

ITERATIVE LINEAR SYSTEM SOLVERS FOR LARGE, SPARSE MATRICES

ITERATIVE METHODS

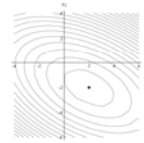
Mostly for SPD systems

- conjugate gradient and its variants

$$Ax = b$$

$$f(x) = \frac{1}{2}x^T Ax - b^T x$$

$$\nabla f = Ax - b = 0$$



- minimize through successive line searches

STEEPEST DESCENT

Decrease function

- negative of gradient

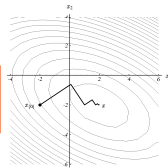
$$-\nabla f(x^{(i)}) = b - Ax^{(i)} = r^{(i)}$$

- how much?

$$\frac{\partial f}{\partial \alpha} \Big|_{x^{(i+1)}} = 0 = -r^{(i+1)T} r^{(i)}$$

$$f(x^{(i)} + \alpha r^{(i)})$$

$$x^{(i+1)}$$



STEEPEST DESCENT

How much?

$$r^{(i+1)T} r^{(i)} = 0$$

$$\leadsto (b - Ax^{(i+1)})^T r^{(i)} = 0$$

$$\leadsto (b - Ax^{(i)})^T r^{(i)} = \alpha^{(i)} r^{(i)T} A r^{(i)}$$

$$\leadsto \alpha^{(i)} = \frac{r^{(i)T} r^{(i)}}{r^{(i)T} A r^{(i)}}$$

STEEPEST DESCENT

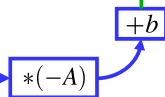
Algorithm

$$r^{(i)} = b - Ax^{(i)}$$

$$\alpha^{(i)} = \frac{r^{(i)T} r^{(i)}}{r^{(i)T} A r^{(i)}}$$

$$r^{(i+1)} = r^{(i)} - \alpha^{(i)} A r^{(i)}$$

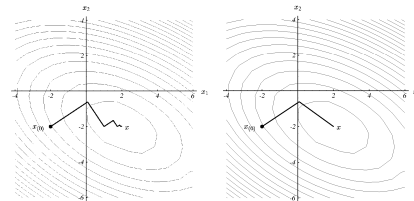
$$x^{(i+1)} = x^{(i)} + \alpha^{(i)} r^{(i)}$$



LET'S DO BETTER

Conjugate directions

- don't undo earlier gains



CONJUGATE GRADIENT

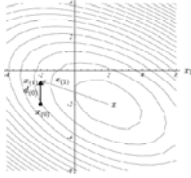
$$d_i^T A d_j = 0$$

Directions A-orthogonal to error

$$x^{(i+1)} = x^{(i)} + \alpha^{(i)} d^{(i)}$$

$$\left. \frac{\partial f}{\partial \alpha} \right|_{x^{(i+1)}} = 0 = -r^{(i+1)T} d^{(i)}$$

$$\alpha^{(i)} = \frac{d^{(i)T} r^{(i)}}{d^{(i)T} A d^{(i)}}$$



CONJUGATE GRADIENT

Next direction?

- orthogonal to all others
- easy update:

$$d^{(i+1)} = r^{(i+1)} + \frac{r^{(i+1)T} r^{(i+1)}}{r^{(i)T} r^{(i)}} d^{(i)}$$

ALGORITHM

What happened to d ?
(see section 7.3 of Shewchuk)

$$d^{(0)} = r^{(0)} = b - Ax^{(0)}$$

$$\alpha^{(i)} = r^{(i)T} r^{(i)} / d^{(i)T} A d^{(i)}$$

$$x^{(i+1)} = x^{(i)} + \alpha^{(i)} d^{(i)}$$

$$r^{(i+1)} = r^{(i)} - \alpha^{(i)} A d^{(i)}$$

$$\beta^{(i+1)} = r^{(i+1)T} r^{(i+1)} / r^{(i)T} r^{(i)}$$

$$d^{(i+1)} = r^{(i+1)} + \beta^{(i+1)} d^{(i)}$$

Sweeps
matrix multiply
dot product acc.
update
dot product acc.
update

ITERATIVE METHODS

What you'll need

- you supply Ax
- given topological datastructure
 - implement matrix multiplies
- stopping criterion: relative residual

TYPICAL IMPLEMENTATION

What lives where?

- degrees of freedom at vertices
 - need additional variables for solver
- matrix entries on edges

Matrix multiplies

- iterate over mesh!
- mesh is sparse matrix data structure

ITERATIVE METHODS

Other issues

- number of iterations proportional to κ (steepest descent), $\sqrt{\kappa}$ (CG)
- preconditioning
 - use approximate inverse
 - diagonal (Jacobi)
 - incomplete Cholesky, hierarchical,...

$$\kappa(M^{-1}A) = 1$$

OTHER SYSTEM TYPES

SPD

- conjugate gradients
- Shewchuk, “CG w/o the Agony”

non-S PD & non-S non-PD

- bi-conjugate gradients
- see Numerical Recipes
 - <http://www.nr.com/> (chapter 2.7)