# Conjugate Gradients II: CG and Variants

CS 205A:
Mathematical Methods for Robotics, Vision, and Graphics

Justin Solomon

# Idea

"Easy to apply,
hard to invert."

$$\min_{\vec{x}} \left[\frac{1}{2}\vec{x}^\top A\vec{x} - \vec{b}^\top\vec{x} + c\right]$$

# Line Search Along $\vec{d}$ from $\vec{x}$

$$\min_{\alpha} g(\alpha) \equiv f(\vec{x} + \alpha\vec{d})$$

$$\alpha = \frac{\vec{d}^\top(\vec{b} - A\vec{x})}{\vec{d}^\top A\vec{d}} = \frac{\vec{d}^\top \vec{r}}{\vec{d}^\top A\vec{d}}$$

# Conjugate Directions

## $A$-Conjugate Vectors

Two vectors $\vec{v}$ and $\vec{w}$ are $A$-conjugate if $\vec{v}^\top A\vec{w} = 0$.

## Corollary

Suppose $\{\vec{v}_1, \ldots, \vec{v}_n\}$ are $A$-conjugate. Then, $f$ is minimized in at most $n$ step by line searching in direction $\vec{v}_1$, then direction $\vec{v}_2$, and so on.

# Another Clue

$$\vec{r}_k \equiv \vec{b} - A\vec{x}_k$$

$$\vec{r}_{k+1} = \vec{r}_k - \alpha_{k+1} A\vec{v}_{k+1}$$

## Proposition

When performing gradient descent on $f$, $\text{span}\,\{\vec{r}_0, \ldots, \vec{r}_k\} = \text{span}\,\{\vec{r}_0, A\vec{r}_0, \ldots, A^k\vec{r}_0\}$.

# Gradient Descent: Issue

$$\vec{x}_k - \vec{x}_0 \neq \underset{\vec{v} \in \operatorname{span}\{\vec{r}_0, A\vec{r}_0, \ldots, A^{k-1}\vec{r}_0\}}{\arg\min} f(\vec{x}_0 + \vec{v})$$

**But if this did hold...**

Convergence in $n$ steps!

# Outline for CG

1. [Somehow] generate search direction $\vec{v}_k$

   (initialize to $\vec{r}_0$)

2. Line search: $\alpha_k = \frac{\vec{v}_k^\top \vec{r}_{k-1}}{\vec{v}_k^\top A \vec{v}_k}$

3. Update estimate: $\vec{x}_k = \vec{x}_{k-1} + \alpha_k \vec{v}_k$

4. Update residual: $\vec{r}_k = \vec{r}_{k-1} - \alpha_k A \vec{v}_k$

# What We (Greedily) Want

1. Easy way to generate $n$ conjugate directions $\{\vec{v}_1, \ldots, \vec{v}_n\}$
2. $\operatorname{span}\{\vec{v}_1, \ldots, \vec{v}_k\} = \operatorname{span}\{\vec{r}_0, A\vec{r}_0, \ldots, A^{k-1}\vec{r}_0\}$ for all $k$

# What We (Greedily) Want

1. Easy way to generate $n$ conjugate directions $\{\vec{v}_1, \ldots, \vec{v}_n\}$
2. $\operatorname{span}\{\vec{v}_1, \ldots, \vec{v}_k\} = \operatorname{span}\{\vec{r}_0, A\vec{r}_0, \ldots, A^{k-1}\vec{r}_0\}$ for all $k$

---

1. Converges in $n$ steps
2. Always better than gradient descent

# One Additional Observation

For $A$-conjugate search directions:

$$-\nabla f(\vec{x}_k) \perp \{\vec{v}_1, \ldots, \vec{v}_k\}$$

# One Additional Observation

For $A$-conjugate search directions:

$$-\nabla f(\vec{x}_k) \perp \{\vec{v}_1, \ldots, \vec{v}_k\}$$

$$\forall k \leq i, \vec{r}_i \cdot \vec{v}_k = 0$$

*Keep dot products straight!*

# Surprising Result

$$\langle \vec{v}_\ell, \vec{r}_k \rangle_A = 0, \text{ when } \ell < k.$$

# Surprising Result

$$\langle \vec{v}_\ell, \vec{r}_k \rangle_A = 0, \text{ when } \ell < k.$$

To generate $\vec{v}_k$ from $\vec{r}_{k-1}$, only need to project out $\vec{v}_{k-1}$!

# Projection Formula

$$\vec{v}_k = \vec{r}_{k-1} - \frac{\langle \vec{r}_{k-1}, \vec{v}_{k-1} \rangle_A}{\langle \vec{v}_{k-1}, \vec{v}_{k-1} \rangle_A} \vec{v}_{k-1}$$

# Conjugate Gradients Algorithm

1. Update search direction:
$$\vec{v}_k = \vec{r}_{k-1} - \frac{\langle \vec{r}_{k-1}, \vec{v}_{k-1} \rangle_A}{\langle \vec{v}_{k-1}, \vec{v}_{k-1} \rangle_A} \vec{v}_{k-1}$$
2. Line search: $\alpha_k = \frac{\vec{v}_k^\top \vec{r}_{k-1}}{\vec{v}_k^\top A \vec{v}_k}$
3. Update estimate: $\vec{x}_k = \vec{x}_{k-1} + \alpha_k \vec{v}_k$
4. Update residual: $\vec{r}_k = \vec{r}_{k-1} - \alpha_k A \vec{v}_k$

# Properties

- $\vec{x}_k$ optimal in subspace spanned by first $k$ directions $\vec{v}_i$

# Properties

- $\vec{x}_k$ optimal in subspace spanned by first $k$ directions $\vec{v}_i$
- $f(\vec{x}_k)$ upper-bounded by $f$ of $k$-th iterate of gradient descent

# Properties

- $\vec{x}_k$ optimal in subspace spanned by first $k$ directions $\vec{v}_i$
- $f(\vec{x}_k)$ upper-bounded by $f$ of $k$-th iterate of gradient descent
- Converges in $n$ steps

# Nicer Form

*See notes.*

- Update search direction: $\beta_k = \frac{\vec{r}_{k-1}^\top \vec{r}_{k-1}}{\vec{r}_{k-2}^\top \vec{r}_{k-2}}$
  $\vec{v}_k = \vec{r}_{k-1} + \beta_k \vec{v}_{k-1}$
- Line search: $\alpha_k = \frac{\vec{r}_{k-1}^\top \vec{r}_{k-1}}{\vec{v}_k^\top A \vec{v}_k}$
- Update estimate: $\vec{x}_k = \vec{x}_{k-1} + \alpha_k \vec{v}_k$
- Update residual: $\vec{r}_k = \vec{r}_{k-1} - \alpha_k A \vec{v}_k$

# Typical Stopping Conditions

*Don't run to completion!*

$$\frac{\|\vec{r}_k\|}{\|\vec{r}_0\|} < \varepsilon$$

# Convergence

$$\frac{f(\vec{x}_k) - f(\vec{x}^*)}{f(\vec{x}_0) - f(\vec{x}^*)} \leq 2\left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1}\right)^k$$

# Convergence

$$\frac{f(\vec{x}_k) - f(\vec{x}^*)}{f(\vec{x}_0) - f(\vec{x}^*)} \leq 2\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^k$$

*Still depends on condition number!*

# Observation

$$\operatorname{cond} A \neq \operatorname{cond} PA$$

*But we can solve*

$$PA\vec{x} = P\vec{b}$$

# Preconditioning

Solve $PA\vec{x} = P\vec{b}$ for $P \approx A^{-1}$.

# Two Issues

- $PA$ may not be symmetric or positive definite
- Need to find an "easy" $P$

# Symmetrization

$$P \text{ symmetric and positive definite} \implies P^{-1} = EE^\top.$$

# Symmetrization

$P$ symmetric and positive definite

$$\implies P^{-1} = EE^\top.$$

## Proposition

$$\operatorname{cond} PA = \operatorname{cond} E^{-1}AE^{-\top}.$$

# Symmetrization

$P$ symmetric and positive definite
$$\implies P^{-1} = EE^\top.$$

## Proposition

$$\operatorname{cond} PA = \operatorname{cond} E^{-1}AE^{-\top}.$$

1. Solve $E^{-1}AE^{-\top}\vec{y} = E^{-1}\vec{b}$
2. Solve $\vec{x} = E^{-\top}\vec{y}$

# Better-Conditioned CG

**Update search direction:** $\beta_k = \dfrac{\vec{r}_{k-1}^{\top}\vec{r}_{k-1}}{\vec{r}_{k-2}^{\top}\vec{r}_{k-2}}$

$\vec{v}_k = \vec{r}_{k-1} + \beta_k \vec{v}_{k-1}$

**Line search:** $\alpha_k = \dfrac{\vec{r}_{k-1}^{\top}\vec{r}_{k-1}}{\vec{v}_k^{\top} E^{-1} A E^{-\top} \vec{v}_k}$

**Update estimate:** $\vec{y}_k = \vec{y}_{k-1} + \alpha_k \vec{v}_k$

**Update residual:** $\vec{r}_k = \vec{r}_{k-1} - \alpha_k E^{-1} A E^{-\top} \vec{v}_k$

# Preconditioned CG

$$\textbf{Update search direction: } \beta_k = \frac{\tilde{r}_{k-1}^\top P \tilde{r}_{k-1}}{\tilde{r}_{k-2}^\top P \tilde{r}_{k-2}}$$

$$\tilde{v}_k = P\tilde{r}_{k-1} + \beta_k \tilde{v}_{k-1}$$

$$\textbf{Line search: } \alpha_k = \frac{\vec{r}_{k-1}^\top P \vec{r}_{k-1}}{\tilde{v}_k^\top A \tilde{v}_k}$$

$$\textbf{Update estimate: } \vec{x}_k = \vec{x}_{k-1} + \alpha_k \tilde{v}_k$$

$$\textbf{Update residual: } \tilde{r}_k = \tilde{r}_{k-1} - \alpha_k A \tilde{v}_k$$

# Common Preconditioners

- Diagonal ("Jacobi") $A \approx D$
- Sparse approximate inverse
- Incomplete Cholesky $A \approx L_* L_*^\top$
- Domain decomposition

# Other Iterative Schemes I

- Splitting: $A = M - N \implies M\vec{x} = N\vec{x} + \vec{b}$
- Conjugate gradient normal equation residual (CGNR): $A^\top A\vec{x} = A^\top \vec{b}$
- Conjugate gradient normal equation error (CGNE): $AA^\top \vec{y} = \vec{b}$; $\vec{x} = A^\top \vec{y}$
- MINRES, SYMLQ: $g(\vec{x}) \equiv \|\vec{b} - A\vec{x}\|_2^2$ for symmetric $A$
- LSQR, LSMR: Normal equations, same $g$

# Other Iterative Schemes II

- GMRES, QMR, BiCG, CGS, BiCGStab: Any invertible $A$
- Fletcher-Reeves, Polak-Ribière: Nonlinear problems; replace residual with $-\nabla f$ and add back line search

Next