Subgrid Upscaling Preconditioners for Darcy Flow

James M. Rath

Center for Subsurface Modeling / ICES

Committee:
Todd Arbogast (advisor)
Steve Bryant
Clint Dawson
Robert van de Geijn
Jack Xin

Outline:

- ***The Problem:***
  - Modeling subsurface flows
- ***Subgrid Upscaling***
- ***Two-level Multigrid-like Scheme***
- ***Change the Matrix, not the RHS***
- ***Applications***

Small piece of the model:

Darcy's law

$$\mathbf{u} = -\frac{\kappa}{\mu}(\nabla p - \rho \mathbf{g})$$

and the continuity equation

$$\nabla \cdot \mathbf{u} = q$$

for single-phase steady-state flow.

Approximate using a cell-centered finite difference method (CCFD) for the pressures.

Motivation:

- **A single multiphase, time-dependent simulation requires repeated computations of approximations like these.**
- **And many more for statistical sampling and/or optimization studies.**

Motivation:

- A single multiphase, time-dependent simulation requires repeated computations of approximations like these.
- And many more for statistical sampling and/or optimization studies.

Goals:

- Fast, accurate approximations (in as much detail as possible).
- Iterative solution methods.
- Subgrid upscaling used for accelerated convergence.

Difficulties:

- Geostatistical description of permeability has high-resolution.
  Details necessary for correct flow prediction.
- Permeability is heterogeneous.
  Big jumps occur on short spatial scales.
- Together produce an ill-conditioned
  and computationally expensive problem.

Outline:

- *The Problem*
- ***Subgrid Upscaling***
  - Mixed Variational Form and Its Approximation
  - Subgrid Upscaling and Its Approximation
  - Examples
- *Two-level Multigrid-like Scheme*
- *Change the Matrix, not the RHS*
- *Applications*

Mixed Variational Form:

Let

$$W = L^2(\Omega)/\mathbb{R},$$
$$\mathbf{V} = \{\mathbf{v} \in H(\mathrm{div}, \Omega) \mid \mathbf{v} \cdot \nu = 0 \text{ on } \partial\Omega\}.$$

Find $p \in W$ and $\mathbf{u} \in \mathbf{V}$ such that

$$(\nabla \cdot \mathbf{u}, w) = (q, w) \qquad \forall w \in W,$$
$$(\mu\kappa^{-1}\mathbf{u}, \mathbf{v}) - (p, \nabla \cdot \mathbf{v}) = (\rho\mathbf{g}, \mathbf{v}) \qquad \forall \mathbf{v} \in \mathbf{V}.$$

Finite Element Approximation:

Fine-scale problem:

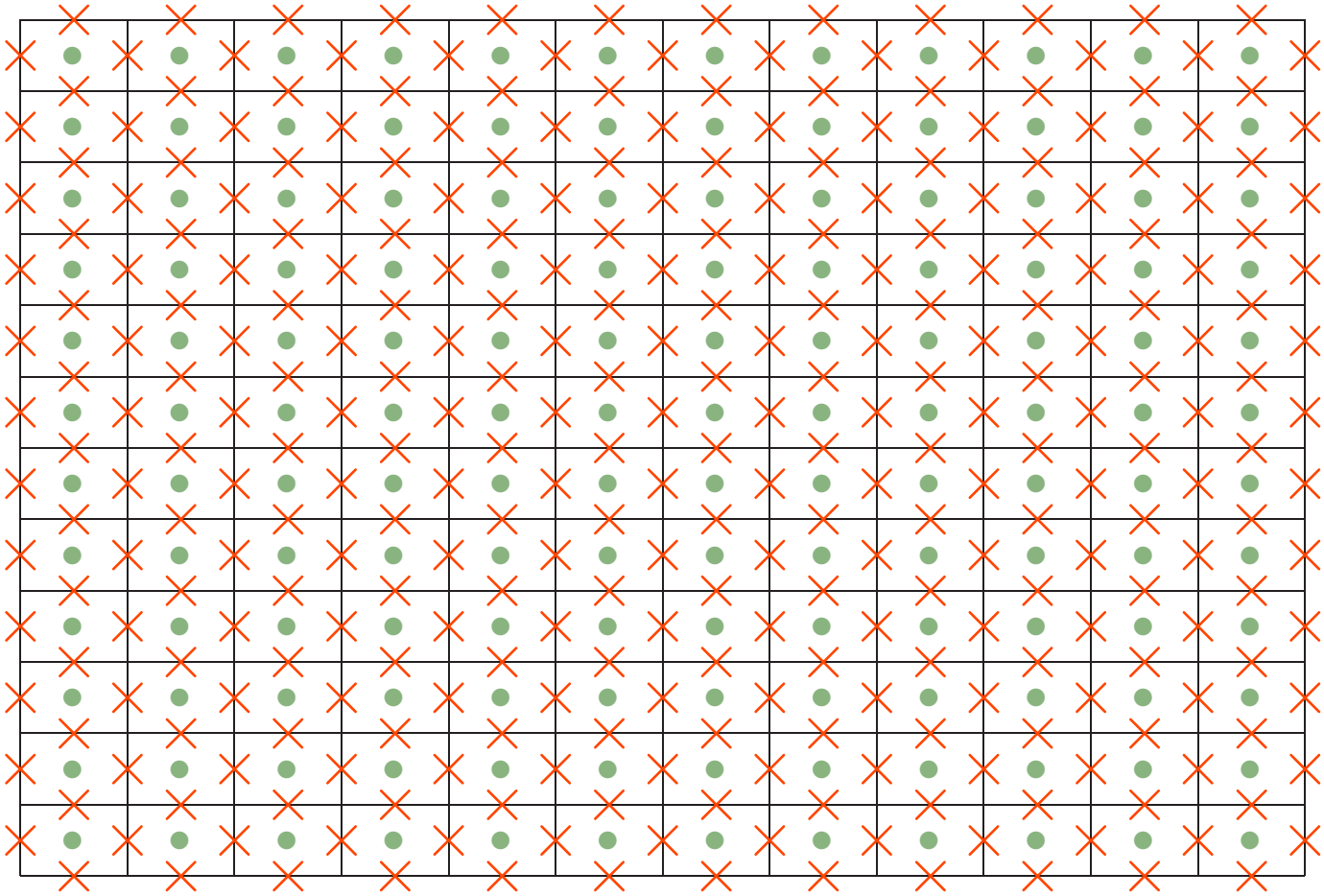
- **Grid with spacing h,**
- **κ piecewise constant,**
- **and lowest-order Raviart-Thomas-Nedelec (RT0) elements $W_h \times \mathbf{V}_h \subset W \times \mathbf{V}$.**

Find $p_h \in W_h$ and $\mathbf{u}_h \in \mathbf{V}_h$ such that

$$(\nabla \cdot \mathbf{u}_h, w_h) = (q, w_h) \qquad \forall w_h \in W_h,$$
$$(\mu\kappa^{-1}\mathbf{u}_h, \mathbf{v}_h) - (p_h, \nabla \cdot \mathbf{v}_h) = (\rho\mathbf{g}, \mathbf{v}_h) \qquad \forall \mathbf{v}_h \in \mathbf{V}_h.$$

Using quadrature — the trapezoidal rule — to compute $(\mu\kappa^{-1}\mathbf{u}_h, \mathbf{v}_h)$ is equivalent to solving the original problem using CCFD.

Degrees of freedom for 2-D RT0 elements on a fine rectangular grid:



× Velocity

• Pressure

Coarse-Scale Approximation:

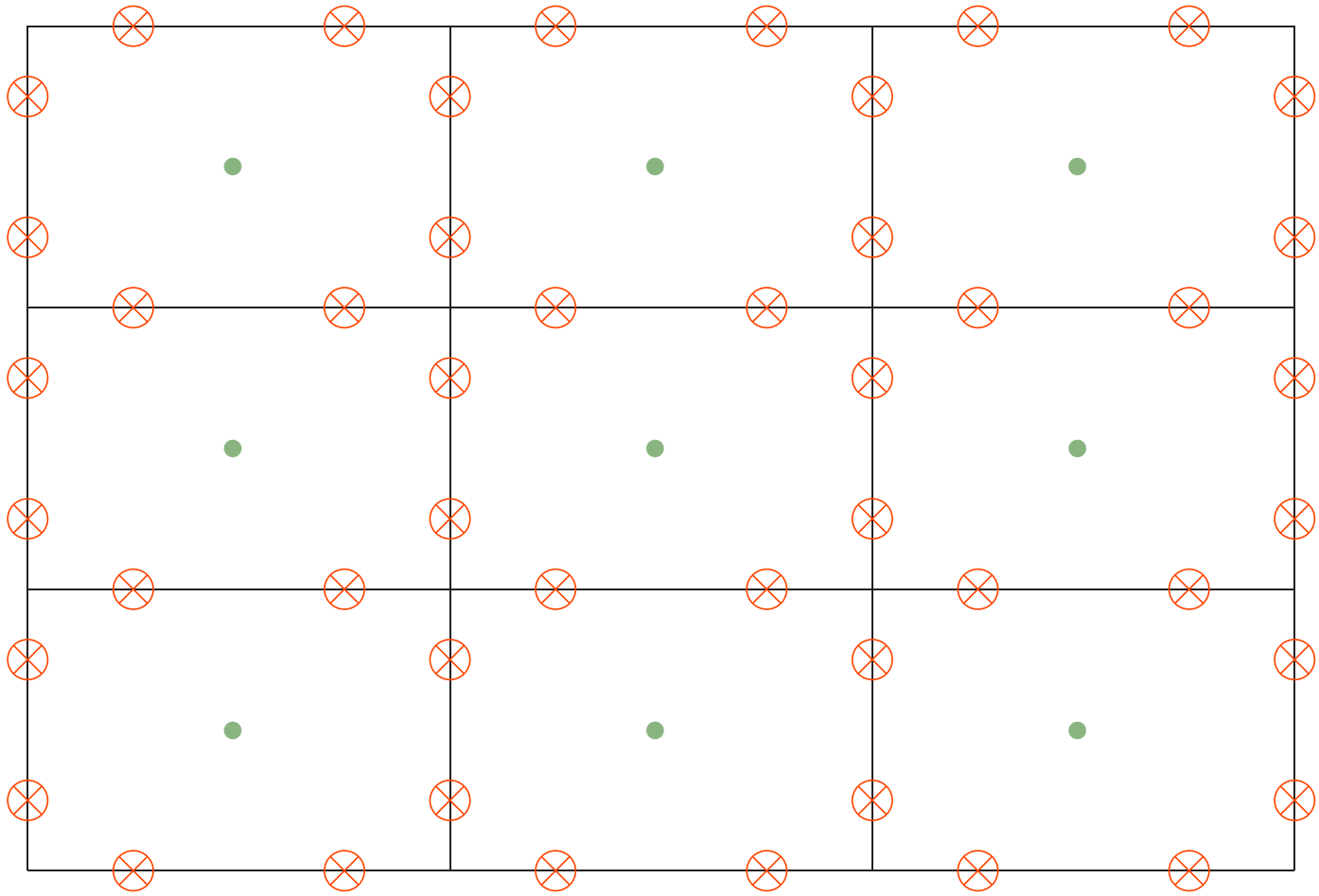
Coarse-scale problem:

- **Grid with spacing $H = Ch$,**
- **κ piecewise constant (with h resolution),**
- **and RT0 or lowest-order Brezzi-Douglas-Duràn-Fortin (BDDF1) elements $W_H \times \mathbf{V}_H \subset W \times \mathbf{V}$.**

Find $p_H \in W_H$ and $\mathbf{u}_H \in \mathbf{V}_H$ such that

$$(\nabla \cdot \mathbf{u}_H, w_H) = (q, w_H) \qquad \forall w_H \in W_H,$$
$$(\mu\kappa^{-1}\mathbf{u}_H, \mathbf{v}_H) - (p_H, \nabla \cdot \mathbf{v}_H) = (\rho\mathbf{g}, \mathbf{v}_H) \qquad \forall \mathbf{v}_H \in \mathbf{V}_H.$$

Degrees of freedom for 2-D BDM1 elements on a coarse rectangular grid:



⊗ Velocity (linear)

• Pressure

The Upscaling Technique:

Decompose W and $\mathbf{V}$ (using a chosen coarse grid) so that:

(1) no information is lost, $W = W_c \oplus \delta W$, $\mathbf{V} = \mathbf{V}_c \oplus \delta \mathbf{V}$;

(2) mass is conserved, $\nabla \cdot \mathbf{V}_c = W_c$, $\nabla \cdot \delta \mathbf{V} = \delta W$;

(3) fine-scale velocities are locally supported over the coarse grid, $\delta \mathbf{V} \cdot \nu = 0$ on the boundary of each coarse cell.

- Disconnect the subgrid problems from one another — but not from the coarse grid problem.
- Use Green's or influence function approach for forward elimination of subgrid problems into coarse problem.
- Follow with back substitution to recover the subgrid solutions.
- Recombine to obtain fine-scale solution.

Upscaling Approximation:

Approximate $\delta W \times \delta \mathbf{V}$ by RT0, $\delta W_h \times \delta \mathbf{V}_h$, and for $W_c \times \mathbf{V}_c$ use BDDF1, $W_H \times \mathbf{V}_H$.

Let $W_{H,h} = W_H \oplus \delta W_h$ and $\mathbf{V}_{H,h} = \mathbf{V}_H \oplus \delta \mathbf{V}_h$.

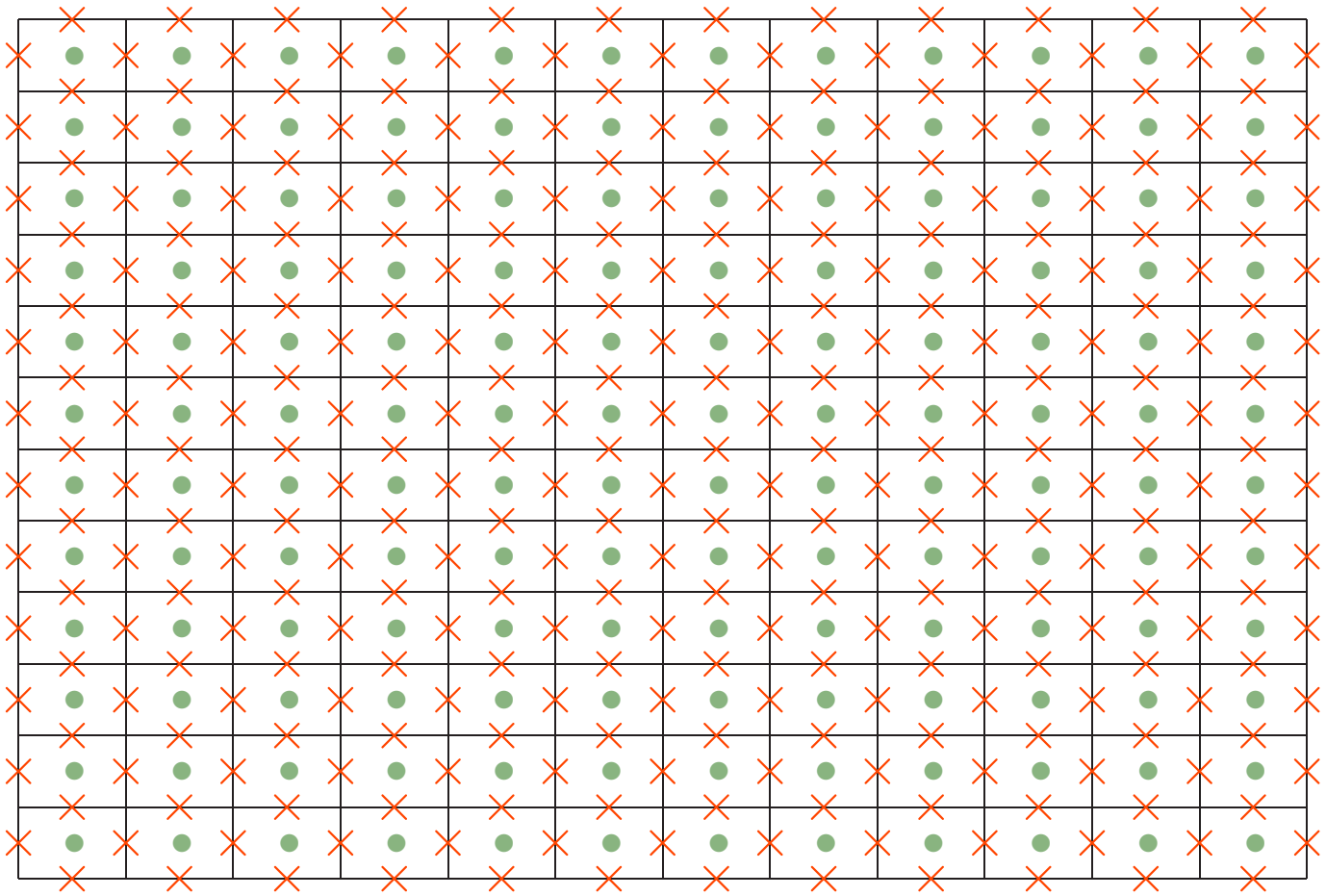
W_H**: piecewise constant on coarse cells**

δW_h**: piecewise constant on fine cells, with zero average over each coarse cell**

$\mathbf{V}_H$**: normal components continuous across coarse faces, and linear variation along coarse faces**

$\delta \mathbf{V}_h$**: normal components continuous across fine faces, and constant along those faces; normal components across coarse faces are zero**

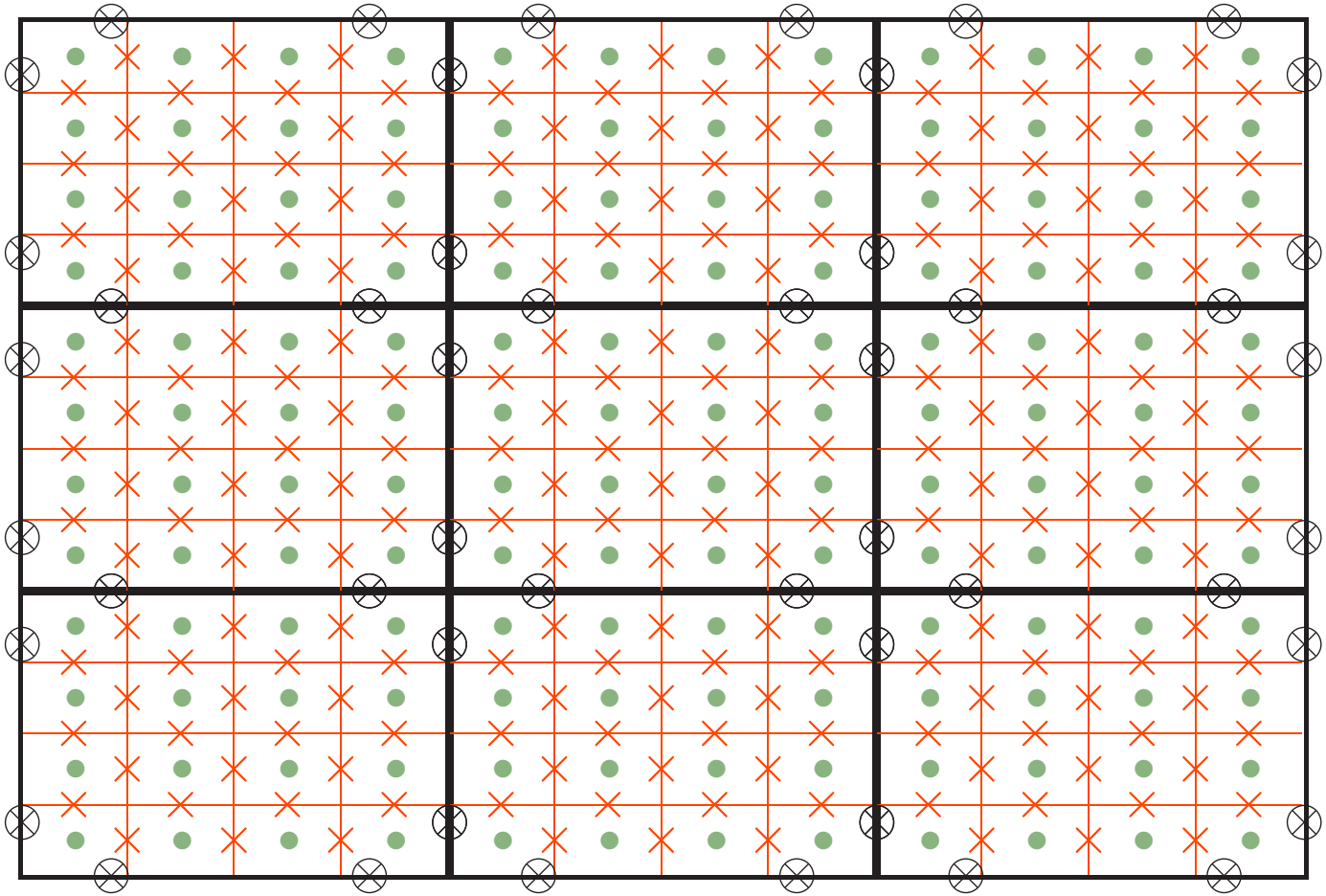
Degrees of freedom for 2-D RT0 elements on a fine rectangular grid:



× Velocity

• Pressure

Degrees of freedom for 2-D RT0/BDDF1 elements on an "upscaled" rectangular grid:



⊗ Coarse velocity (linear) × Subgrid velocity

• Pressure

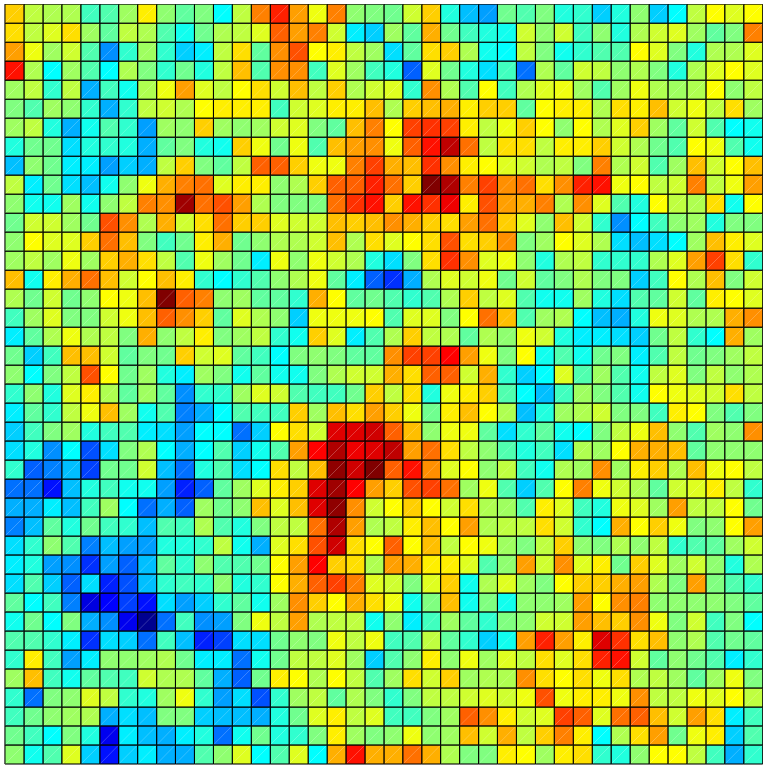
Advantages:

- Subgrid problems are independent.
  Solve in parallel with no communication between coarse cells.
- Many fewer degrees of freedom in subgrid problems than the whole (by about a factor of the number of coarse grid cells).
- Many fewer degrees of freedom in coarse problem than the whole (by about a factor of the number of fine cells per coarse cell).
- When H is a small multiple of h, nearly all work in solving the upscaled problem is in solving the coarse problem.
- The "closure" assumption is that coarse scale functions approximate all flux across coarse element faces.
- Examples in this section use a higher order coarse basis (BDDF1 vs. RT0) to compensate for this grainy restriction.

Some Practical Results:

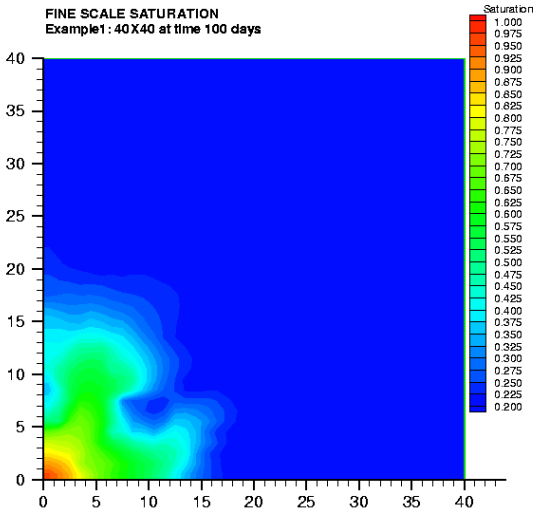
Quarter Five-spot Oil Reservoir Water Flood

Logarithm of the Permeability Field

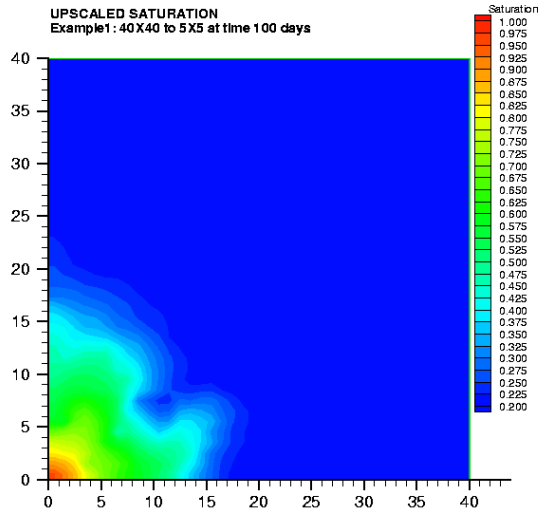


Fine 40×40

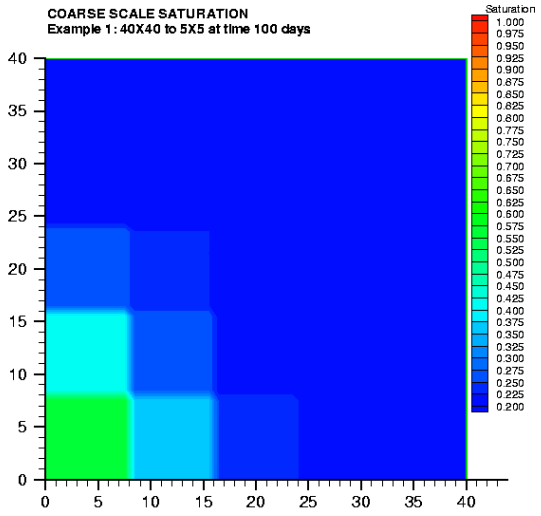
Water saturation contours at 100 days



Fine 40×40

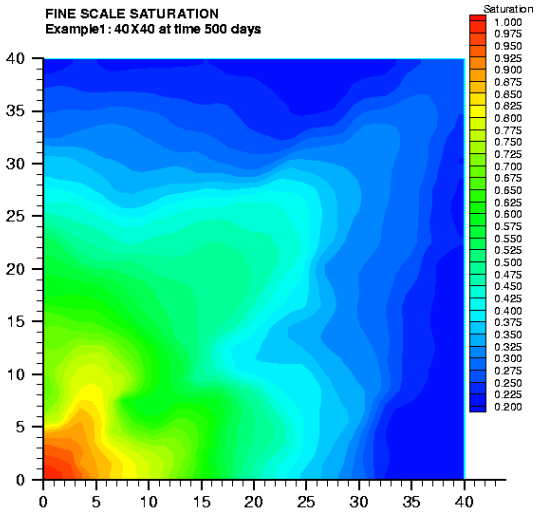


Fully upscaled to 5×5

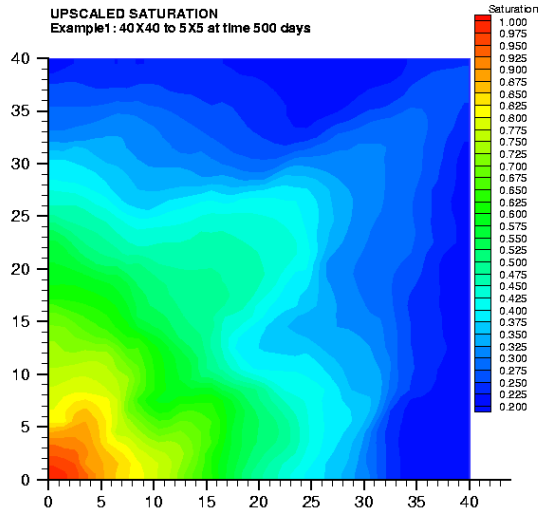


Coarse 5×5

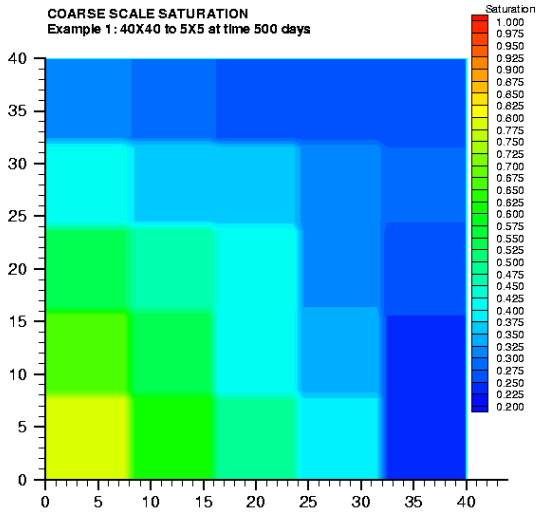
Water saturation contours at 500 days



Fine 40×40



Fully upscaled to 5×5



Coarse 5×5

Outline:

- *The Problem*
- *Subgrid Upscaling*
- ***Two-level Multigrid-like Scheme***
  - Algorithm Description
  - Examples
  - Further Study
- *Change the Matrix, not the RHS*
- *Applications*

Idea:

Use a two-level multigrid-like scheme
to solve a fine-scale problem.

Multigrid (MG) uses a fine approximation as one level and a coarse approximation as another.

Replace the coarse approximation with the subgrid upscaling approximation.

Upscaling retains more of the fine-scale information. It's a better approximation for about the same computational cost.

An overall faster scheme results.

Notation for Two-Level Scheme:

A: **matrix from pressure CCFD**

$\mathbf{x}$: **pressure unknowns**

$\mathbf{b}$: **information from sources, boundary terms, and gravity**

Then the problem is to solve the linear system $A\mathbf{x} = \mathbf{b}$**.**

$\mathbf{x}_i$: i**th guess at pressure unknowns**

$\mathbf{r}_i$: i**th-stage residual** $\mathbf{b} - A\mathbf{x}_i$

$\mathbf{c}_i$: i**th-stage smoothing corrector**

B_{J}: $\mathrm{diag}(A)$ **for Jacobi smoothing step**

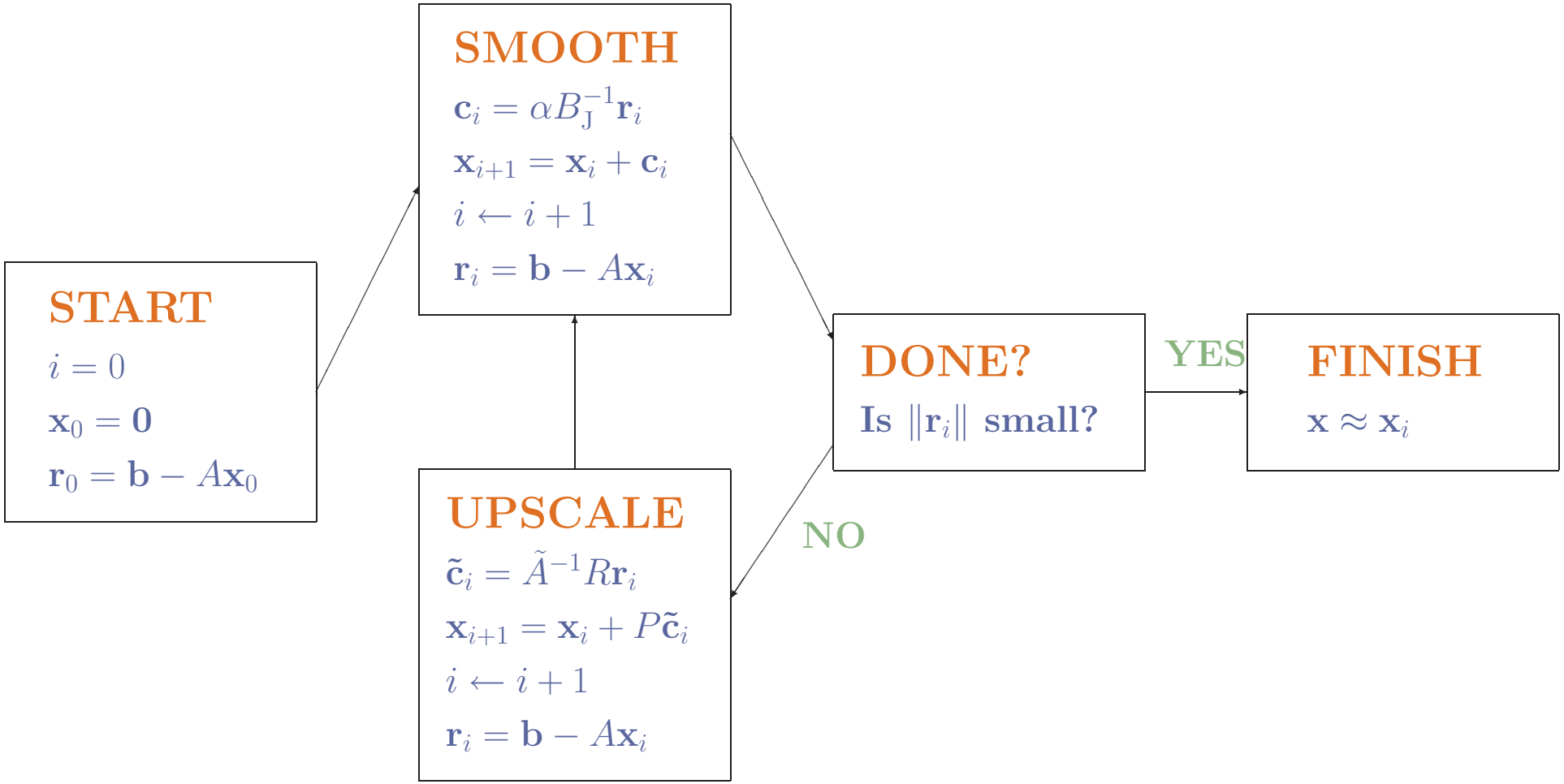
$\tilde{A}$: **matrix for corresponding upscaling MFEM problem**

$\tilde{\mathbf{c}}_i$: i**th-stage upscaling corrector (pressure and velocity unknowns)**

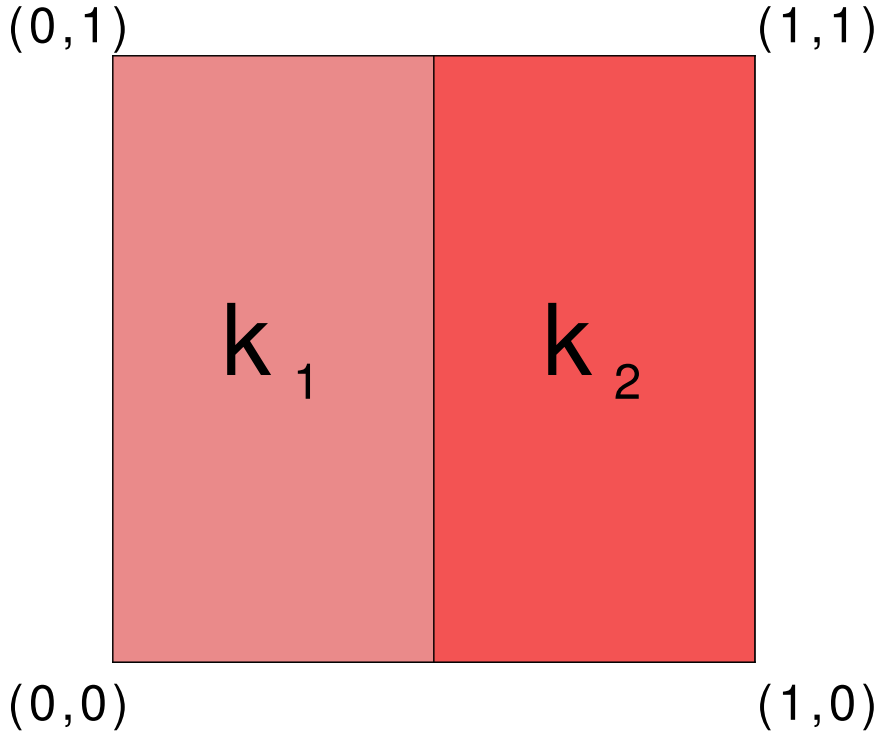
R: **restriction operator** $R : W_h \to W_{H,h} \times \mathbf{0}$

P: **prolongation operator** $P : W_{H,h} \times \mathbf{V}_{H,h} \to W_h$

Two-Level Schematic:



A Simple Test Case:

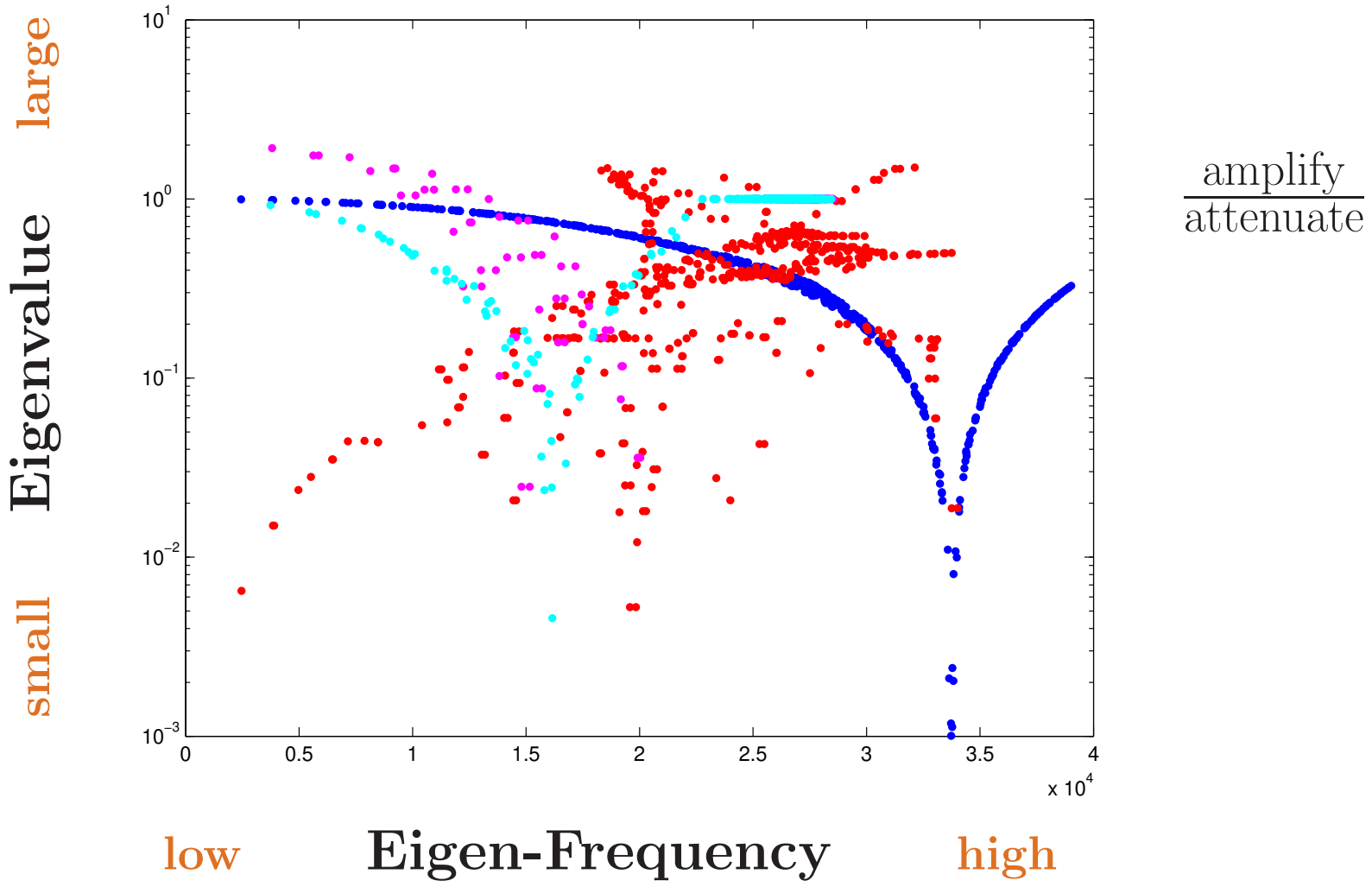


$k_1 = 1 \qquad k_2 > 0$

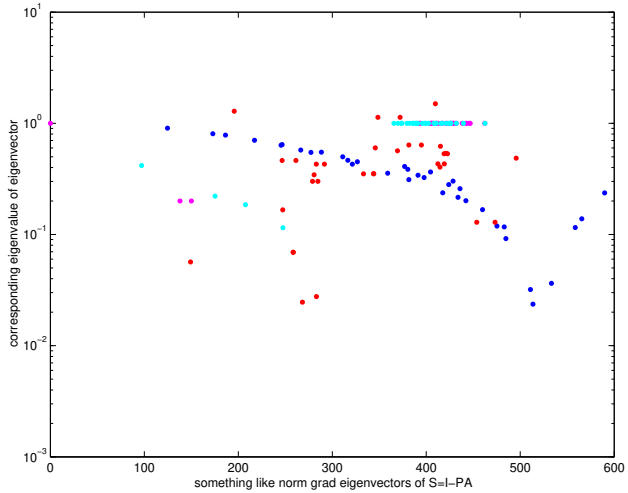
A Permeability Field

A "Typical" Result:

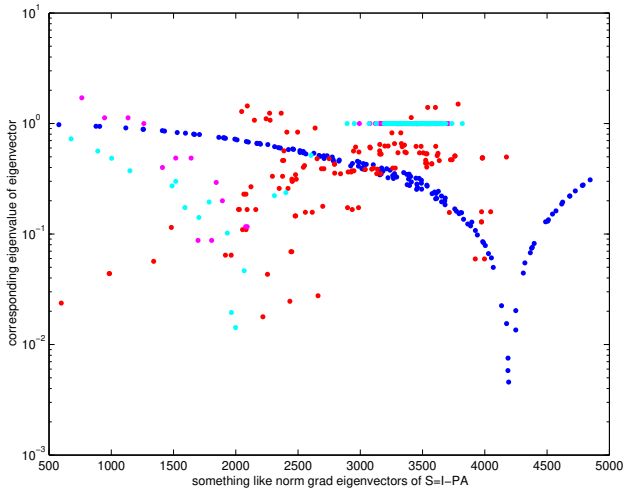
Eigenstructure of Smoother and Upscaler



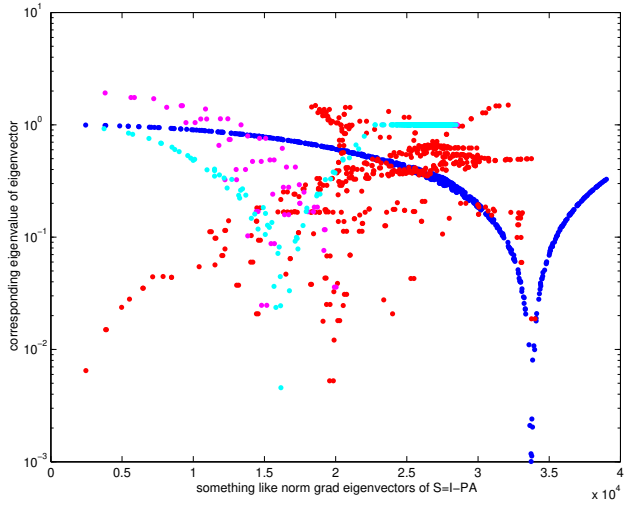
Vary h ; Fix $H/h = 3$; Fix $k_2 = 0.1$



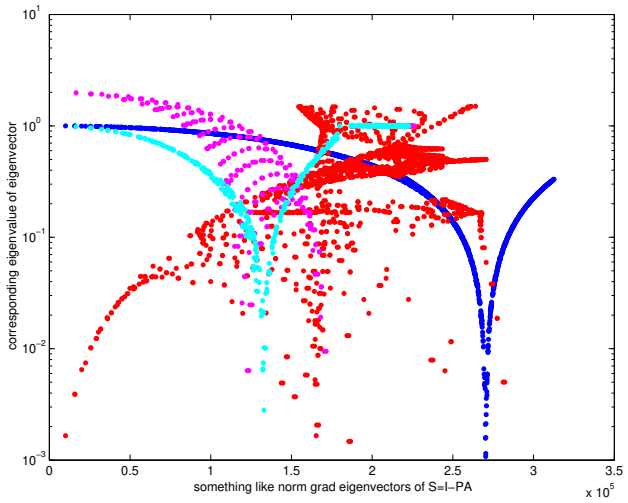
6×6



12×12

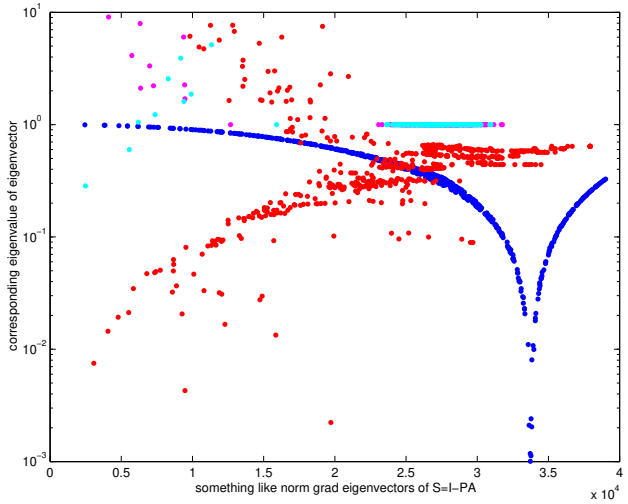


24×24

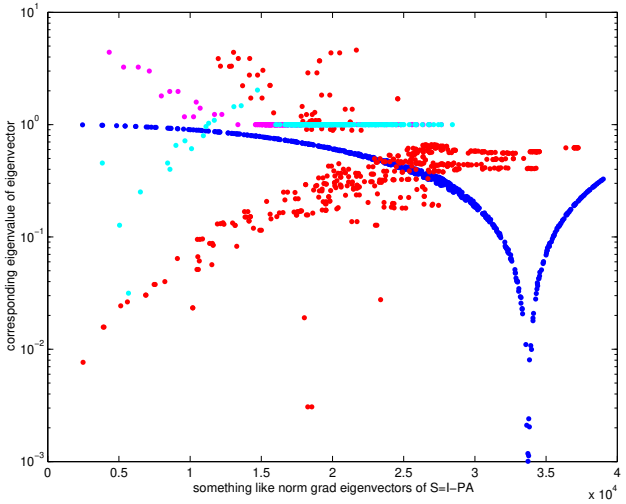


48×48

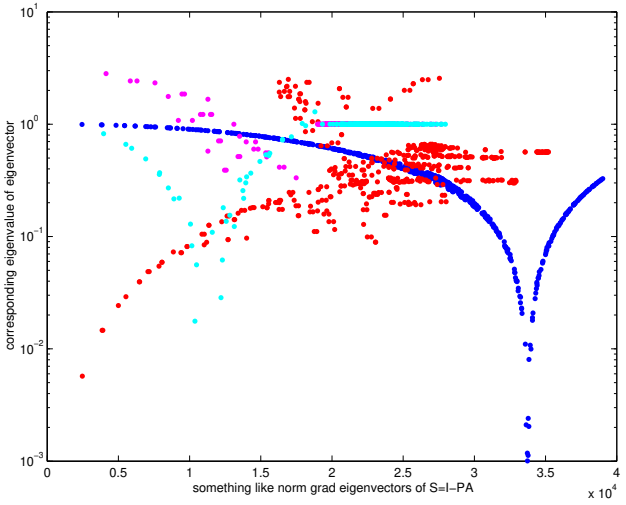
Fix $h = 1/24$ **; Vary** H/h **; Fix** $k_2 = 0.1$



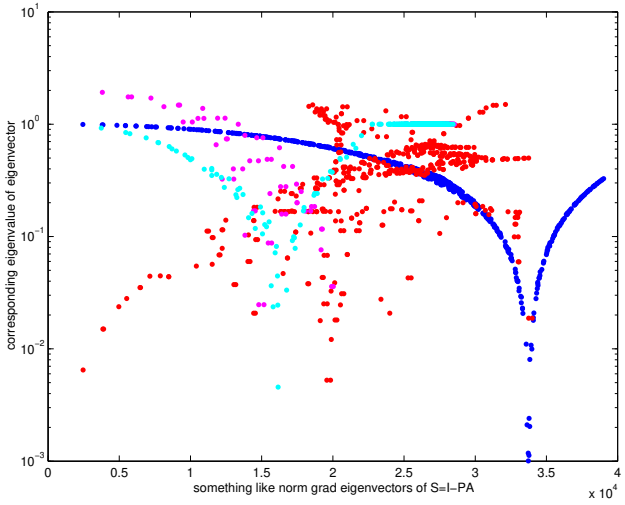
$H/h = 8$



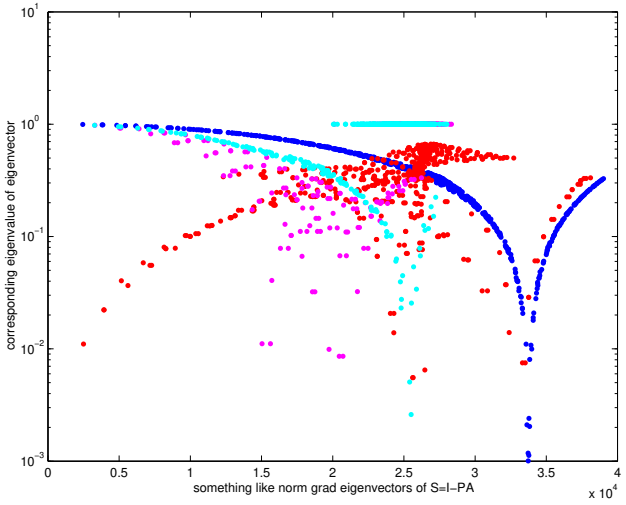
$H/h = 6$



$H/h = 4$



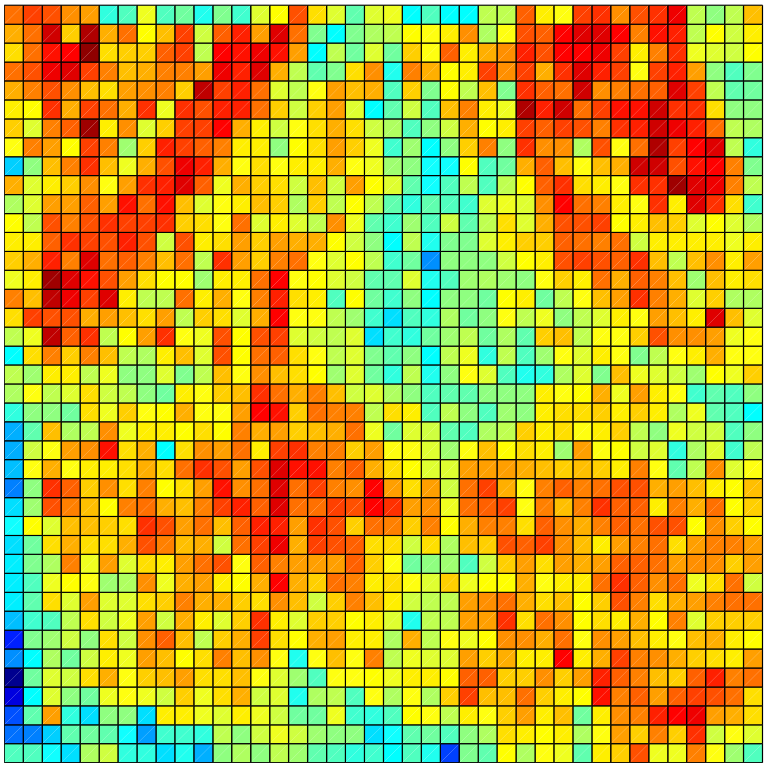
$H/h = 3$



$H/h = 2$

Two Other Permeabililty Fields:

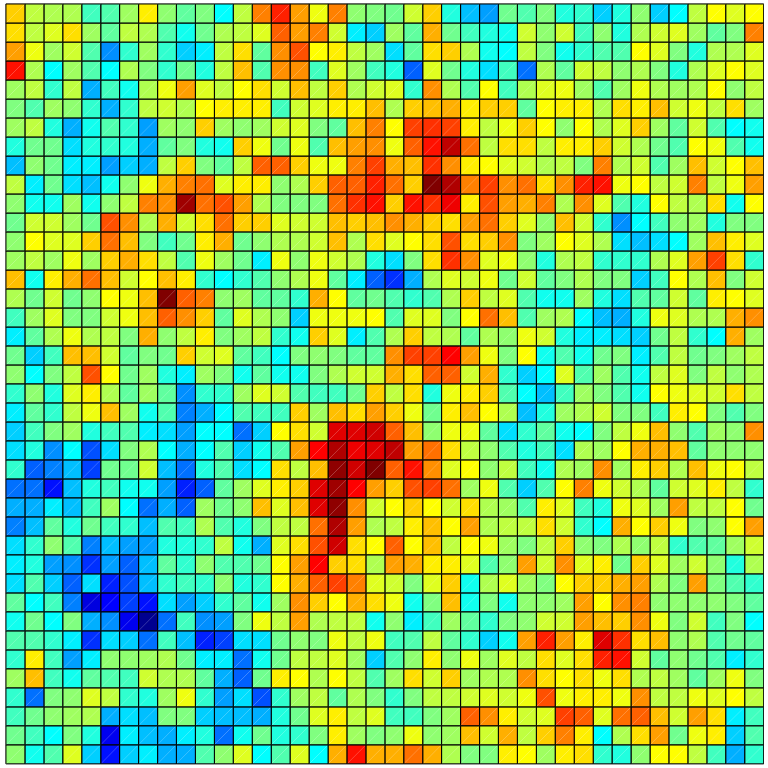
"Medium"



log10min = -0.2916

log10max = 0.1477

"Hard"

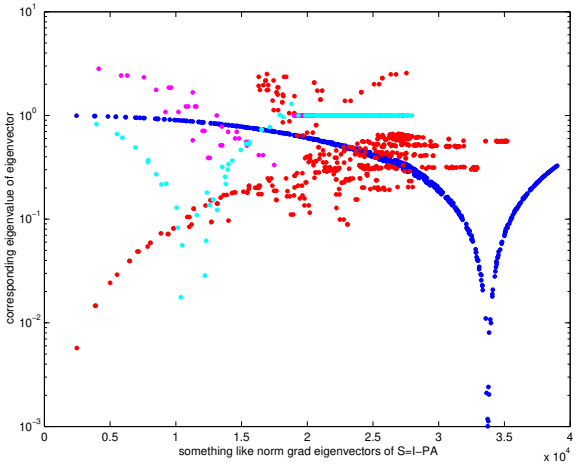


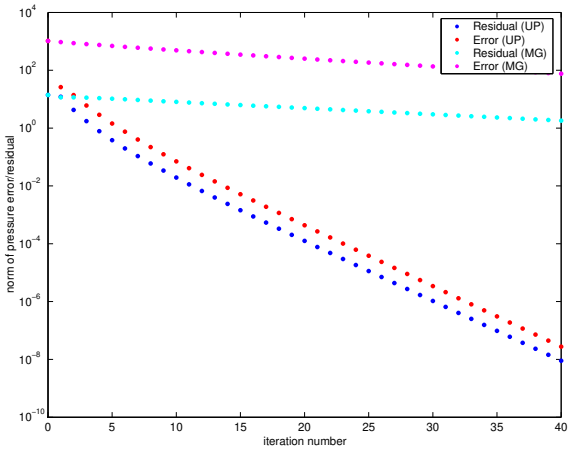
log10min = -0.6909

log10max = 3.5230

Sample Convergence Histories:

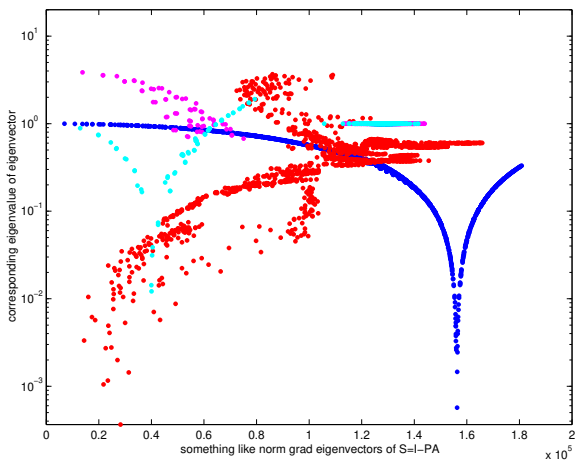
Simple

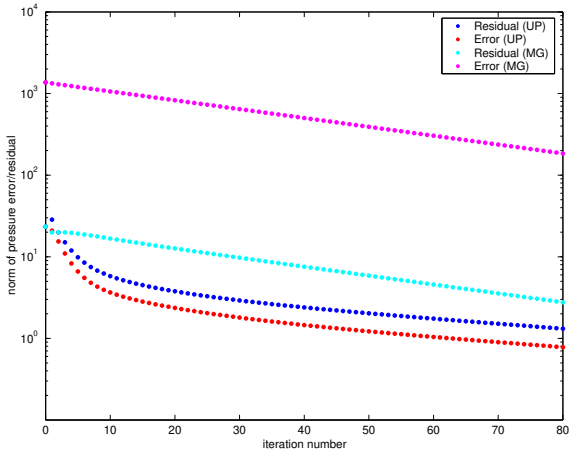




$\nu = 1$

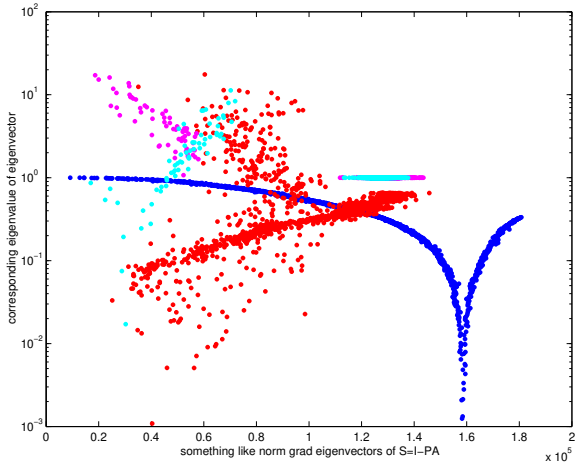
Medium

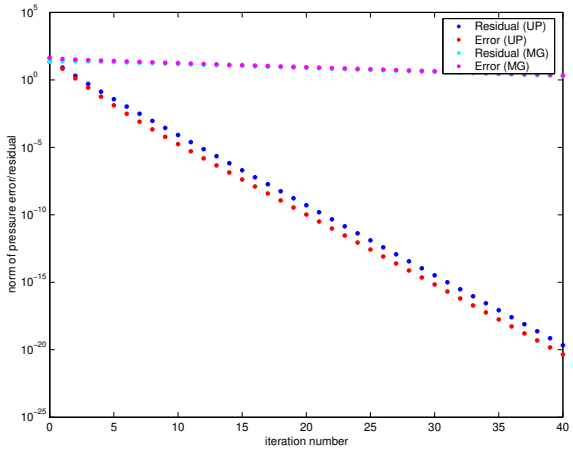




$\nu = 1$

Hard

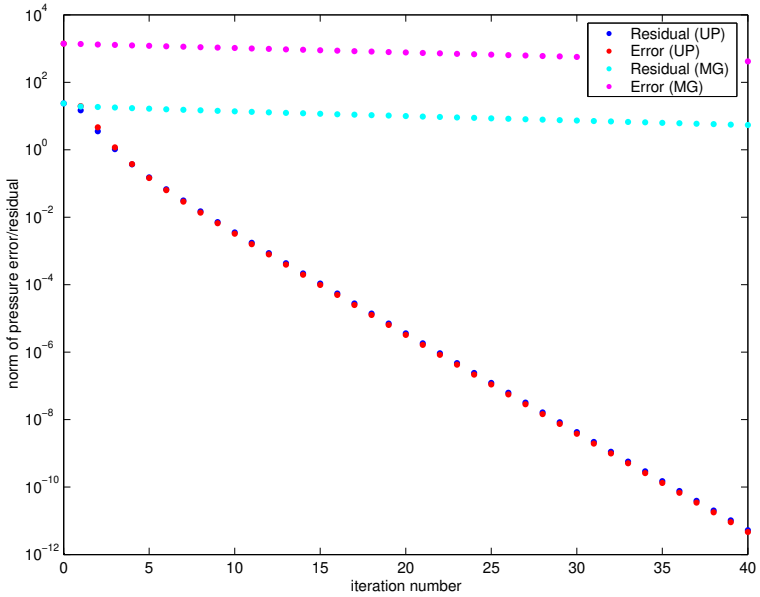




$\nu = 40$

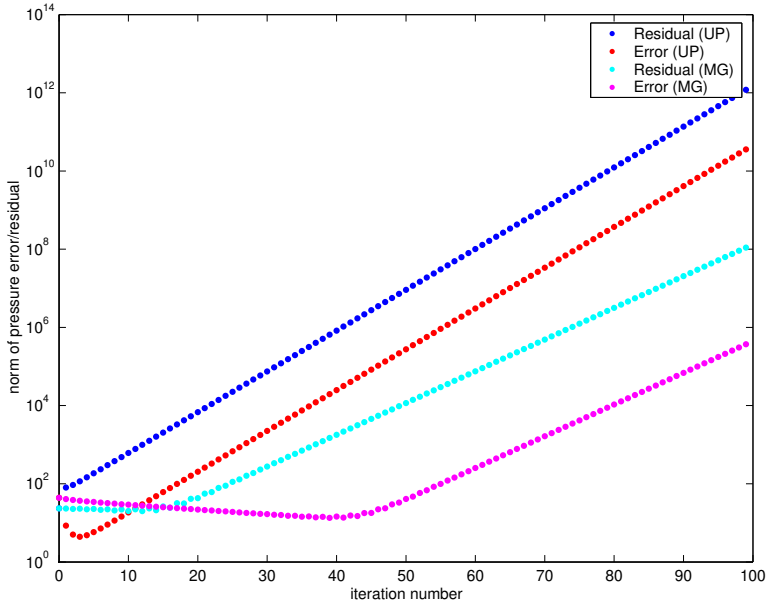
Over- and Under-Estimating Number of Smoothings:

Over-estimating ν for Medium



$\nu = 2$

Under-estimating ν for Hard



$\nu = 5$

Some Conclusions:

- Although we cannot prove it yet, these test cases suggest:
  - performance depends only on H/h and heterogeneity in κ (and on geometric factors),
  - convergence factor is independent of h (like MG but unlike Jacobi PCG),
  - and the number of smoothing steps ν is adjustable to obtain a contraction for any H/h and any κ.
- Because there is only one coarse mesh, not several, this scheme is a bit easier to apply than MG.

Work to be Done:

- Use MG techniques to prove convergence for some ν (with a convergence factor independent of h).
- For a given ν and H/h, will this scheme converge more quickly than a corresponding two-level MG scheme?
- Add a strategy to improve performance on middle frequencies (staggered grids, semi-coarsened grids, Krylov methods, or an ordinary coarse smoother).
- Use super coarsening for large problems (for a true *multi*-level method).

And Beyond:

- Apply to DG and expanded mixed methods.
- Apply to other physical problems.

Outline:

- *The Problem*
- *Subgrid Upscaling*
- *Two-level Multigrid-like Scheme*
- ***Change the Matrix, not the RHS***
  - Algorithm Description
  - Example
  - Further Study
- *Applications*

Goal:

Obtain fine-scale solution $(p_h, \mathbf{u}_h)$ as a direct result of an upscaled computation.

Problem of Spaces:

Fine-scale solution is

$$(p_h, \mathbf{u}_h) \in W_h \times \mathbf{V}_h.$$

Upscaled solution is

$$(p_{H,h}, \mathbf{u}_{H,h}) \in W_{H,h} \times \mathbf{V}_{H,h}.$$

But almost always

$$p_h \neq p_{H,h} \qquad \textbf{and} \qquad \mathbf{u}_h \neq \mathbf{u}_{H,h}.$$

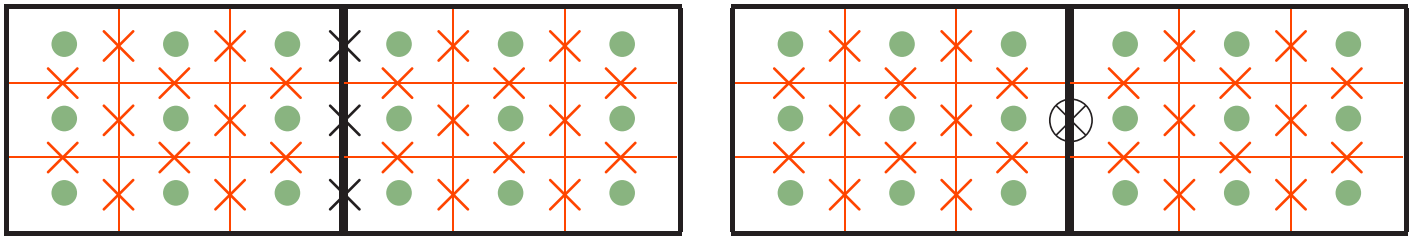
Even though

$$p_h \in W_{H,h} = W_h \qquad \textbf{and} \qquad \mathbf{V}_{H,h} \subset \mathbf{V}_h,$$

what's worse is ...

Difficulty:

$$\mathbf{u}_h \notin \mathbf{V}_{H,h}$$



Coarse elements are too restrictive: all flow across a coarse edge is constant along that edge.

Work-Around:

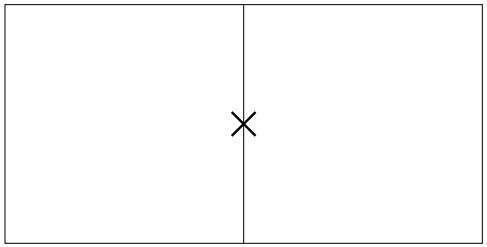
Allow coarse elements to vary along coarse edges:
this is a macro-element!

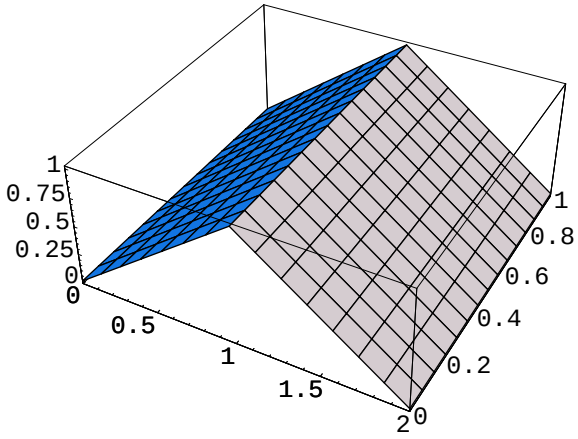
For an upscaled computation,
fix the shape along the coarse edge,
and use the one DOF to control its scale.

Use parameters β, a vector of weights,
to control the shape.

Coarse Basis Elements:

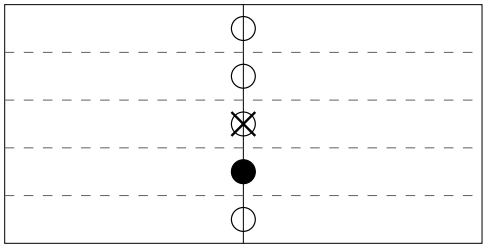
Original (Unmodified) Elements $\mathbf{V}_H$

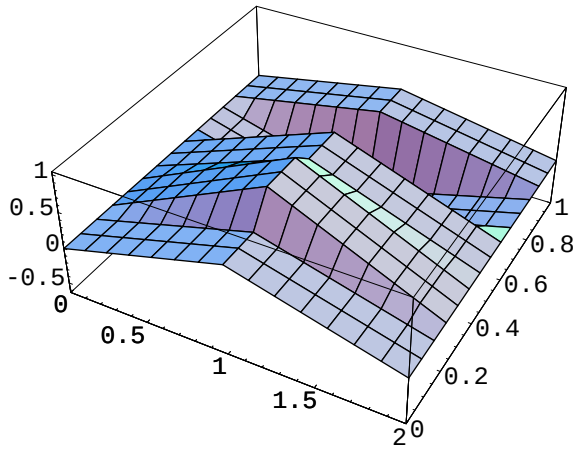




"$\beta = 1$"

Modified Elements $\tilde{\mathbf{V}}_H$





$\beta \neq 1$

$\tilde{\mathbf{V}}_H(\boldsymbol{\beta})$ is an RT0-like parameterized family of approximation subspaces of $\mathbf{V}_c$.

- $\dim(\mathbf{V}_H) = \dim(\tilde{\mathbf{V}}_H)$,
- **spatial support of elements stays the same,**
- $\delta\mathbf{V}_h$ **and** $W_{H,h}$ **stay the same,**
- **and usually** $\mathbf{V}_H \neq \tilde{\mathbf{V}}_H(\boldsymbol{\beta})$ **but** $\mathbf{V}_H = \tilde{\mathbf{V}}_H(\mathbf{1})$.

Like

$$\mathbf{V}_{H,h} = \mathbf{V}_H \oplus \delta\mathbf{V}_h,$$

let

$$\tilde{\mathbf{V}}_{H,h}(\boldsymbol{\beta}) = \tilde{\mathbf{V}}_H(\boldsymbol{\beta}) \oplus \delta\mathbf{V}_h.$$

Original Fine-Scale Problem:

With $p_h \in W_h$ and $\mathbf{u}_h \in \mathbf{V}_h$, solve

$$\begin{bmatrix} 0 & B^T \\ -B & D \end{bmatrix} \begin{bmatrix} p_h \\ \mathbf{u}_h \end{bmatrix} = \begin{bmatrix} b \\ c \end{bmatrix}$$

Decompose:

$$p_{H,h} = p_h \in W_h = W_{H,h}$$
$$\tilde{\mathbf{u}}_{H,h} + \check{\mathbf{u}}_{H,h} \equiv \mathbf{u}_h \in \mathbf{V}_h \equiv \tilde{\mathbf{V}}_{H,h} \oplus \check{\mathbf{V}}_{H,h}$$

And Partition:

$$\begin{bmatrix} 0 & \tilde{B}^T & \check{B}^T \\ -\tilde{B} & \tilde{\tilde{D}} & \tilde{\check{D}} \\ -\check{B} & \check{\tilde{D}} & \check{\check{D}} \end{bmatrix} \begin{bmatrix} p_h \\ \tilde{\mathbf{u}}_{H,h} \\ \check{\mathbf{u}}_{H,h} \end{bmatrix} = \begin{bmatrix} b \\ \tilde{c} \\ \check{c} \end{bmatrix}$$

Decompose:

$$p_{H,h} = p_h \in W_h = W_{H,h}$$

$$\tilde{\mathbf{u}}_{H,h} + \check{\mathbf{u}}_{H,h} \equiv \mathbf{u}_h \in \mathbf{V}_h \equiv \tilde{\mathbf{V}}_{H,h} \oplus \check{\mathbf{V}}_{H,h}$$

And Partition:

$$\begin{bmatrix} 0 & \tilde{B}^T & \check{B}^T \\ -\tilde{B} & \tilde{\tilde{D}} & \tilde{\check{D}} \\ -\check{B} & \check{\tilde{D}} & \check{\check{D}} \end{bmatrix} \begin{bmatrix} p_h \\ \tilde{\mathbf{u}}_{H,h} \\ \check{\mathbf{u}}_{H,h} \end{bmatrix} = \begin{bmatrix} b \\ \tilde{c} \\ \check{c} \end{bmatrix}$$

Decompose:

$$p_{H,h} = p_h \in W_h = W_{H,h}$$
$$\tilde{\mathbf{u}}_{H,h} + \check{\mathbf{u}}_{H,h} \equiv \mathbf{u}_h \in \mathbf{V}_h \equiv \tilde{\mathbf{V}}_{H,h} \oplus \check{\mathbf{V}}_{H,h}$$

And Partition:

$$\begin{bmatrix} 0 & \tilde{B}^T & \check{B}^T \\ -\tilde{B} & \tilde{\tilde{D}} & \tilde{\check{D}} \\ -\check{B} & \check{\tilde{D}} & \check{\check{D}} \end{bmatrix} \begin{bmatrix} p_h \\ \tilde{\mathbf{u}}_{H,h} \\ \check{\mathbf{u}}_{H,h} \end{bmatrix} = \begin{bmatrix} b \\ \tilde{c} \\ \check{c} \end{bmatrix}$$

Uzawa-like Method:

Goal: Find $\boldsymbol{\beta}^*$ so that $\check{\mathbf{u}}(\boldsymbol{\beta}^*) = \mathbf{0}$.

Initialize: $m = 0$ and $\boldsymbol{\beta}^0 = \mathbf{1}$.

Solve:

$$\begin{bmatrix} 0 & (\tilde{B}^m)^T \\ -\tilde{B}^m & \tilde{\tilde{D}}^m \end{bmatrix} \begin{bmatrix} p^m \\ \tilde{\mathbf{u}}^m \end{bmatrix} = \begin{bmatrix} b \\ \tilde{c}^m \end{bmatrix}$$

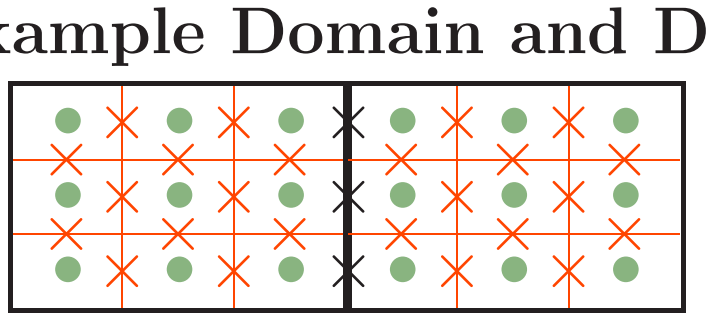
Compute: $\check{\mathbf{r}}^m = \check{c}^m + \check{B}^m p^m - \check{\tilde{D}}^m \tilde{\mathbf{u}}^m$.

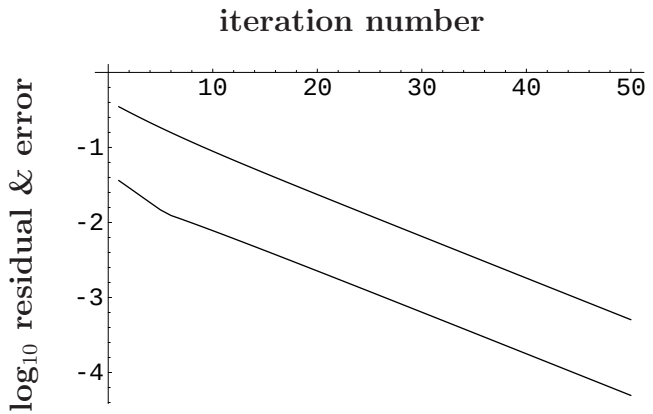
Test: If $\|\check{\mathbf{r}}^m\|$ is small, go to Stop.

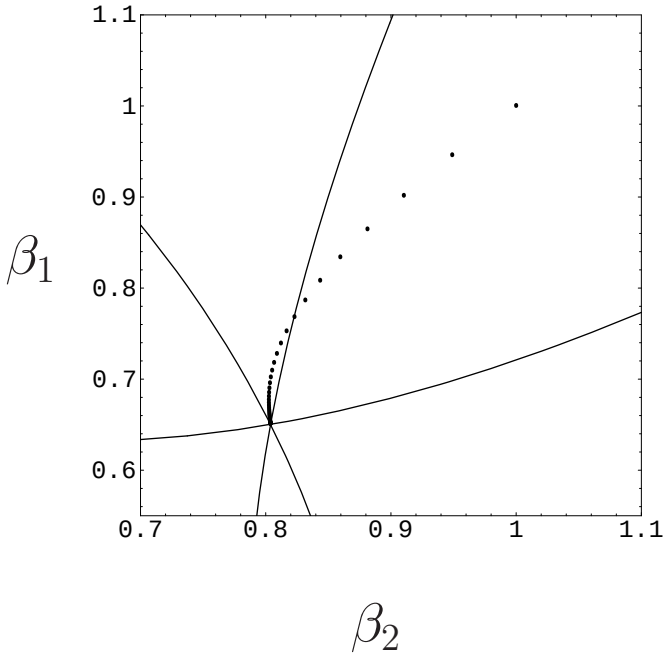
Update: $\boldsymbol{\beta}^{m+1} = \boldsymbol{\beta}^m + R\omega \check{\tilde{D}}^{-1} \check{\mathbf{r}}^m$ and $m \leftarrow m + 1$.
Go to Solve.

Stop: $\boldsymbol{\beta}^* \approx \boldsymbol{\beta}^m$

Convergence History of Method:







Another way to look at the problem:

Find $\boldsymbol{\beta}^*$ so $\check{\mathbf{r}}(\boldsymbol{\beta}^*) = \mathbf{0}$.

$\check{\mathbf{r}}(\boldsymbol{\beta})$ is a rational polynomial function of $\boldsymbol{\beta}$.

We want its root.

Another way to look at the problem:

Find $\boldsymbol{\beta}^*$ so $\check{\mathbf{r}}(\boldsymbol{\beta}^*) = \mathbf{0}$.

$\check{\mathbf{r}}(\boldsymbol{\beta})$ is a rational polynomial function of $\boldsymbol{\beta}$.

We want its root.

Use Newton's Method!

(or Broyden's etc.)

Uzawa-like Method:

Goal: Find $\boldsymbol{\beta}^*$ so that $\check{\mathbf{u}}(\boldsymbol{\beta}^*) = \mathbf{0}$.

Initialize: $m = 0$ **and** $\boldsymbol{\beta}^0 = \mathbf{1}$.

Solve:

$$\begin{bmatrix} 0 & (\tilde{B}^m)^T \\ -\tilde{B}^m & \tilde{\check{D}}^m \end{bmatrix} \begin{bmatrix} p^m \\ \tilde{\mathbf{u}}^m \end{bmatrix} = \begin{bmatrix} b \\ \tilde{c}^m \end{bmatrix}$$

Compute: $\check{\mathbf{r}}^m = \check{c}^m + \check{B}^m p^m - \check{\check{D}}^m \tilde{\mathbf{u}}^m$.

Test: If $\|\check{\mathbf{r}}^m\|$ is small, go to Stop.

Update: $\boldsymbol{\beta}^{m+1} = \boldsymbol{\beta}^m + R\omega \check{\check{D}}^{-1} \check{\mathbf{r}}^m$ **and** $m \leftarrow m + 1$.
Go to Solve.

Stop: $\boldsymbol{\beta}^* \approx \boldsymbol{\beta}^m$

Nonlinear Corrector Method:

Goal: Find $\boldsymbol{\beta}^*$ so that $\check{\mathbf{r}}(\boldsymbol{\beta}^*) = \mathbf{0}$.

Initialize: $m = 0$ and $\boldsymbol{\beta}^0 = \mathbf{1}$.

Solve:

$$\begin{bmatrix} 0 & \left(\tilde{B}^m\right)^T \\ -\tilde{B}^m & \tilde{\check{D}}^m \end{bmatrix} \begin{bmatrix} p^m \\ \tilde{\mathbf{u}}^m \end{bmatrix} = \begin{bmatrix} b \\ \tilde{c}^m \end{bmatrix}$$

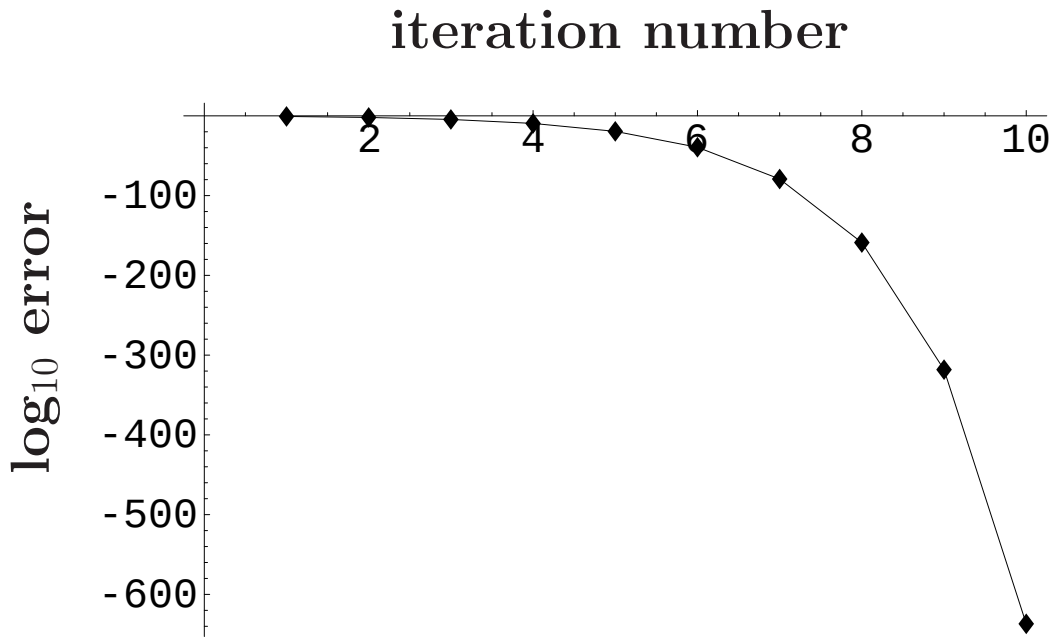
Compute: $\check{\mathbf{r}}^m = \check{c}^m + \check{B}^m p^m - \check{\check{D}}^m \tilde{\mathbf{u}}^m$.

Test: If $\|\check{\mathbf{r}}^m\|$ is small, go to Stop.

Update: $\boldsymbol{\beta}^{m+1} = \boldsymbol{\beta}^m + \mathcal{C}\left(R\omega \check{\check{D}}^{-1} \check{\mathbf{r}}^m\right)$ and $m \leftarrow m + 1$.
Go to Solve.

Stop: $\boldsymbol{\beta}^* \approx \boldsymbol{\beta}^m$

Convergence History of Method (With Newton Corrector):



Quadratic convergence!

Further Study:

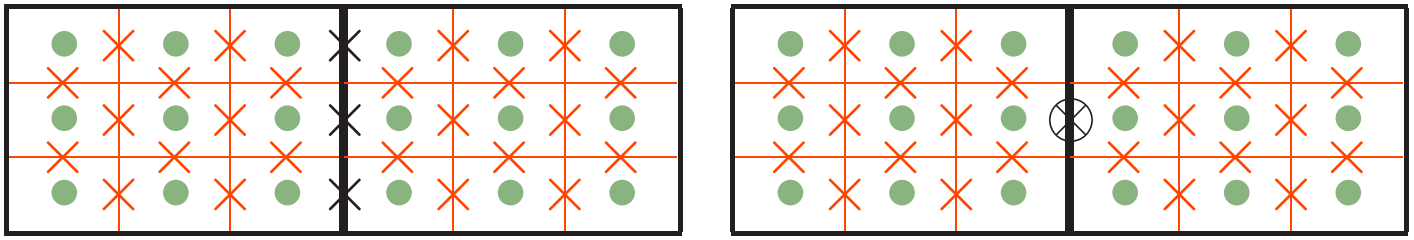
- **Theoretical framework for β, $\check{\mathbf{V}}_{H,h}$, normalization, and application of corrector.**
- **Conditions under which we can prove convergence.**
- **Different correctors.**
  Time, memory, and parallelization. Initialization.

Further Study:

- Theoretical framework for β, $\check{\mathbf{V}}_{H,h}$, normalization, and application of corrector.
- Conditions under which we can prove convergence.
- Different correctors.
  Time, memory, and parallelization. Initialization.

And Beyond:

- Apply to other physical problems.
- DG, expanded mixed, and hp methods.



Outline:

- *The Problem*
- *Subgrid Upscaling*
- *Two-level Multigrid-like Scheme*
- *Change the Matrix, not the RHS*
- ***Applications***

Goal:

Faster simulations.
(Bigger problems, Better sampling, Etc.)

Goal:

Faster simulations.
(Bigger problems, Better sampling, Etc.)

Some current computationally infeasible problems:

- **Lawyer Canyon**
- **Vuggy Data**
- **Geostatistical correlation length Vs. H**

Goal:

Faster simulations.
(Bigger problems, Better sampling, Etc.)

More applications:

- **Other phyical problems**
- **Other FEMs**