Multiscale Methods for Porous Media Flow

Knut–Andreas Lie
Vegard Kippe, Stein Krogstad, Jørg E. Aarnes

SINTEF ICT, Dept. Applied Mathematics

NSCM-18, November 28-29, 2005

Outline

Pressure equation:

$$-\nabla\big(\mathbf{K}(\mathbf{x})\lambda(S)\nabla p\big) = q, \qquad \mathbf{v} = -\mathbf{K}(\mathbf{x})\lambda(S)\nabla p,$$

Fluid transport:

$$\phi\partial_t S + \nabla\cdot(\mathbf{v}f(S)) = \varepsilon\nabla\big(\mathbf{D}(S,\mathbf{x})\nabla S\big)$$

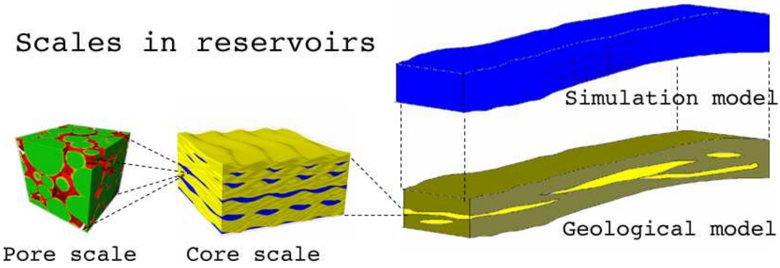
Porous sandstones often have repetitive layered structures, but faults and fractures caused by stresses in the rock disrupt flow patterns[1]:

[1] Photo: Silje Søren Berg, CIPR, Univ. Bergen

Scales in Porous Media Flow

The scales that impact fluid flow in oil reservoirs range from

- the micrometer scale of pores and pore channels
- via dm–m scale of well bores and laminae sediments
- to sedimentary structures that stretch across entire reservoirs.



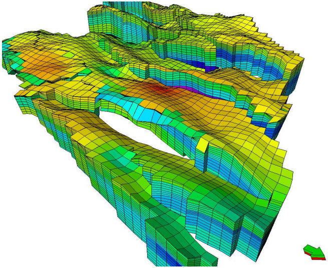
Geological Models
The knowledge database in the oil company

Geomodels consist of geometry and rock parameters (permeability $\mathbf{K}$ and porosity ϕ):

- $\mathbf{K}$ spans many length scales and has multiscale structure

$$\max \mathbf{K} / \min \mathbf{K} \sim 10^3\text{–}10^{10}$$

- Details on all scales impact flow

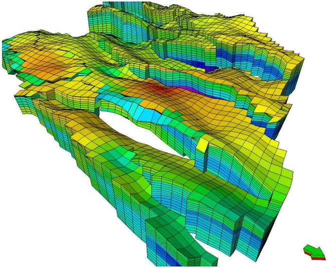


Geomodels consist of geometry and rock parameters (permeability $\mathbf{K}$ and porosity ϕ):

- $\mathbf{K}$ spans many length scales and has multiscale structure

$$\max \mathbf{K} / \min \mathbf{K} \sim 10^3\text{–}10^{10}$$

- Details on all scales impact flow



Gap between simulation models and geomodels:

- High-resolution geomodels may have $10^7 - 10^9$ cells
- Conventional simulators are capable of about $10^5 - 10^6$ cells

Traditional solution: upscaling of parameters

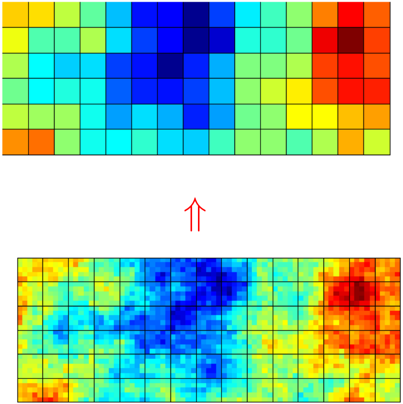
Upscaling Geomodels

Upscaling a geomodel to a coarser simulation grid:

- Combine cells to derive coarse grid
- Derive new efficient cell properties
- Fewer cells ⇒ faster simulation/less storage

However:

- Robust upscaling can be difficult and work-intensive



Upscaling the Pressure Equation

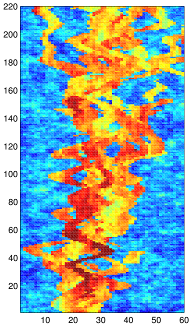
Assume that u satisfies the elliptic PDE:

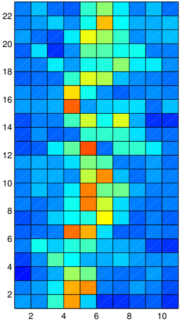
$$-\nabla\big(a(x)\nabla u\big) = f.$$

Upscaling amounts to finding a new field $a^*(\bar{x})$ on a coarser grid such that

$$-\nabla\big(a^*(\bar{x})\nabla u^*\big) = \bar{f},$$

$$u^* \sim \bar{u}, \qquad q^* \sim \bar{q}\ .$$



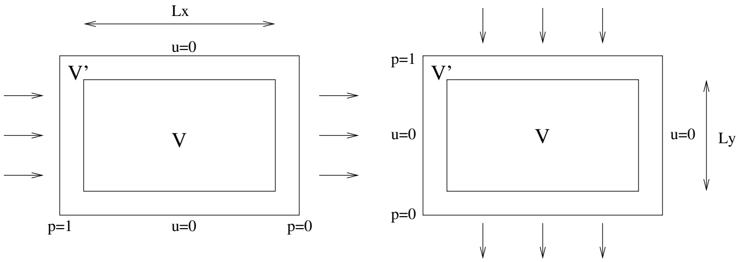


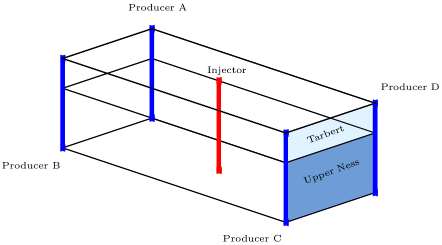
Here the overbar denotes averaged quantities on a coarse grid.

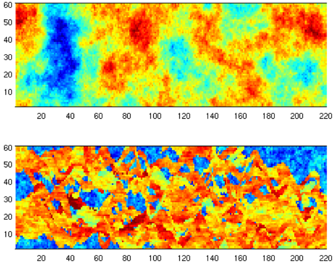
Upscaling the Pressure Equation, cont'd

How do we represent fine-scale heterogeneities on a coarse scale?

- Arithmetic, geometric, harmonic, or power averaging
  $\left(\frac{1}{|V|}\int_V a(x)^p \, dx\right)^{1/p}$
- Equivalent permeabilities ($a^*_{xx} = -Q_x L_x / \Delta P_x$)

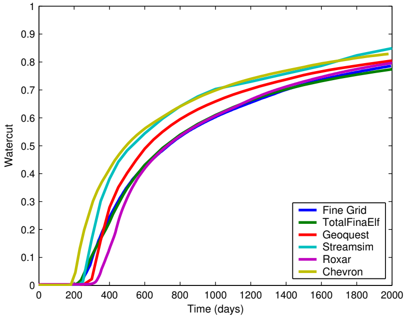


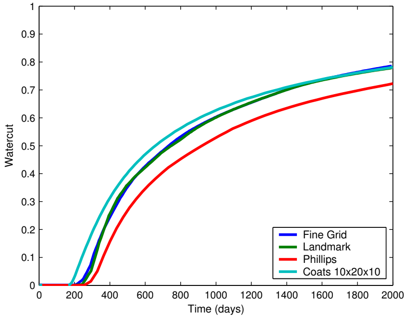




- Geomodel: $60 \times 220 \times 85 \approx 1,1$ million grid cells
- Simulation: 2000 days of production

10th SPE Comparative Solution Project
Upscaling results reported by industry





Developing an Alternative to Upscaling

Observation:

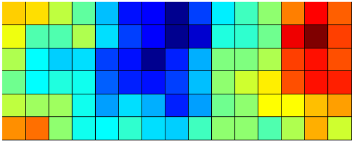
- Variations on small scale can have large impact on large-scale flow patterns

We therefore seek a methodology which:

- gives a detailed image of the flow pattern on the fine scale, without having to solve the full fine-scale system
- is robust and flexible with respect to the coarse grid
- is robust and flexible with respect to the fine grid and the fine-grid solver
- is accurate and conservative
- is fast and easy to parallelise

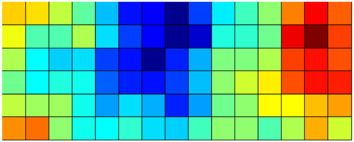
Standard method

Upscaled model:



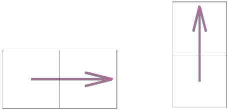
Standard method

Upscaled model:



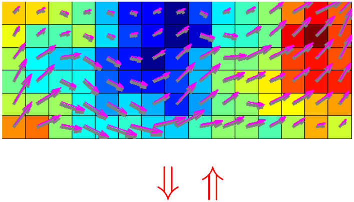
$\Downarrow$

Building blocks:

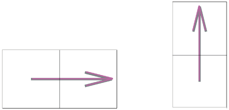


Standard method

Upscaled model:

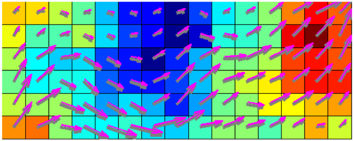


Building blocks:



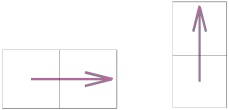
Standard method

Upscaled model:



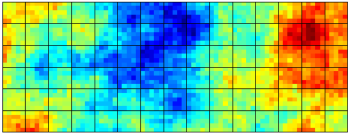
⇓ ⇑

Building blocks:



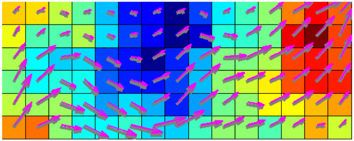
Two-scale method

Geomodel:



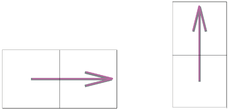
Standard method

Upscaled model:



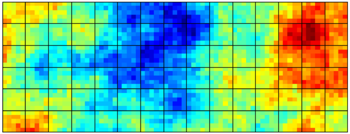
$\Downarrow \quad \Uparrow$

Building blocks:



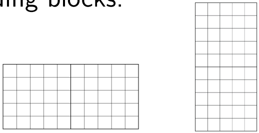
Two-scale method

Geomodel:



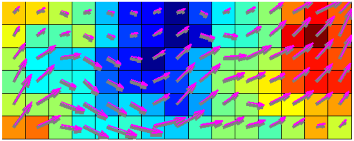
$\Downarrow$

Building blocks:



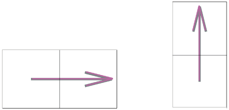
Standard method

Upscaled model:



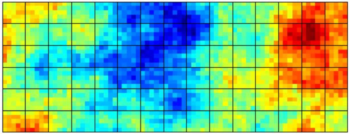
⇓ ⇑

Building blocks:



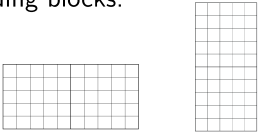
Two-scale method

Geomodel:

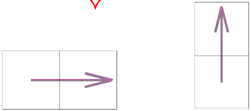


⇓

Building blocks:



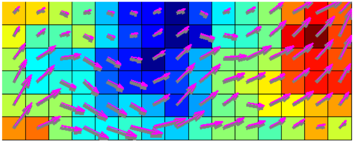
⇓



From Upscaling to Multiscale Methods

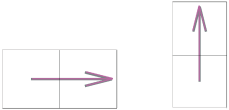
Standard method

Upscaled model:



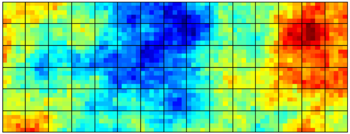
⇓ ⇑

Building blocks:



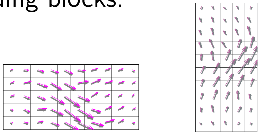
Two-scale method

Geomodel:

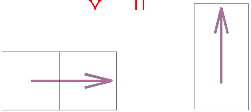


⇓

Building blocks:



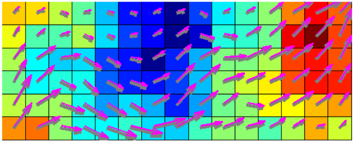
⇓ ⇑



From Upscaling to Multiscale Methods

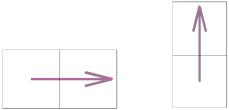
Standard method

Upscaled model:



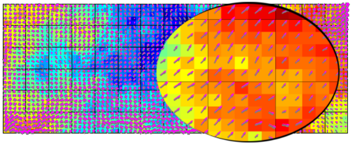
$\Downarrow$ $\Uparrow$

Building blocks:



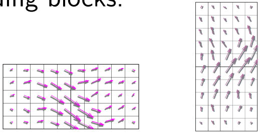
Two-scale method

Geomodel:

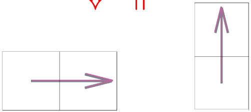


$\Downarrow$ $\Uparrow$

Building blocks:



$\Downarrow$ $\Uparrow$



From Upscaling to Multiscale Methods, cont'd

1. Global upscaling methods (Nielsen, Holden, Tveito)
   - global boundary conditions, minimization of error functional
2. Local-global upscaling methods (Durlofsky et al.)
   - global boundary conditions + iterative improvement
3. Nested gridding (Gautier, Blunt & Christie)
   - Upscaling + local reconstruction of fine-scale velocities
4. Multiscale finite elements
   - basis functions with subscale resolution
   - finite elements (Hou & Wu) – pressure
   - mixed elements (Chen & Hou; Aarnes et al.) — velocity
   - finite volumes (Jenny et al.) — pressure
5. Variatonal multiscale methods (Hughes et al.; Arbogast; Larson & Målqvist; Juanes)
   - direct decomposition of the solution, $V = V_c \oplus V_f$

Mixed formulation:

Find $(v, p) \in H_0^{1,\text{div}} \times L^2$ such that

$$\int (\lambda K)^{-1} u \cdot v \, dx - \int p \nabla \cdot u \, dx = 0, \qquad \forall u \in H_0^{1,\text{div}},$$
$$\int \ell \nabla \cdot v \, dx = \int q \ell \, dx, \quad \forall \ell \in L^2.$$

Multiscale discretisation:

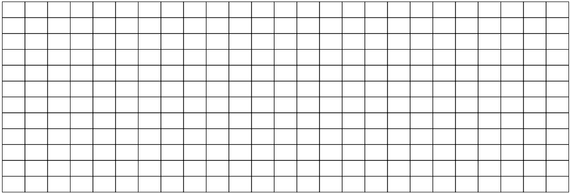
Seek solutions in low-dimensional subspaces

$$U^{ms} \subset H_0^{1,\text{div}} \text{ and } V \in L^2,$$

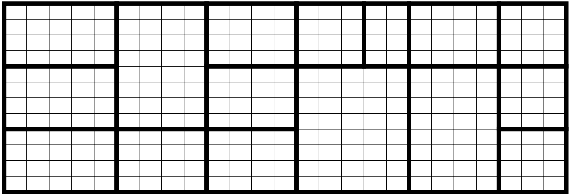
where local fine-scale properties are incorporated into the basis functions.

Multiscale Mixed Finite Elements
Grids and Basis Functions

We assume we are given a *fine* grid with permeability and porosity attached to each fine-grid block.

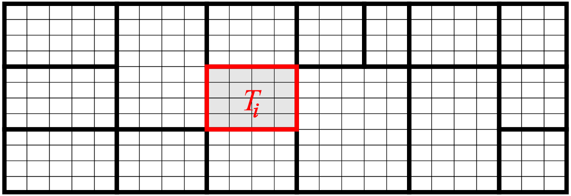


We assume we are given a *fine* grid with permeability and porosity attached to each fine-grid block.



We construct a *coarse* grid, and choose the discretisation spaces V and U^{ms} such that:

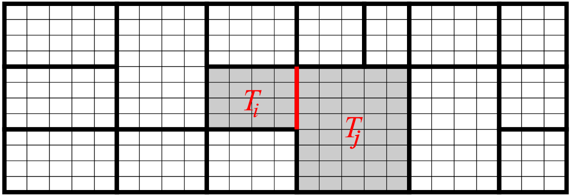
We assume we are given a *fine* grid with permeability and porosity attached to each fine-grid block.



We construct a *coarse* grid, and choose the discretisation spaces V and U^{ms} such that:

- For each coarse block T_i, there is a basis function $\phi_i \in V$.

We assume we are given a *fine* grid with permeability and porosity attached to each fine-grid block.



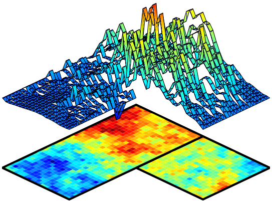
We construct a *coarse* grid, and choose the discretisation spaces V and U^{ms} such that:

- For each coarse block T_i, there is a basis function $\phi_i \in V$.
- For each coarse edge Γ_{ij}, there is a basis function $\psi_{ij} \in U^{ms}$.

For each coarse edge Γ_{ij}, define a basis function

$$\psi_{ij} : T_i \cup T_j \to R^2$$

with unit flux through Γ_{ij} and no flow across $\partial(T_i \cup T_j)$.



We use $\psi_{ij} = -\lambda K \nabla \phi_{ij}$ with

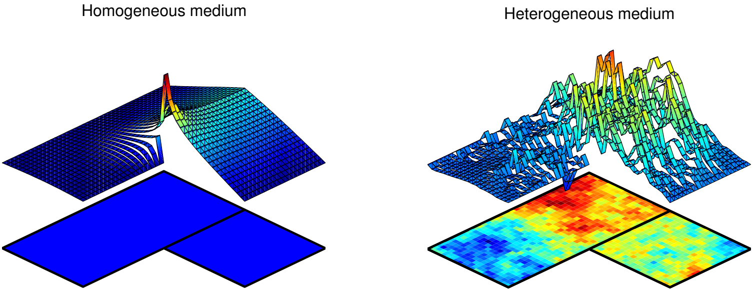
$$\nabla \cdot \psi_{ij} = \begin{cases} w_i(x), & \text{for } x \in T_i, \\ -w_j(x), & \text{for } x \in T_j, \end{cases}$$

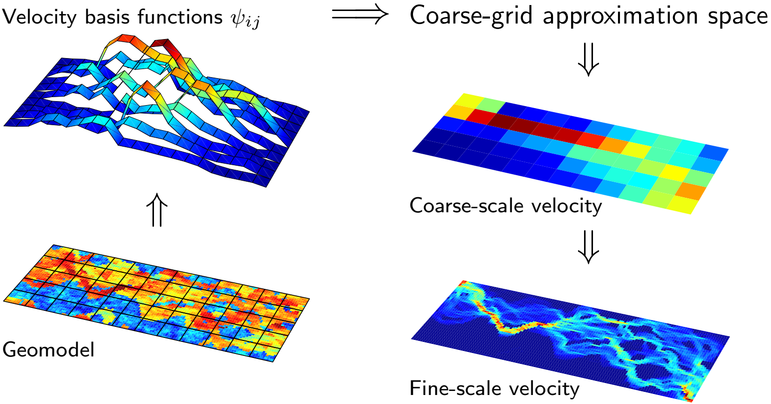
with boundary conditions $\psi_{ij} \cdot n = 0$ on $\partial(T_i \cup T_j)$.

Multiscale Mixed Finite Elements
Basis for Velocity Field, cont'd

Homogeneous coefficients and rectangular support domain:
basis function = lowest order Raviart-Thomas basis

MsMFEM = extension to cases with subscale variation in coefficients and non-rectangular support domain





For the MsMFEM the fine-scale velocity field is a linear superposition of basis functions: $v = \sum_{ij} v_{ij}^{*} \psi_{ij}$.

Multiscale:
Incorporates small-scale effects into coarse-scale solution

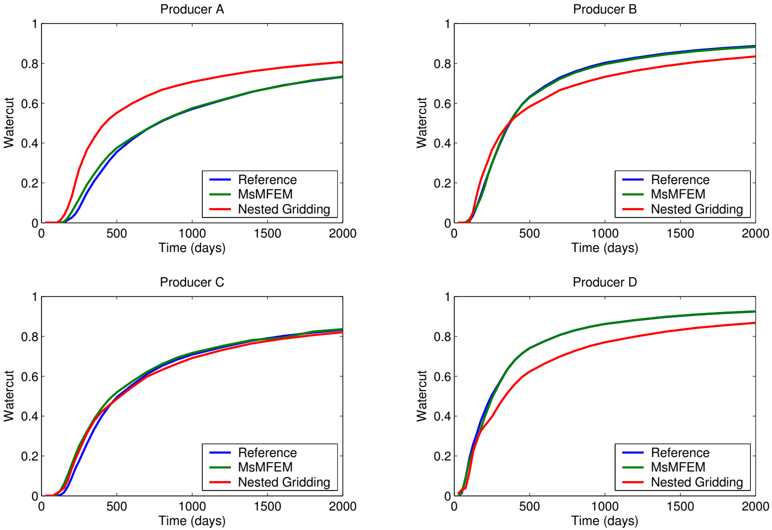
Conservative:
Mass conservative on coarse grid and on the subgrid scale

Scalable:
Well suited for parallel implementation since basis functions are processed independently

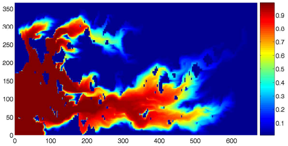
Flexible:
No restrictions on subgrids and subgrid numerical method. Few restrictions on the shape of the coarse blocks

Fast:
The method is fast when avoiding regeneration of (most of) the basis functions at every time step

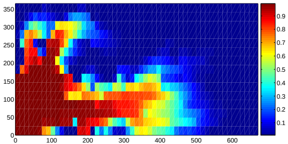
SPE10 Revisited ($5 \times 11 \times 17$ Coarse Grid)



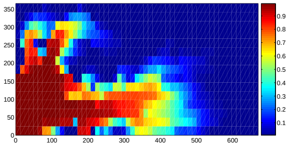
Nested gridding: upscaling + downscaling



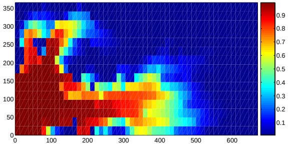
reference (240×880)



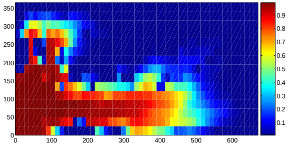
MsMFEM



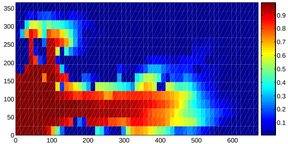
MsFVM



ALGU-NG

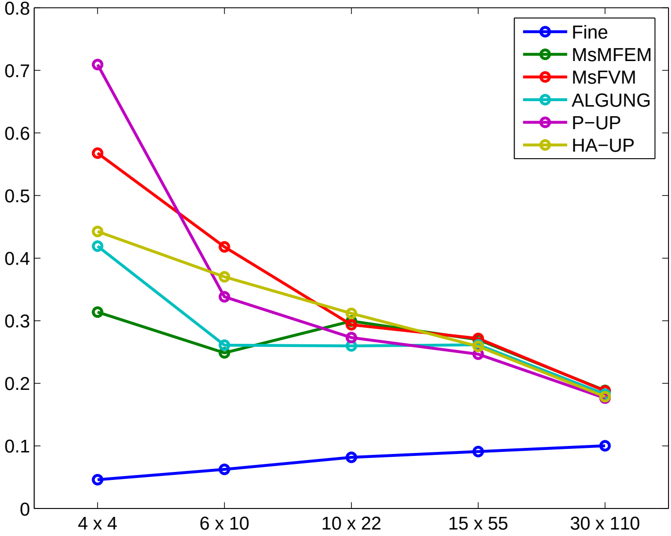


pressure method

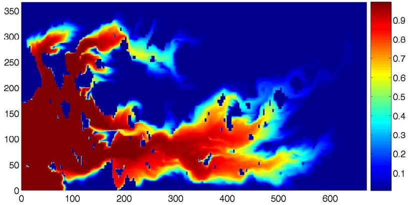


harmonic-arithmetic

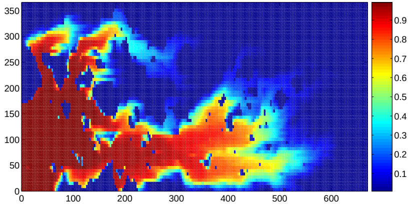
Multiscale vs. Upscaling
Saturation Errors on the Upscaled Grid



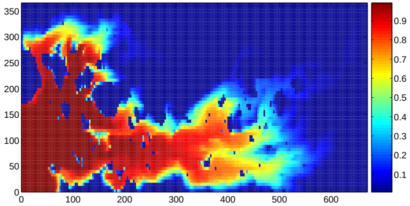
Multiscale vs. Upscaling/Downscaling



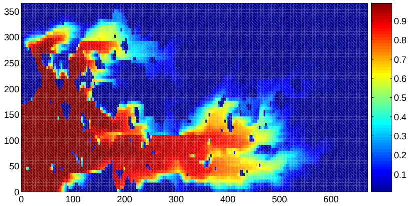
reference (240 × 880)



MsMFEM

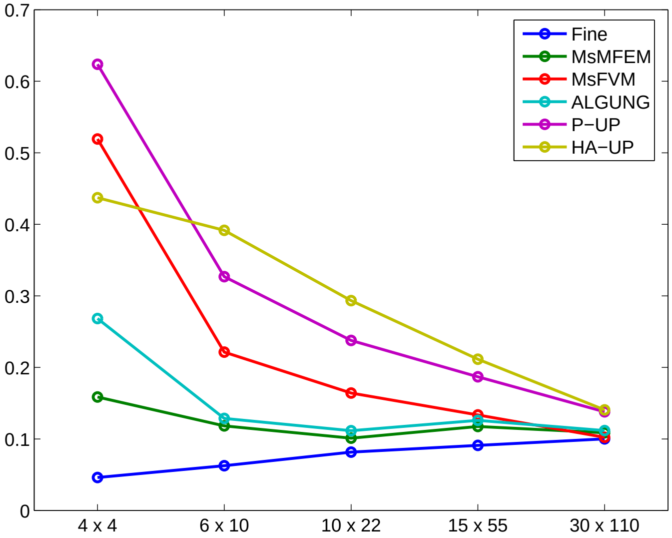


MsFVM

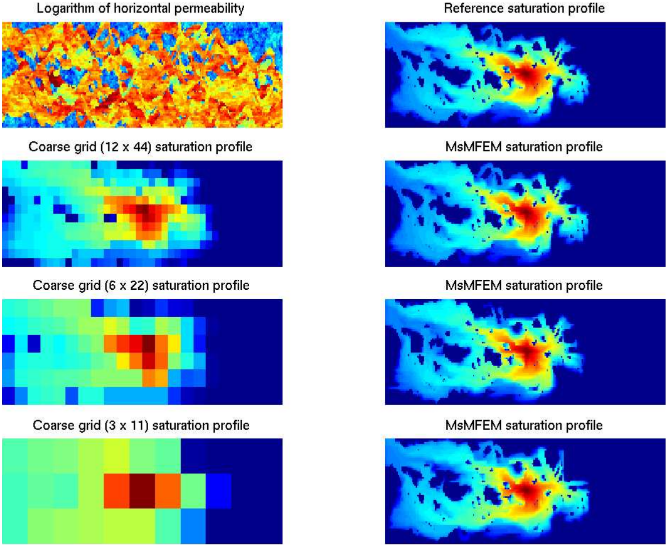


ALGU-NG

Multiscale vs. Upscaling
Saturation Errors on the Fine Grid

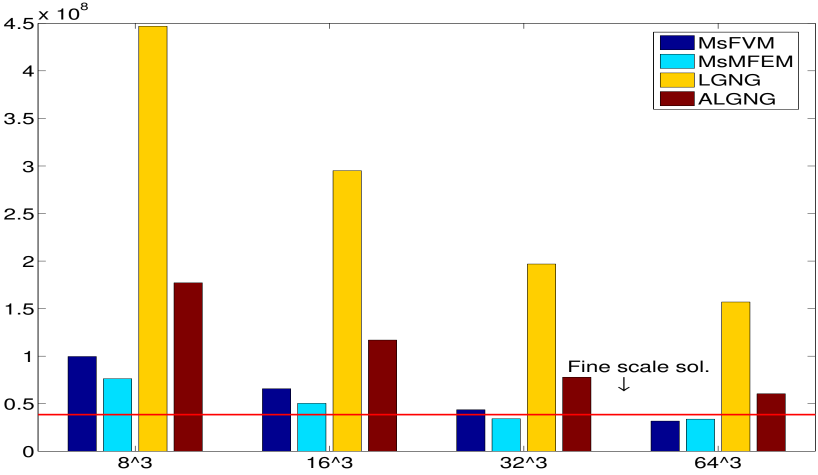


Robustness
SPE10, Layer 85 (60×220 Grid)



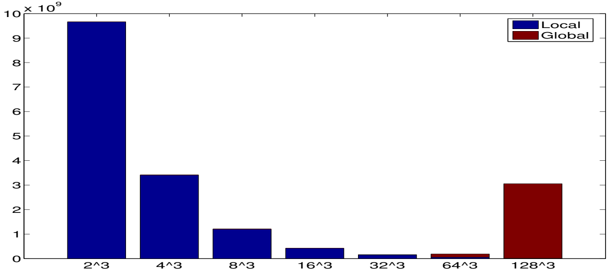
Computational Complexity
Order of Magnitude Argument

Example: 3D (128x128x128), $\alpha = 1.2$ and $m = 3$



Direct solution more efficient, so why bother with multiscale?

- Full simulation: $\mathcal{O}(10^2)$ steps.
- Basis functions need not be recomputed

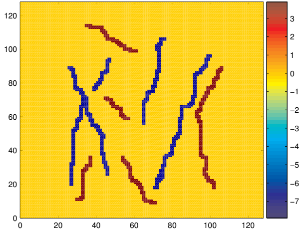


Also:

- Possible to solve very large problems
- Easy parallelization

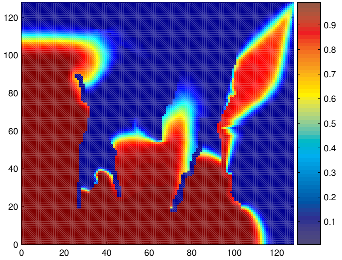
Strongly Heterogeneous Structures

Logarithm of k_x

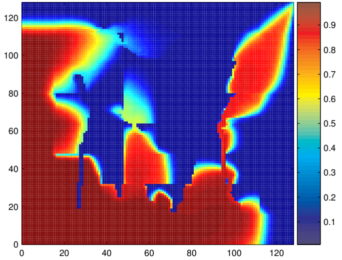


$$k_{\text{red}} = 10^4$$
$$k_{\text{yellow}} = 1$$
$$k_{\text{blue}} = 10^{-8}$$

Coarse grid $= 8 \times 8$.



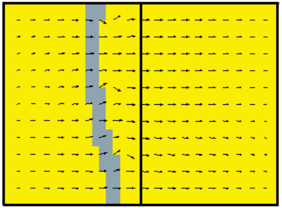
Reference



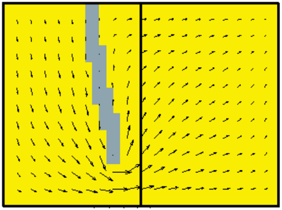
MsMFEM

Problem: Traversing Barriers

Problems occur when a basis function forces flow through a barrier:



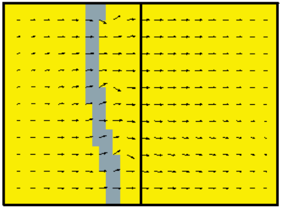
Potential problem



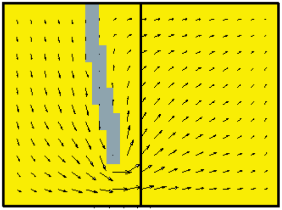
No problem

Problem: Traversing Barriers

Problems occur when a basis function forces flow through a barrier:



Potential problem



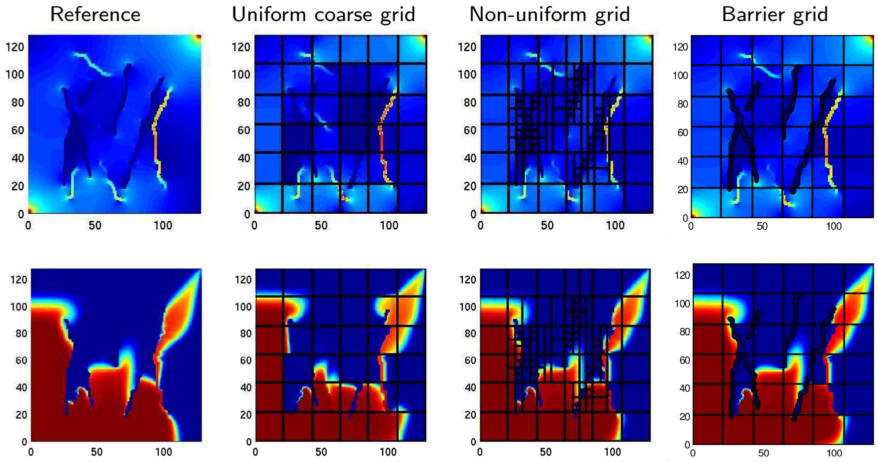
No problem

Problem-cases can be detected automatically through the indicator

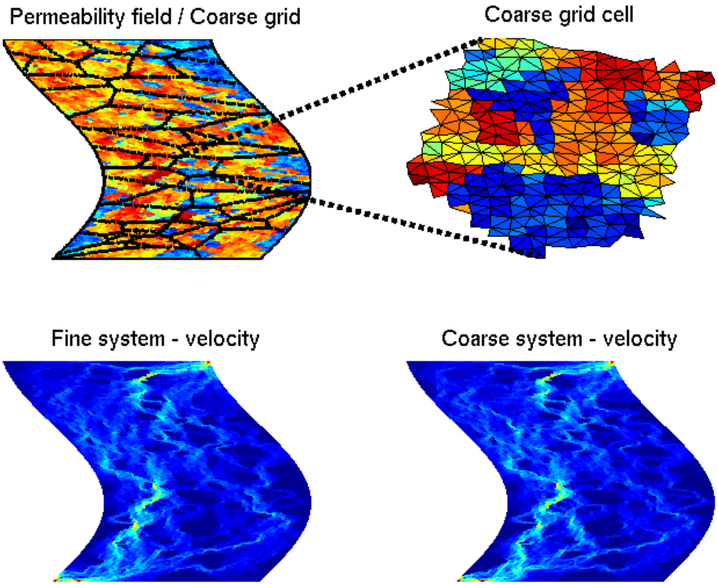
$$v_{ij} = \psi_{ij} \cdot (\lambda K)^{-1} \psi_{ij}.$$

If $v_{ij}(x) > C$ for some $x \in T_i$, then split T_i, and generate basis functions for the new faces.

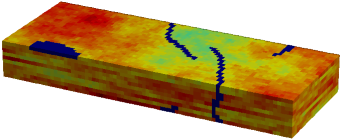
Barrier Case, revisited

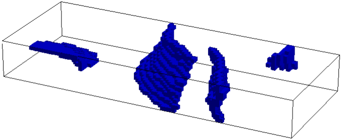


Flexibility wrt. Grids

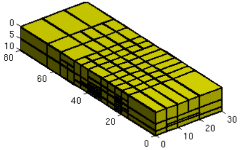


Flexibility wrt. Grids
Around Flow Barriers

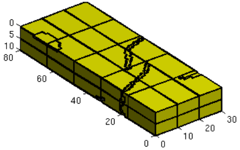




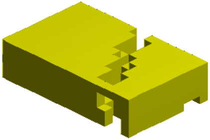
Non-uniform grid, hexahedral cells



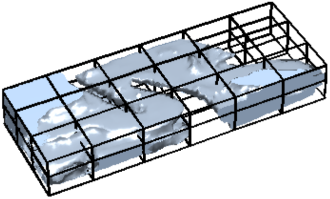
Non-uniform grid, general cells



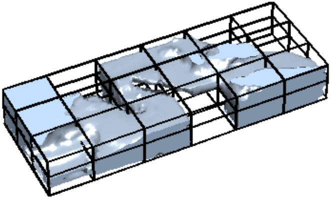
General grid-cell



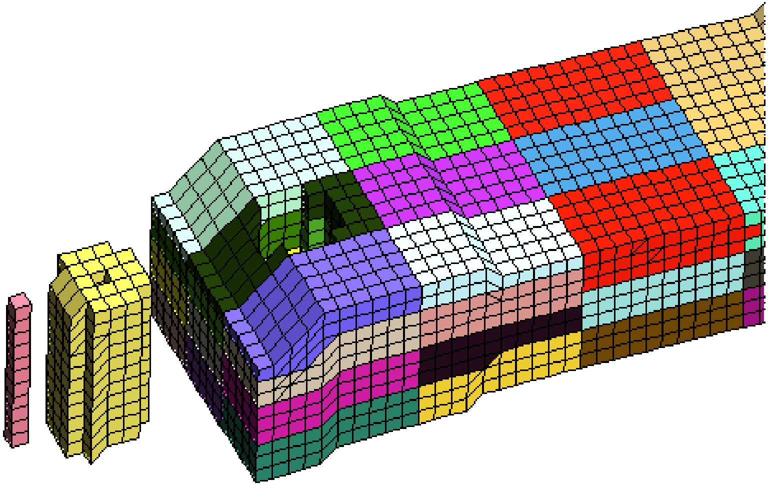
Saturation-plot from reference solution



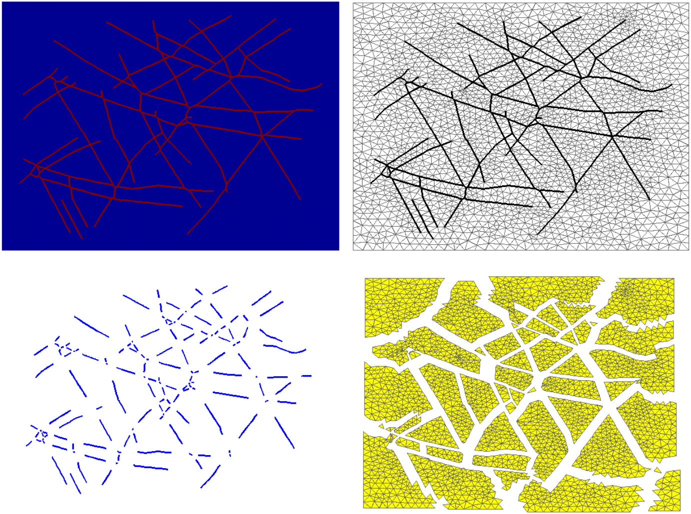
Saturation-plot from coarse-grid solution



Flexibility wrt. Grids
Around Wells



Flexibility wrt. Grids
Fracture Networks


[2]

[2] Courtesy of M. Karimi-Fard, Stanford

Concluding Remarks

Upscaling is *and will* be an important part of the reservoir modelling workflow

Multiscale methods may replace upscaling/downscaling for simulation purposes, because they:

- give better resolution
- are more flexible
- may be faster

However, a lot of (exciting) research needs to be done..