Algebraic Multigrid

Michael Wathen

UBC Computer Science

8^{th} April 2016

Motivation

- Want model real world applications
- Stokes, Navier-Stokes, Maxwell's, Cahn-Hilliard, MHD, etc ...
- Involves solving linear systems based on FEM, FV and FD

Problem

- Consider solving the $n \times n$ system:

$$Ax = b$$

- How do we solves these systems optimally?

Problem

- Consider solving the $n \times n$ system:

$$Ax = b$$

- How do we solves these systems optimally?
- We have two options Direct or Iterative
- Direct: flops$\approx \mathcal{O}(n^3)$, large memory cost
- Iterative: flop$\approx \mathcal{O}(n)$???, low memory cost

Generic iterative method form

- Most iterative methods have the following form, where r_k is the residual at iteration k

$$x_{k+1} = x_k + M^{-1} r_k$$

- Let $e_k = x - x_k$ be the error, and note that $r_k = Ae_k$
- The error propagation for the iterative method is

$$e_{k+1} = (I - M^{-1}A)e_k$$

Why parallel?

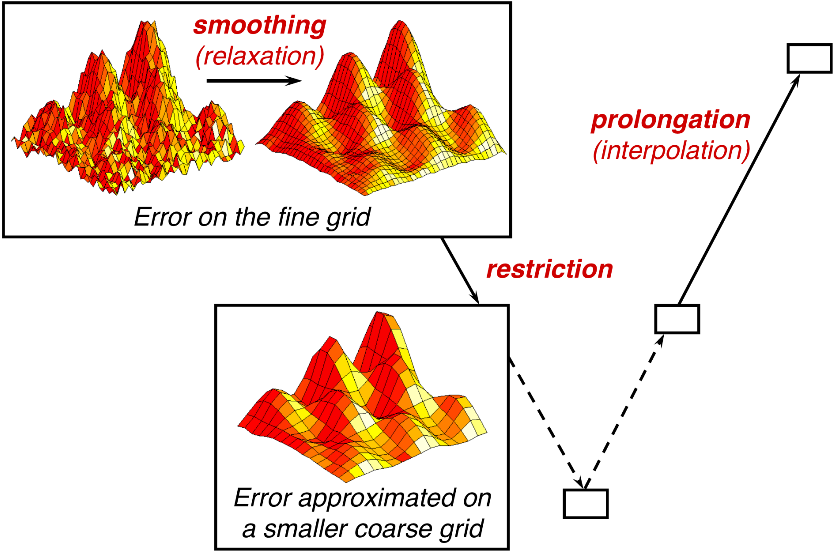
Why parallel?

Basic answer: real world problems are huge!!!

Why parallel?

- Modeling earths mantle - grid cells about 1km across
- Geophysical surveys - seismic, EM, DC
- Weather forecasting
- Multi-phase porous media flow
- ...

Geometric multigrid



Geometric multigrid (continued)

General multigrid algorithm

Step 1: smooth error (residual)

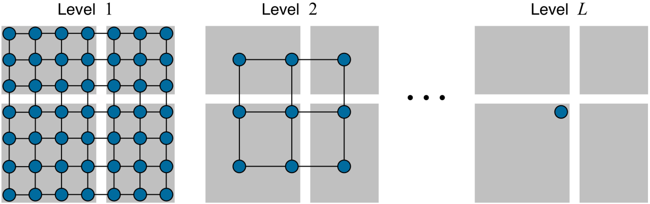
Step 2: restrict to coarse grid

Step 3: Solve on coarse grid

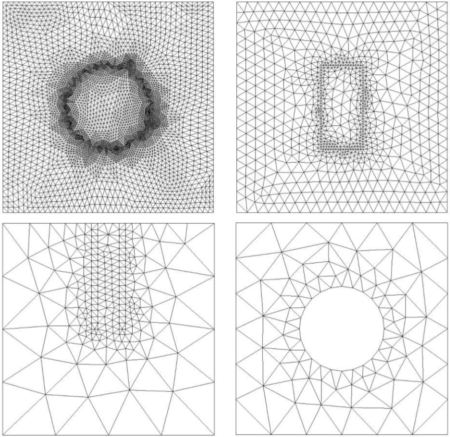
Step 4: prolongate and correct

Parallel geometric multigrid

Basic idea: split up discretisation grid



Unstructured grids



Algebraic multigrid

How do we do this in without defining a sequence of grids

- Algebraic smoothness
- Algebraic restriction (how to define a sequence of grids)

Algebraic multigrid: smoothness

Smoothness:

$$e^{\mathrm{T}}Ae = \lambda \ll 1$$

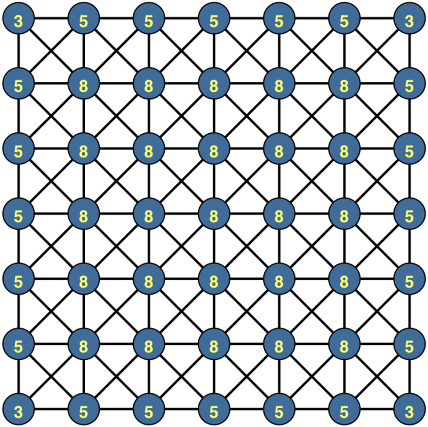
$$e^{\mathrm{T}}Ae = \sum_{i<j}(-a_{ij})(e_i - e_j)^2 \ll 1$$

Algebraic multigrid: grid selection

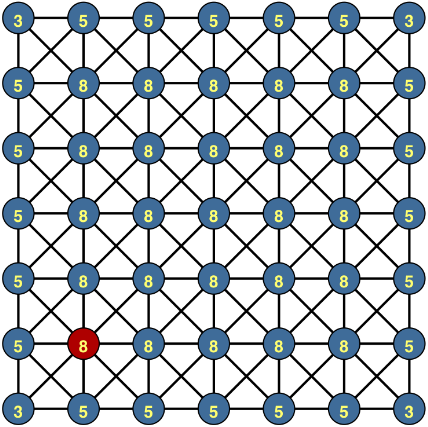
Strength of Connection

$$-a_{ij} \geq \theta \max_{k \neq i}\{-a_{ik}\} \quad \text{where } \theta \in (0, 1]$$

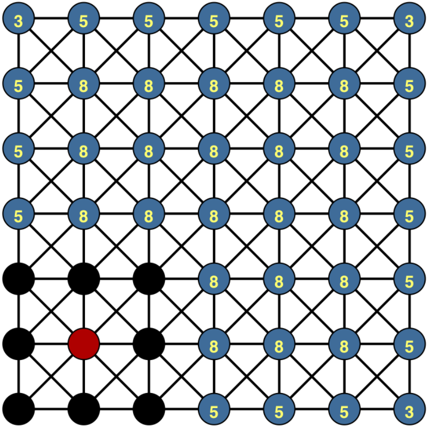
1. Define strength matrix A_s
2. Choose set of fine points based on A_s
3. Choose extra points to satisfy interpolation requirements



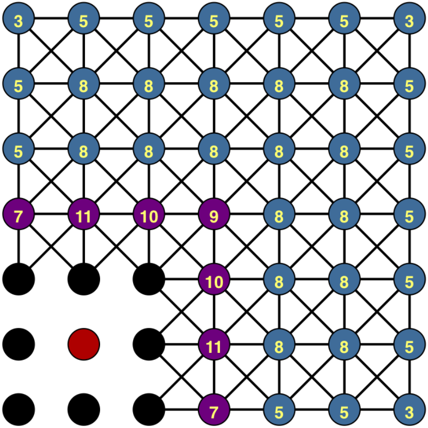
- select C-pt with maximal measure
- select neighbours as F-pts
- update measures of F-pt neighbours



- select C-pt with maximal measure
- select neighbours as F-pts
- update measures of F-pt neighbours



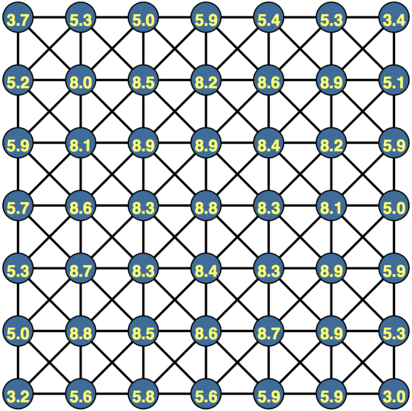
- select C-pt with maximal measure
- select neighbours as F-pts
- update measures of F-pt neighbours



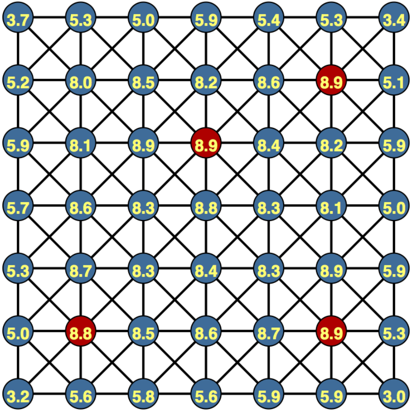
- select C-pt with maximal measure
- select neighbours as F-pts
- update measures of F-pt neighbours

Parallel algebraic multigrid

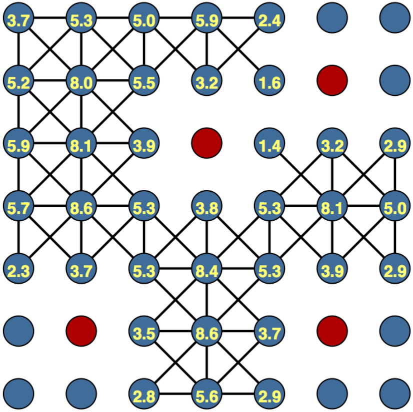
- This process is inherently sequential.....
- Parallel algorithms:
  - CLJP (Cleary-Luby-Jones-Plassmann) – one-pass approach with random numbers to get concurrency
  - Falgout – C-AMG on processor interior, then CLJP to finish
  - PMIS – CLJP without the 'C'; parallel version of C-AMG first pass
  - HMIS – C-AMG on processor interior, then PMIS to finish
  - ...



- select C-pts with maximal measure locally
- remove neighbour edges
- update neighbour measures



- select C-pts with maximal measure locally
- remove neighbour edges
- update neighbour measures



- select C-pts with maximal measure locally
- remove neighbour edges
- update neighbour measures

Summary

- Multigrid methods are optimal and have good scaling potential
- GM: relies on a sequence of predetermined geometric grids
- AMG: uses matrix coefficients to determine a sequence of "grids"
- P-AMG: additional restrictions on AMG algorithmic development

Summary

- Multigrid methods are optimal and have good scaling potential
- GM: relies on a sequence of predetermined geometric grids
- AMG: uses matrix coefficients to determine a sequence of "grids"
- P-AMG: additional restrictions on AMG algorithmic development

THANK YOU