. of $Ax=b$ → $Ax^*=b$

$$g(x)=<x,Ax>-2<x,b>$$

$$=<x-x^*,A(x-x^*)>+<x,Ax^*>+<x^*,Ax>$$

$$-<x^*,Ax^*>-2<x,b>$$

$$=<x-x^*,A(x-x^*)>-<x^*,Ax^*>$$

$$+2<x,Ax^*>-2<x,b>$$

$$=<x-x^*,A(x-x^*)>-<x^*,Ax^*>+2<x,Ax^*-b>$$

$$=<x-x^*,A(x-x^*)>-<x^*,Ax^*>$$

$<x-x^*,A(x-x^*)>\geq 0$ → $g(x^*)=\min_{x\in\mathbb{R}^n} g(x)$

# Proof

**Assume** $g(x^*) = \min_{x \in \mathbb{R}^n} g(x)$

Fixed vectors $x$ and $v$, for any $\alpha \in \mathbb{R}$

$$f(\alpha) \equiv g(x+\alpha v) = < x+\alpha v, Ax+\alpha Av > -2 < x+\alpha v, b >$$

$$= < x, Ax > + \alpha < v, Ax > + \alpha < x, Av > + \alpha^2 < v, Av >$$

$$-2 < x, b > -2\alpha < v, b >$$

$$= < x, Ax > -2 < x, b > +2\alpha < v, Ax >$$

$$-2\alpha < v, b > + \alpha^2 < v, Av >$$

$$= g(x) + 2\alpha < v, Ax-b > + \alpha^2 < v, Av >$$

# Proof

$$f(\alpha) = g(x) + 2\alpha < v, Ax - b > + \alpha^2 < v, Av >$$

→ $f$ is a quadratic function of $\alpha$

$A:$ s.p.d → $f$ has a minimal value when $f'(\alpha) = 0$

$$f'(\hat{\alpha}) = 2 < v, Ax - b > + 2\hat{\alpha} < v, Av > = 0$$

→ $$\hat{\alpha} = -\frac{< v, Ax - b >}{< v, Av >} = \frac{< v, b - Ax >}{< v, Av >}$$

→ $$g(x + \hat{\alpha} v) = f(\hat{\alpha}) = g(x) - 2\frac{< v, b - Ax >}{< v, Av >} < v, b - Ax >$$

$$+ \left( \frac{< v, b - Ax >}{< v, Av >} \right)^2 < v, Av > = g(x) - \frac{< v, b - Ax >^2}{< v, Av >}$$

# Proof

$$g(x+\hat{\alpha}v) = g(x) - \frac{<v, b-Ax>^2}{<v, Av>}$$

$\forall\, v \neq 0$

$\rightarrow$ $$g(x+\hat{\alpha}v) < g(x) \;\text{ if }\; <v, b-Ax> \neq 0$$

$$g(x+\hat{\alpha}v) = g(x) \;\text{ if }\; <v, b-Ax> = 0$$

Suppose that

$$g(x^*) = \min_{x\in\mathbb{R}^n} g(x)$$

$\rightarrow$ $$g(x^*+\hat{\alpha}v) \geq g(x^*) \;\text{ for any }\; v$$

$\rightarrow$ $<v, b-Ax^*> = 0, \forall\, v$ $\rightarrow$ $Ax^* = b$

$$\alpha = \frac{< v, b - Ax >}{< v, Av >} = \frac{< v, r >}{< v, Av >}, \quad r \equiv b - Ax$$

If $r \neq 0$ and $< v, r > \neq 0$

→ $$g(x + \alpha v) = g(x) - \frac{< v, b - Ax >^2}{< v, Av >} < g(x)$$

→ $x + \alpha v$ is closer to $x^*$ than is $x$

Given $x^{(0)}$ and $v^{(1)} \neq 0$

For $k = 1, 2, 3, \cdots$

$$\alpha_k = \frac{< v^{(k)}, b - Ax^{(k-1)} >}{< v^{(k)}, Av^{(k)} >}, \quad x^{(k)} = x^{(k-1)} + \alpha_k v^{(k)}$$

Choose a new search direction $v^{(k+1)}$

# Steepest descent

**Question** How to choose $\{v^{(k)}\}$ s.t. $\{x^{(k)}\} \to x^*$ rapidly?

Let $\Phi : \mathbb{R}^n \to \mathbb{R}$ be a differentiable function on $x$

$$\frac{\Phi(x+\varepsilon p) - \Phi(x)}{\varepsilon} = \nabla\Phi(x)^\top p + \mathcal{O}(\varepsilon)$$

The right hand side takes minimum at

$$p = -\frac{\nabla\Phi(x)}{\|\nabla\Phi(x)\|} \quad \text{(i.e., the largest descent)}$$

for all $p$ with $\|p\| = 1$ (neglect $\mathcal{O}(\varepsilon)$ )

# Steepest descent direction of g

Denote $x = [x_1, x_2, \cdots, x_n]^\top$

$$g(x) = < x, Ax > -2 < x, b > \; = \sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij} x_i x_j - 2\sum_{i=1}^{n} x_i b_i$$

$$\frac{\partial g}{\partial x_k}(x) = 2\sum_{i=1}^{n} a_{ki} x_i - 2b_k = 2\big(A(k,:)x - b_k\big)$$

$$\nabla g(x) = \left[\frac{\partial g}{\partial x_1}(x), \frac{\partial g}{\partial x_2}, \cdots, \frac{\partial g}{\partial x_n}(x)\right]^\top = 2(Ax - b) = -2r$$

# Steepest descent method (gradient method)

Given $x^{(0)} \neq 0$

For $k = 1, 2, 3, \cdots$

$r_{k-1} = b - Ax^{(k-1)}$

If $r_{k-1} = 0$, then

Stop;

Else

$$\alpha_k = \frac{< r_{k-1}, r_{k-1} >}{< r_{k-1}, Ar_{k-1} >}$$

$$x^{(k)} = x^{(k-1)} + \alpha_k r_{k-1}$$

End

End

**Convergence Theorem**

$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n > 0$ : eigenvalues

$x^{(k)}, x^{(k-1)}$ : approx. sol.

$x^*$ : exact sol.

$$\Rightarrow \| x^{(k)} - x^* \|_A \leq \left( \frac{\lambda_1 - \lambda_n}{\lambda_1 + \lambda_n} \right) \| x^{(k-1)} - x^* \|_A$$

where $\| x \|_A = \sqrt{x^\top A x}$

# Conjugate gradient method

# A-orthogonal

If $\kappa(A) = \dfrac{\lambda_1}{\lambda_n}$ is large → $\dfrac{\lambda_1 - \lambda_n}{\lambda_1 + \lambda_n} \approx 1$

→ Convergence is very slow

→ NOT recommend it

**Improvement** Choose A-orthogonal search directions

**Definition**

$p, q \in \mathbb{R}^n$ are called A-orthogonal (A-conjugate) if

$$p^\top A q = 0$$

# Lemma

$$v_1, \ldots, v_n \neq 0: \text{ pairwisely A-conjugate}$$

$\Rightarrow$ $v_1, \ldots, v_n$ : linearly independent

**Proof**

$$0 = \sum_{j=1}^{n} c_j v_j$$

$$\Rightarrow \quad 0 = (v_k)^\top A \left( \sum_{j=1}^{n} c_j v_j \right) = \sum_{j=1}^{n} c_j (v_k)^\top A v_j = c_k (v_k)^\top A v_k$$

$\Rightarrow$ $c_k = 0, k = 1, \ldots, n$

$\Rightarrow$ $v_1, \ldots, v_n$ : linearly independent

# Theorem

$A$ : symmetric positive definite

$v_1, \ldots, v_n \neq 0 \in \mathbb{R}^n$ : pairwisely A-conjugate

$x_0$ : given

For $k = 1, \ldots, n,$ let

$$\alpha_k = \frac{< v_k, b - Ax_{k-1} >}{< v_k, Av_k >}$$

$$x_k = x_{k-1} + \alpha_k v_k$$

Then

$$Ax_n = b$$

$$< b - Ax_k, v_j > = 0, \text{ for } j = 1, 2, \ldots, k$$

# Proof

$$x_k = x_{k-1} + \alpha_k v_k$$

$$\Rightarrow \quad Ax_n = Ax_{n-1} + \alpha_n Av_n = (Ax_{n-2} + \alpha_{n-1} Av_{n-1}) + \alpha_n Av_n$$

$$= \cdots = Ax_0 + \alpha_1 Av_1 + \alpha_2 Av_2 + \cdots + \alpha_n Av_n$$

$$\Rightarrow \quad < Ax_n - b, v_k >$$

$$= < Ax_0 - b, v_k > + \alpha_1 < Av_1, v_k > + \cdots + \alpha_n < Av_n, v_k >$$

$$= < Ax_0 - b, v_k > + \alpha_1 < v_1, Av_k > + \cdots + \alpha_n < v_n, Av_k >$$

$$= < Ax_0 - b, v_k > + \alpha_k < v_k, Av_k >$$

$$= < Ax_0 - b, v_k > + \frac{< v_k, b - Ax_{k-1} >}{< v_k, Av_k >} < v_k, Av_k >$$

$$= < Ax_0 - b, v_k > + < v_k, b - Ax_{k-1} >$$

# Proof

$$<Ax_n - b, v_k> = <Ax_0 - b, v_k> + <v_k, b - Ax_{k-1}>$$

$$= <Ax_0 - b, v_k>$$
$$+ <v_k, b - Ax_0 + Ax_0 - Ax_1 + \cdots - Ax_{k-2} + Ax_{k-2} - Ax_{k-1}>$$

$$= <Ax_0 - b, v_k> + <v_k, b - Ax_0>$$
$$+ <v_k, Ax_0 - Ax_1> + \cdots + <v_k, Ax_{k-2} - Ax_{k-1}>$$

$$= <v_k, Ax_0 - Ax_1> + \cdots + <v_k, Ax_{k-2} - Ax_{k-1}>$$

$x_i = x_{i-1} + \alpha_i v_i, \forall i$ → $Ax_i = Ax_{i-1} + \alpha_i A v_i$

→ $Ax_{i-1} - Ax_i = -\alpha_i A v_i$

$$<Ax_n - b, v_k> = -\alpha_1 <v_k, Av_1> - \cdots - \alpha_{k-1} <v_k, Av_{k-1}> = 0$$

→ $Ax_n = b$

# Proof $<b - Ax_k, v_j> = 0, \text{ for } j = 1,2,\ldots,k$

**Assume** $<r_{k-1}, v_j> = 0, \text{ for } j = 1,2,\ldots,k-1$

$$r_k = b - Ax_k = b - A(x_{k-1} + \alpha_k v_k) = r_{k-1} - \alpha_k A v_k$$

$$\Rightarrow \quad <r_k, v_k> = <r_{k-1}, v_k> - \alpha_k <Av_k, v_k>$$

$$= <r_{k-1}, v_k> - \frac{<v_k, b - Ax_{k-1}>}{<v_k, Av_k>} <Av_k, v_k> = 0$$

For $j = 1,\ldots,k-1$

$$<r_k, v_j> = \underbrace{<r_{k-1}, v_j>}_{\text{Assumption}} - \alpha_k \underbrace{<Av_k, v_j>}_{\text{A-conjugate}} = 0$$

which is completed the proof by the mathematic induction.

# Method of conjugate directions

Given $x^{(0)}$, $v_1,\ldots,v_n \in \mathbb{R}^n \setminus \{0\}$: pairwisely A-orthogonal

$r_0 = b - Ax^{(0)}$

For $k = 1,\ldots,n$

$$\alpha_k = \frac{< v_k, r_{k-1} >}{< v_k, Av_k >}, \quad x^{(k)} = x^{(k-1)} + \alpha_k v_k$$

$$r_k = r_{k-1} - \alpha_k A v_k = b - Ax^{(k-1)}$$

End

**Question**

How to find A-orthogonal search directions?

# A-orthogonalization

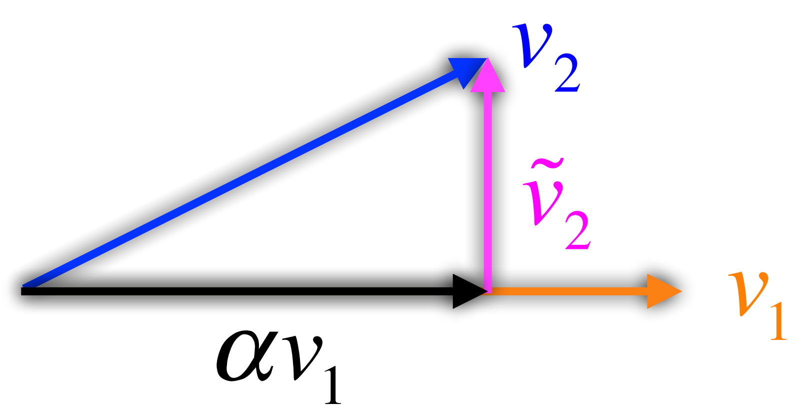
$$\tilde{v}_2 = v_2 - \alpha v_1 \perp v_1$$

↓

$$0 = v_1^\top \tilde{v}_2 = v_1^\top v_2 - \alpha v_1^\top v_1$$

↓

$$\alpha = \frac{v_1^\top v_2}{v_1^\top v_1}$$




**A-orthogonal**

$$\tilde{v}_2 = v_2 - \alpha v_1 \perp_A v_1$$

↓

$$0 = v_1^\top A \tilde{v}_2 = v_1^\top A v_2 - \alpha v_1^\top A v_1$$

↓

$$\alpha = \frac{v_1^\top A v_2}{v_1^\top A v_1}$$

# A-orthogonalization

$$\tilde{v}_2 = v_2 - \frac{v_1^\top A v_2}{v_1^\top A v_1} v_1 \ \perp_A \ v_1$$

$$\{v_1, v_2\} \rightarrow \{v_1, \tilde{v}_2\} : A\text{-orthogonal}$$

$$\{v_1, v_2, v_3\} \rightarrow \{v_1, \tilde{v}_2, \tilde{v}_3\} : A\text{-orthogonal}$$

$$\tilde{v}_3 = v_3 - \alpha_1 v_1 - \alpha_2 \tilde{v}_2 \ \perp_A \ \{v_1, \tilde{v}_2\}$$

$$\rightarrow 0 = v_1^\top A \tilde{v}_3 = v_1^\top A v_3 - \alpha_1 v_1^\top A v_1 \rightarrow \alpha_1 = v_1^\top A v_3 / v_1^\top A v_1$$

$$\rightarrow 0 = \tilde{v}_2^\top A \tilde{v}_3 = \tilde{v}_2^\top A v_3 - \alpha_2 \tilde{v}_2^\top A \tilde{v}_2 \rightarrow \alpha_2 = \tilde{v}_2^\top A v_3 / \tilde{v}_2^\top A \tilde{v}_2$$

# Practical Implementation

Given $x^{(0)}$ → $r_0 = b - Ax^{(0)}$ → $v_1 = r_0$

→ $\alpha_1 = \dfrac{<v_1, r_0>}{<v_1, Av_1>}, \quad x^{(1)} = x^{(0)} + \alpha_1 v_1$

→ $r_1 = r_0 - \alpha_1 A v_1$ steepest descent direction

→ $\{v_1, r_1\}$ NOT A-orthogonal set

**Construct A-orthogonal vector**

$$v_2 = r_1 + \beta_1 v_1, \quad \beta_1 = -\frac{<v_1, Ar_1>}{<v_1, Av_1>}$$

→ $\alpha_2 = \dfrac{<v_2, r_1>}{<v_2, Av_2>}, \quad x^{(2)} = x^{(1)} + \alpha_2 v_2$ → $r_2 = r_1 - \alpha_2 A v_2$

# Construct A-orthogonal vector

$$\{v_1, v_2, r_2\}$$

$$v_3 = r_2 + \beta_{21} v_1 + \beta_{22} v_2, \quad \beta_{21} = -\frac{v_1^\top A r_2}{v_1^\top A v_1}, \ \beta_{22} = -\frac{v_2^\top A r_2}{v_2^\top A v_2}$$

$$r_1 = r_0 - \alpha_1 A v_1 \quad \rightarrow \quad v_1^\top A r_2 = r_2^\top A v_1 = \alpha_1^{-1}\left(r_2^\top r_0 - r_2^\top r_1\right)$$

$$v_2^\top r_2 = v_2^\top r_1 - \alpha_2 v_2^\top A v_2 = v_2^\top r_1 - \frac{v_2^\top r_1}{v_2^\top A v_2} v_2^\top A v_2 = 0$$

$$0 = v_2^\top r_2 = \left(r_1^\top + \beta_1 v_1^\top\right) r_2 = r_1^\top r_2 + \beta_1 v_1^\top r_2$$

$$= r_1^\top r_2 + \beta_1 v_1^\top \left(r_1 - \alpha_2 A v_2\right) = r_1^\top r_2 + \beta_1 v_1^\top r_1$$

$$= r_1^\top r_2 + \beta_1 v_1^\top \left(r_0 - \alpha_1 A v_1\right) = r_1^\top r_2 + \beta_1 \left( v_1^\top r_0 - \frac{< v_1, r_0 >}{< v_1, A v_1 >} v_1^\top A v_1 \right)$$

$$= r_1^\top r_2$$

$$r_1 = r_0 - \alpha_1 A v_1, \quad \alpha_1 = \frac{< v_1, r_0 >}{< v_1, A v_1 >}$$

$$\Rightarrow \quad < v_1, r_1 > = < v_1, r_0 > - \alpha_1 < v_1, A v_1 > = 0$$

$$\Rightarrow \quad < r_2, r_0 > = < r_2, v_1 > = < r_1, v_1 > - \alpha_2 < A v_2, v_1 > = 0$$

$$\Rightarrow \quad v_1^\top A r_2 = \alpha_1^{-1} \left( r_2^\top r_0 - r_2^\top r_1 \right) = 0$$

$$\Rightarrow \quad \beta_{21} = -\frac{v_1^\top A r_2}{v_1^\top A v_1} = 0$$

$$\Rightarrow \quad v_3 = r_2 + \beta_2 v_2, \quad \beta_2 = -\frac{v_2^\top A r_2}{v_2^\top A v_2}$$

# In general case

$$v_k = r_{k-1} + \beta_{k-1} v_{k-1} \quad \text{if } \; r_{k-1} \neq 0$$

$$\rightarrow \quad 0 = < v_{k-1}, Av_k > = < v_{k-1}, Ar_{k-1} + \beta_{k-1} Av_{k-1} >$$

$$= < v_{k-1}, Ar_{k-1} > + \beta_{k-1} < v_{k-1}, Av_{k-1} >$$

$$\rightarrow \quad \beta_{k-1} = -\frac{< v_{k-1}, Ar_{k-1} >}{< v_{k-1}, Av_{k-1} >}$$

**Theorem**

(i). $\{r_0, r_1, \ldots, r_{k-1}\}$ is an orthogonal set

(ii). $\{v_1, \ldots, v_k\}$ is an A-orthogonal set

# Reformula $\alpha_k, \beta_k$

$$v_k = r_{k-1} + \beta_{k-1} v_{k-1}$$

$$\alpha_k = \frac{<v_k, r_{k-1}>}{<v_k, Av_k>} = \frac{<r_{k-1} + \beta_{k-1} v_{k-1}, r_{k-1}>}{<v_k, Av_k>}$$

$$= \frac{<r_{k-1}, r_{k-1}>}{<v_k, Av_k>} + \beta_{k-1} \frac{<v_{k-1}, r_{k-1}>}{<v_k, Av_k>} = \frac{<r_{k-1}, r_{k-1}>}{<v_k, Av_k>}$$

$$<r_{k-1}, r_{k-1}> = \boxed{\alpha_k <v_k, Av_k>}$$

$$r_k = r_{k-1} - \alpha_k A v_k$$

$$<r_k, r_k> = <r_{k-1}, r_k> - \alpha_k <Av_k, r_k> = \boxed{-\alpha_k <r_k, Av_k>}$$

$$\beta_k = -\frac{<v_k, Ar_k>}{<v_k, Av_k>} = -\frac{<r_k, Av_k>}{<v_k, Av_k>} = \frac{<r_k, r_k>}{<r_{k-1}, r_{k-1}>}$$

# Algorithm (Conjugate Gradient Method)

Given $x^{(0)}$, compute $r_0 = b - Ax^{(0)} = v_0$

For $k = 0,1,\ldots$

$$\alpha_k = \frac{< r_k, r_k >}{< v_k, Av_k >}, \quad x^{(k+1)} = x^{(k)} + \alpha_k v_k$$

$$r_{k+1} = r_k - \alpha_k A v_k$$

If $r_{k+1} = 0$, then

Stop;

Else

$$\beta_k = \frac{< r_{k+1}, r_{k+1} >}{< r_k, r_k >}, \quad v_{k+1} = r_{k+1} + \beta_k v_k$$

End

End

Theorem

$$Ax_n = b$$

well-conditioned

$$\| r_{\sqrt{n}} \| < tol$$

ill-conditioned

$$\| r_k \| < tol$$

$$k > n$$

# Conjugate Gradient Method

## Convergence Theorem

$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n > 0$ : eigenvalues

$\left\{x^{(k)}\right\}$ : produced by CG method

$x^*$ : exact sol.

CG is much better than Gradient method

$$\| x^{(k)} - x^* \|_A \leq 2\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^k \| x_0 - x^* \|_A, \quad \kappa = \frac{\lambda_1}{\lambda_n}$$

$\left\{x_G^{(k)}\right\}$ : produced by Gradient method

$$\frac{\kappa-1}{\kappa+1} > \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}$$

$$\| x_G^{(k)} - x^* \|_A \leq \left(\frac{\lambda_1 - \lambda_n}{\lambda_1 + \lambda_n}\right)^k \| x_G^{(0)} - x^* \|_A = \left(\frac{\kappa-1}{\kappa+1}\right)^k \| x_G^{(0)} - x^* \|_A$$

# Preconditioner

$$Ax = b \longrightarrow \underbrace{C^{-1}AC^{-\top}}_{\tilde{A}}\underbrace{C^{\top}x}_{\tilde{x}} = \underbrace{C^{-1}b}_{\tilde{b}}$$

**Goal** Choose $C$ such that $\kappa(C^{-1}AC^{-\top}) < \kappa(A)$

Apply CG method to $\tilde{A}\tilde{x} = \tilde{b}$ → Get $\tilde{x}$ → Solve $x = C^{-\top}\tilde{x}$

**Nothing NEW**

**Question**

Apply CG method to $\tilde{A}\tilde{x} = \tilde{b}$ → Get $x$

# Algorithm (Conjugate Gradient Method)

Given $\tilde{x}^{(0)}$, compute $\tilde{r}_0 = \tilde{b} - \tilde{A}\tilde{x}^{(0)} = \tilde{v}_0$

For $k = 0, 1, \ldots$

$$\tilde{\alpha}_k = \frac{<\tilde{r}_k, \tilde{r}_k>}{<\tilde{v}_k, \tilde{A}\tilde{v}_k>}$$

$$\tilde{x}^{(k+1)} = \tilde{x}^{(k)} + \tilde{\alpha}_k \tilde{v}_k$$

$$\tilde{r}_{k+1} = \tilde{b} - \tilde{A}\tilde{x}^{(k+1)} = C^{-1} r_{k+1}$$

If $\tilde{r}_{k+1} = 0$, then Stop

$$\tilde{\beta}_k = \frac{<\tilde{r}_{k+1}, \tilde{r}_{k+1}>}{<\tilde{r}_k, \tilde{r}_k>} = \frac{<w_{k+1}, w_{k+1}>}{<w_k, w_k>}$$

$$\tilde{v}_{k+1} = \tilde{r}_{k+1} + \tilde{\beta}_k \tilde{v}_k$$

End

$$\tilde{r}_{k+1} = C^{-1}b - \left(C^{-1}AC^{-T}\right)C^{\top}x_{k+1} = C^{-1}(b - Ax_{k+1}) = C^{-1}r_{k+1}$$

Let

$$\tilde{v}_k = C^{\top}v_k, \quad w_k = C^{-1}r_k$$

$$\tilde{\beta}_k = \frac{<C^{-1}r_{k+1}, C^{-1}r_{k+1}>}{<C^{-1}r_k, C^{-1}r_k>} = \frac{<w_{k+1}, w_{k+1}>}{<w_k, w_k>}$$

# Algorithm (Conjugate Gradient Method)

Given $\tilde{x}^{(0)}$, compute $\tilde{r}_0 = \tilde{b} - \tilde{A}\tilde{x}^{(0)} = \tilde{v}_0$

For $k = 0, 1, \ldots$

$$\tilde{\alpha}_k = \frac{<\tilde{r}_k, \tilde{r}_k>}{<\tilde{v}_k, \tilde{A}\tilde{v}_k>} = \frac{<w_k, w_k>}{<v_k, Av_k>}$$

$$\tilde{x}^{(k+1)} = \tilde{x}^{(k)} + \tilde{\alpha}_k \tilde{v}_k$$

$$\tilde{r}_{k+1} = C^{-1} r_{k+1}$$

If $\tilde{r}_{k+1} = 0$, then Stop

$$\tilde{\beta}_k = \frac{<w_{k+1}, w_{k+1}>}{<w_k, w_k>}$$

$$\tilde{v}_{k+1} = \tilde{r}_{k+1} + \tilde{\beta}_k \tilde{v}_k$$

End

$$\tilde{\alpha}_k = \frac{<C^{-1}r_k, C^{-1}r_k>}{<C^\top v_k, C^{-1}AC^{-\top}C^\top v_k>} = \frac{<w_k, w_k>}{<C^\top v_k, C^{-1}Av_k>}$$

$$<C^\top v_k, C^{-1}Av_k> = v_k^\top CC^{-1}Av_k = v_k^\top Av_k$$

$$\rightarrow \quad \tilde{\alpha}_k = \frac{<w_k, w_k>}{<v_k, Av_k>}$$

# Algorithm (Conjugate Gradient Method)

Given $\tilde{x}^{(0)}$, compute $\tilde{r}_0 = \tilde{b} - \tilde{A}\tilde{x}^{(0)} = \tilde{v}_0$

For $k = 0,1,\ldots$

$$\tilde{\alpha}_k = \frac{< w_k, w_k >}{< v_k, Av_k >}$$

$$\tilde{x}^{(k+1)} = \tilde{x}^{(k)} + \tilde{\alpha}_k \tilde{v}_k$$

$$\tilde{r}_{k+1} = \tilde{r}_k - \tilde{\alpha}_k \tilde{A} \tilde{v}_k$$

If $\tilde{r}_{k+1} = 0$, then Stop

$$\tilde{\beta}_k = \frac{< w_{k+1}, w_{k+1} >}{< w_k, w_k >}$$

$$\tilde{v}_{k+1} = \tilde{r}_{k+1} + \tilde{\beta}_k \tilde{v}_k$$

End

$$C^\top x^{(k+1)} = C^\top x^{(k)} + \tilde{\alpha}_k C^\top v_k$$

$$\rightarrow \quad x^{(k+1)} = x^{(k)} + \tilde{\alpha}_k v_k$$

$$C^{-1} r_{k+1} = C^{-1} r_k - \tilde{\alpha}_k C^{-1} A C^{-\top} C^\top v_k$$

$$\rightarrow \quad r_{k+1} = r_k - \tilde{\alpha}_k A v_k$$

$$\tilde{v}_k = C^\top v_k, \quad w_k = C^{-1} r_k$$

$$C^\top v_{k+1} = C^{-1} r_{k+1} + \tilde{\beta}_k C^\top v_k$$

$$\rightarrow \quad v_{k+1} = C^{-\top} C^{-1} r_{k+1} + \tilde{\beta}_k v_k$$

$$= C^{-\top} w_{k+1} + \tilde{\beta}_k v_k$$

# Algorithm (Conjugate Gradient Method)

Given $\tilde{x}^{(0)}$, compute $\tilde{r}_0 = \tilde{b} - \tilde{A}\tilde{x}^{(0)} = \tilde{v}_0$

For $k = 0, 1, \ldots$

$$\tilde{\alpha}_k = \frac{< w_k, w_k >}{< v_k, A v_k >}$$

need $w_0$

need $v_0$

$$x^{(k+1)} = x^{(k)} + \tilde{\alpha}_k v_k$$

$$r_{k+1} = r_k - \tilde{\alpha}_k A v_k$$

If $r_{k+1} = 0$, then Stop

$$\tilde{\beta}_k = \frac{< w_{k+1}, w_{k+1} >}{< w_k, w_k >}$$

Solve $C\, w_{k+1} = r_{k+1}$

$$v_{k+1} = C^{-\top} w_{k+1} + \tilde{\beta}_k v_k$$

End

$$w_k = C^{-1} r_k$$

$$w_0 = C^{-1} r_0 = C^{-1}(b - A x^{(0)})$$

$$v_{k+1} = C^{-\top} w_{k+1} + \tilde{\beta}_k v_k$$

$$v_0 = C^{-\top} w_0$$

# Algorithm (CG Method with preconditioner C)

Given $C$ and $x^{(0)}$, compute $r_0 = b - Ax^{(0)}$

Solve $Cw_0 = r_0$ and $C^\top v_1 = w_0$

For $k = 0,1,\ldots$

$$\alpha_k = < w_k, w_k > / < v_k, Av_k >$$

$$x^{(k+1)} = x^{(k)} + \alpha_k v_k$$

$$r_{k+1} = r_k - \alpha_k A v_k$$

If $r_{k+1} = 0$, then Stop

Solve $Cw_{k+1} = r_{k+1}$ and $C^\top z_{k+1} = w_{k+1}$

$$\beta_k = < w_{k+1}, w_{k+1} > / < w_k, w_k >$$

$$v_{k+1} = z_{k+1} + \beta_k v_k$$

End

$$r_{k+1} = CC^\top z_{k+1} \equiv M z_{k+1}$$

$$\beta_k = \frac{< C^{-1} r_{k+1}, C^{-1} r_{k+1} >}{< C^{-1} r_k, C^{-1} r_k >} = \frac{< z_{k+1}, r_{k+1} >}{< z_k, r_k >}$$

$$\alpha_k = \frac{< C^{-1} r_k, C^{-1} r_k >}{< C^\top v_k, C^{-1} A v_k >} = \frac{< z_k, r_k >}{< v_k, A v_k >}$$

# Algorithm (CG Method with preconditioner M)

Given $M$ and $x^{(0)}$, compute $r_0 = b - Ax^{(0)}$

Solve $Mz_0 = r_0$ and set $v_1 = z_0$

For $k = 0,1,\ldots$

Compute $\alpha_k = < z_k, r_k > / < v_k, Av_k >$

Compute $x^{(k+1)} = x^{(k)} + \alpha_k v_k$

Compute $r_{k+1} = r_k - \alpha_k Av_k$

If $r_{k+1} = 0$, then Stop

Solve $Mz_{k+1} = r_{k+1}$

Compute $\beta_k = < z_{k+1}, r_{k+1} > / < z_k, r_k >$

Compute $v_{k+1} = z_{k+1} + \beta_k v_k$

End

# Choices of M (Criterion)

- cond$(M^{-1/2}AM^{-1/2})$ is nearly by 1, i.e.,

$$M^{-1/2}AM^{-1/2} \approx I, \quad A \approx M$$

- The linear system $Mz = r$ must be easily solved. e.g.

$$M = LL^\top$$

- $M$ is symmetric positive definite

# Preconditioner M

- Jacobi method $A = D + (L + U), \quad M = D$

$$x_{k+1} = -D^{-1}(L+U)x_k + D^{-1}b$$
$$= -D^{-1}(A-D)x_k + D^{-1}b$$
$$= x_k + D^{-1}r_k$$

- Gauss-Seidel $A = (D + L) + U, \quad M = D + L$

$$x_{k+1} = -(D+L)^{-1}Ux_k + (D+L)^{-1}b$$
$$= (D+L)^{-1}(D+L-A)x_k + (D+L)^{-1}b$$
$$= x_k + (D+L)^{-1}r_k$$

# Preconditioner M

- SOR: $\omega A = (D+\omega L) - ((1-\omega)D - \omega U) \equiv M - N$

$$x_{k+1} = (D+\omega L)^{-1}\left[(1-\omega)D - \omega U\right]x_k + \omega(D+\omega L)^{-1}b$$

$$= (D+\omega L)^{-1}\left[(D+\omega L) - \omega A\right]x_k + \omega(D+\omega L)^{-1}b$$

$$= \left[I - \omega(D+\omega L)^{-1}A\right]x_k + \omega(D+\omega L)^{-1}b$$

$$= x_k + \omega(D+\omega L)^{-1}r_k$$

- SSOR: $M(\omega) = \dfrac{1}{\omega(2-\omega)}(D+\omega L)D^{-1}\left(D+\omega L^{\top}\right)$