Comparison of Three Multiscale Methods for Two-Phase Flow in Heterogeneous Porous Media

J.E. Aarnes, V. Kippe, S. Krogstad, and Knut–Andreas Lie

SINTEF ICT, Dept. Applied Mathematics

SIAM Conference on Mathematical and Computational Issues in the Geosciences, June 7–10, Avignon, France

Outline

1. Introduction – Three Multiscale Methods
   - Local-global upscaling and nested gridding
   - Multiscale mixed finite elements
   - Multiscale finite volumes
2. Comparison of Methods
   - Computational complexity
   - A fluvial reservoir (SPE 10)
   - Anisotropic medium / high aspect ratios
   - Global boundary conditions for MsFVM and MsMFEM
   - Shale barriers and high-permeable channels
3. Conclusion

Incompressible Two-Phase Flow

Fractional formulation (no gravity or capillary forces):

$$-\nabla\big(k\lambda(S)\nabla p\big) = q, \qquad v = -k\lambda(S)\nabla p,$$
$$\phi\partial_t S + \nabla\cdot\big(vf(S)\big) = 0$$

Numerical solution by operator splitting (each equation by a specialised numerical method):

pressure: hierarchical multiscale method

saturation: finite volumes or streamlines

Iterated implicit (+ domain decomposition) converges within a few iterations and is therefore an alternative to fully implicit!

Multiscale Methods for the Pressure Equation

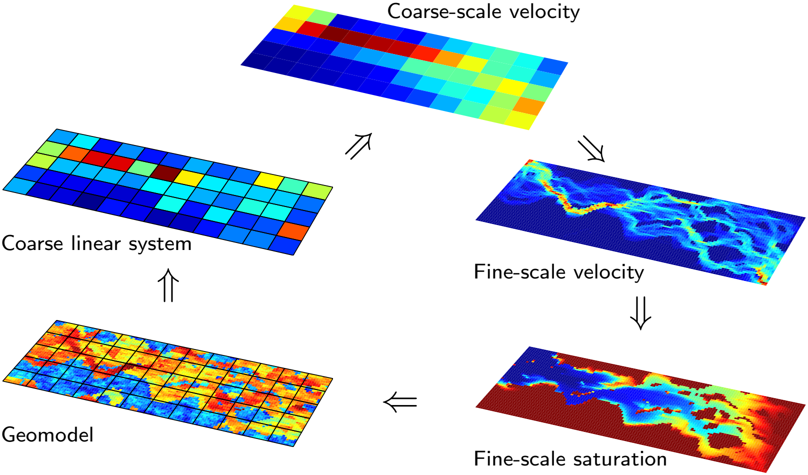
Small scale variations in the permeability can have a strong impact on large scale flow and should be resolved properly.

- the pressure may be well resolved on a coarse grid
- the fluid transport should be solved on the finest scale possible

Multiscale method – accurate velocity fields on the fine scale:

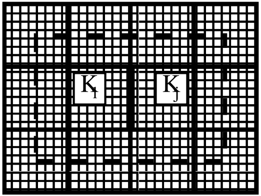
- Local-global upscaling and nested gridding (LGU-NG)
  - iterative global upscaling method
  - reconstructed fine-scale velocity
- Multiscale finite volume method (MsFVM)
  - fine-scale pressure, reconstructed fine-scale velocity
- Mixed multiscale finite element method (MsMFEM)
  - fine-scale velocity, coarse-scale pressure

Multiscale Simulation



1. Upscale transmissibility:

$$-\nabla \cdot k\nabla p = 0 \quad \text{in} \quad \Omega_{lj}$$
$$p = Ip^* \quad \text{in} \quad \partial\Omega_{lj}$$



$$T^*_{lj} = \frac{\int_{\partial K_l \cap \partial K_j} v \cdot n_{lj}\, ds}{\int_{K_l} p\, dx - \int_{K_j} p\, dx}$$

2. Solve coarse-scale problem:

$$\sum_j T^*_{lj}(p_l - p_j) = \int_{K_l} q\, dx \quad \forall K_l$$

3. Construct fine-scale velocity:

$$v = -k\nabla p, \quad \nabla \cdot v = q \quad \text{in } K_l$$
$$v \cdot n = \frac{T_{ki}(v^* \cdot n_{lj})}{\sum_{\gamma_{ki} \subset \Gamma_{lj}} T_{ki}} \quad \text{on } \partial K_l$$

(Here i runs over the underlying fine grid)

Local-Global Upscaling and Nested Gridding

Global upscaling + fine-grid reconstruction = a 'multiscale' method:

- Compute initial T^*_{lj}'s using standard upscaling
- Solve global coarse-scale pressure equation with T^*_{lj}'s
- Until convergence (in v and p)
  - Interpolate between pressures to get BC for local flow problems
  - Compute new T^*_{lj}'s from local flow problems
  - Solve global coarse-scale pressure equation with new T^*_{lj}'s
- Solve coarse-scale problem (wells and BC) with upscaled T^*_{lj}'s
- Reconstruct fine-scale velocity field with nested gridding

Multiscale Mixed Finite Elements

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:

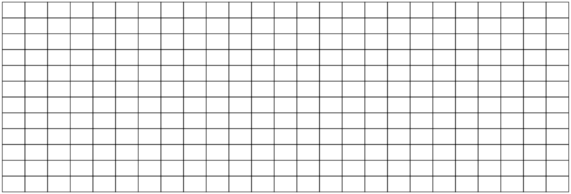
Seek solutions in low-dimensional subspaces

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

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

Grids and Basis Functions

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

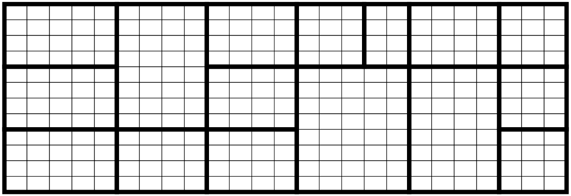


We construct a *coarse* grid, and choose the discretisation spaces V and U^{ms} such that:

- 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 $\psi_{ij} \in U^{ms}$.

Grids and Basis Functions

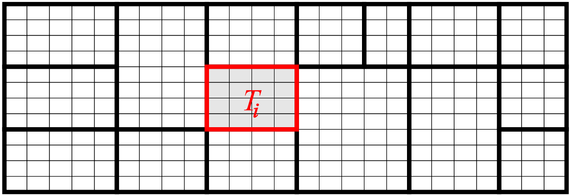
We assume we are given a *fine* grid with permeability and porosity attached to each fine-grid block.



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 $\psi_{ij} \in U^{ms}$.

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

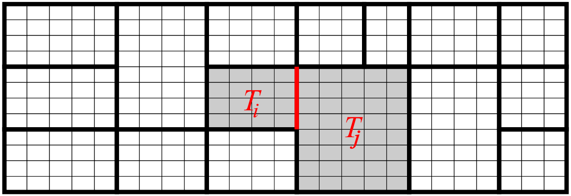


We construct a *coarse* grid, and choose the discretisation spaces V and U^{ms} such that:

- 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 $\psi_{ij} \in U^{ms}$.

Grids and Basis Functions

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



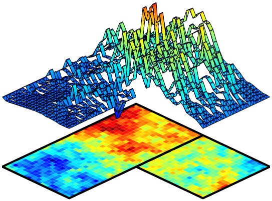
We construct a *coarse* grid, and choose the discretisation spaces V and U^{ms} such that:

- 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 $\psi_{ij} \in U^{ms}$.

Basis for the Velocity Field

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

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



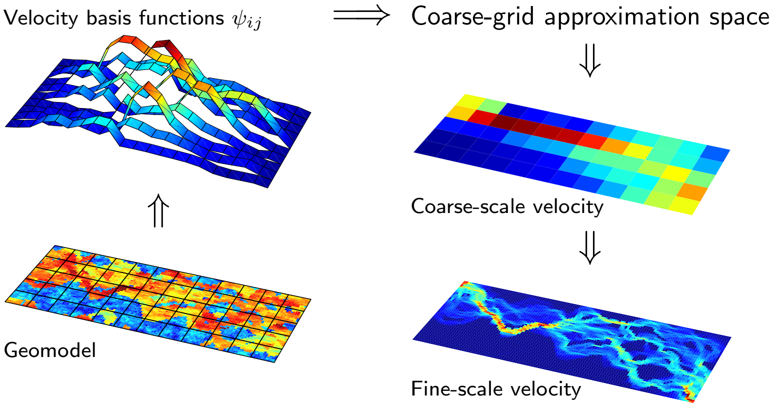
with unit flux through Γ_{ij} and no flow across $\partial(T_i \cup T_j)$.

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



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

Multiscale Finite-Volume Method

- Construct basis functions for pressure: $\phi_j = \phi(K_j)$.
- Compute flux contributions: $f^*_{j,l} = -\int_{\partial K_l} k\nabla\phi_j \cdot n_l \, ds$.
- Solve coarse-scale problem:

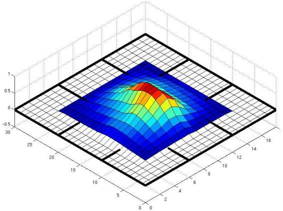
$$\sum_j p^*_j f^*_{j,l} = \int_{K_l} q \, dx$$

  The idea is to express the pressure as a linear superposition of the base functions: $p^* = \sum_j p^*_j \phi_j$.
- Reconstruct fine-grid velocity field from mass-conservative field on coarse grid:

$$v = -k\nabla p, \quad \nabla \cdot v = q, \quad \text{in } K_l$$
$$v \cdot n = -k\nabla p^* \cdot n_l \quad \text{on } \partial K_l.$$

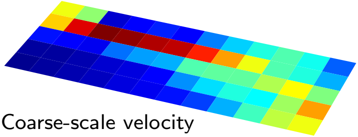
Multiscale Finite-Volume Method

Pressure basis functions



$\Rightarrow$ Mass balance equations on coarse grid

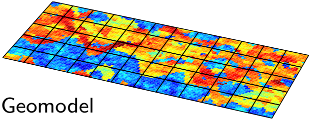
$\Downarrow$



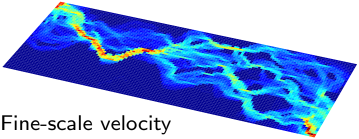
Coarse-scale velocity

$\Downarrow$

$\Uparrow$



Geomodel



Fine-scale velocity

Efficient linear algebra is crucial for effcient pressure solution:

- The finite volume formulation, used by both MsFVM and LGU-NG results in Symmetric Positive Definite (SPD) linear systems.
- The mixed finite element formulation yields a saddle point problem (indefinite system), which is generally thought to be harder to solve.
- However, the MsMFEM can be reformulated as an equivalent Mixed Hybrid FEM† that results in a SPD system.

† Hybrid MFEM: lifting the restriction of continuous edge velocities and reintroducing continuity by applying "Lagrange multipliers" (edge pressures)

Assumptions

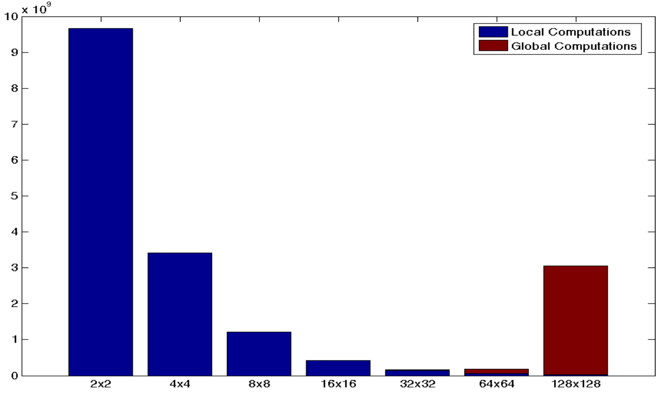
- Dominating factor is solution of linear systems, i.e., we ignore the time associated with assembly and determination of boundary conditions for local problems.
- Time to solve linear system of size N:

$$t(N) \sim N^{\alpha}, \qquad \alpha \leq 2 \text{ for multigrid, etc}$$

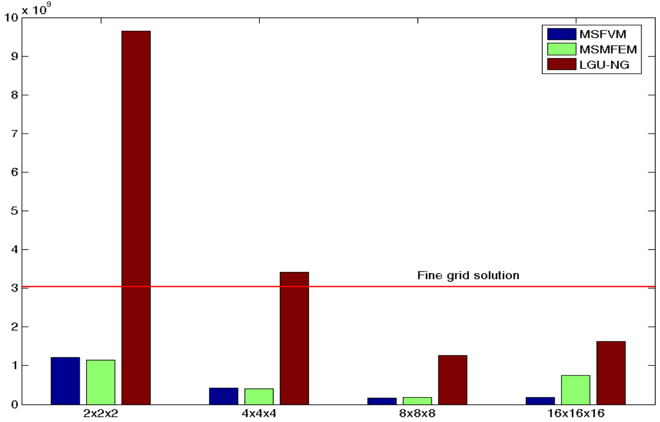
Example: Cartesian D-dimensional grid, N_c coarse blocks, each with N_s fine blocks, m iterations in LGU-NG.

Method	**Local**	**Global**
MsFVM	$(2^D + 1) \cdot N_c \cdot N_s^{\alpha} +$	$1 \cdot N_c^{\alpha}$
MsMFEM	$D \cdot 2^{\alpha} \cdot N_c \cdot N_s^{\alpha} +$	$D^{\alpha} \cdot N_c^{\alpha}$
LGU-NG	$(D + 1 + 2^{D \cdot \alpha} m) \cdot N_c \cdot N_s^{\alpha} +$	$D \cdot m \cdot N_c^{\alpha}$

Typical complexity as a function of coarse grid size:



Example: 3D (128x128x128), $\alpha = 1.5$ and $m = 3$



In practice†:

- Assembly time may become significant when solving many small problems since vectorization is harder.
- Efficient linear solvers typically require an initial setup phase, therefore the solution of many small systems may be more time-consuming than anticipated.

Note

The multiscale methods are not necessarily more efficient than direct solution of a single fine scale problem, but they allow solution of bigger problems, and for non-linear problems where local properties need to be recomputed only in limited regions they have a significant performance advantage.

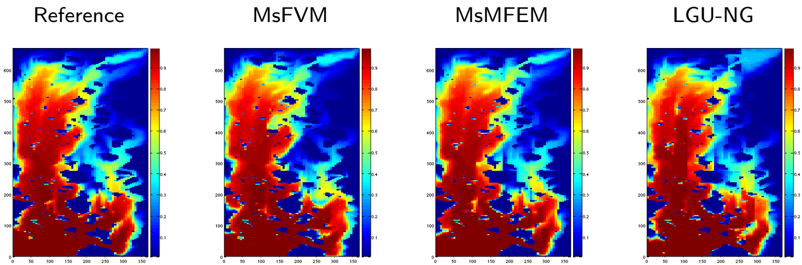
† At least for our fairly naive implementations.

A Few Words About Implementation...

In our experience:

- The really difficult part is the well model!
- MsFVM and LGU-NG: based upon *dual grid*
  $\longrightarrow$ special cases (along global boundaries and internal structures) that complicate the implementation
- MsMFEM: coarse grid = union of cells in fine grid
  $\longrightarrow$ problem with dual grid avoided
  Given a numerical method that works on the fine grid the implementation is straightforward. (Very few lines of MATLAB code for simple grids!)

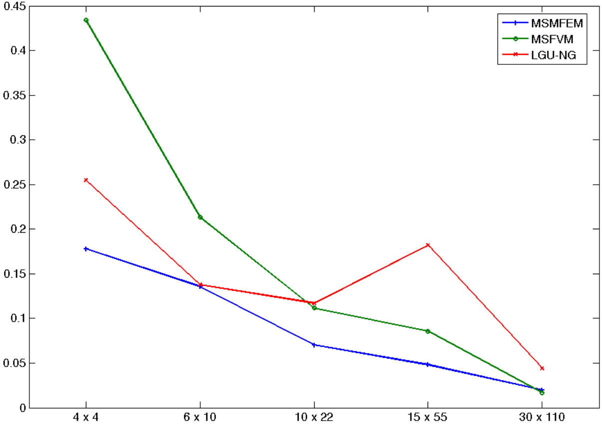
Example (Bottom layer from SPE10 comparative solution project)



Velocity fields may differ both locally and globally, but all methods produce qualitatively similar saturation profiles

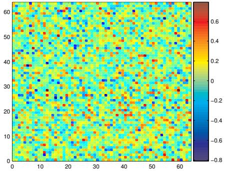
Fluvial Reservoirs (cont'd)

Error in saturation field as function of coarse grid size:



Example

Logarithm of k_x

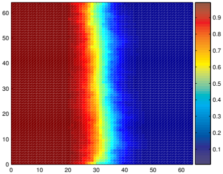


$$k_y/k_x = 10^4$$

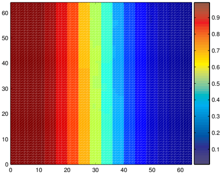
or

$$h_y/h_x = 10^{-2}$$

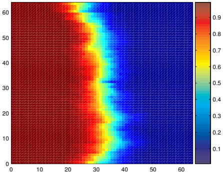
Reference



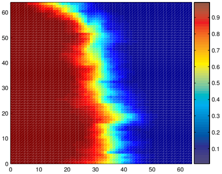
MsFVM



MsMFEM

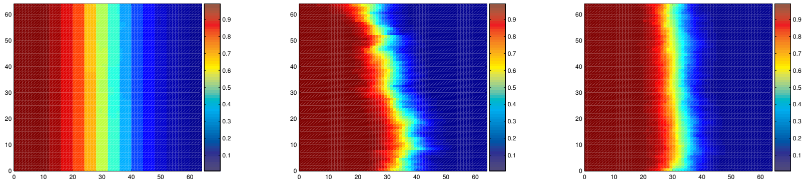


LGU-NG



Example (Anisotropic medium, revisited)

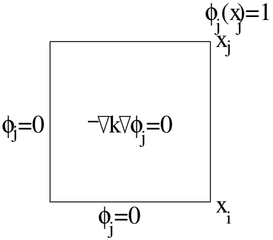
MsFVM | MsFVM – NG | MsFVM global BC



Instability issues for the MsFVM are reduced or eliminated by using global boundary conditions or by replacing the reconstruction procedure with nested gridding

An initial global fine scale pressure solution p_0 can be exploited to improve accuracy of two-phase flow simulations

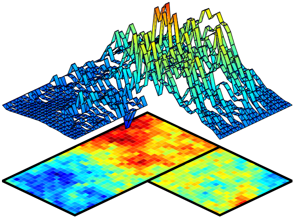
MsFVM:



$$\phi_j(x) = \frac{p_0(x) - p_0(x_j)}{p_0(x_i) - p_0(x_j)}$$

$$x \in [x_i, x_j].$$

MsMFEM

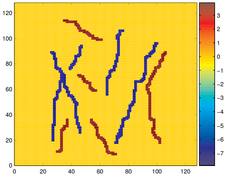


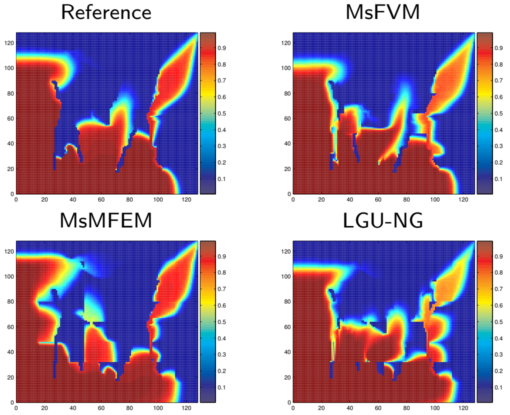
$$\psi_{ij} \cdot n_{ij} = \frac{k\nabla p_0 \cdot n_{ij}}{\int_{\partial K_i \cap \partial K_j} k\nabla p_0 \cdot n_{ij}\, ds}$$

$$x \in \partial K_i \cap \partial K_j.$$

Example

Logarithm of k_x



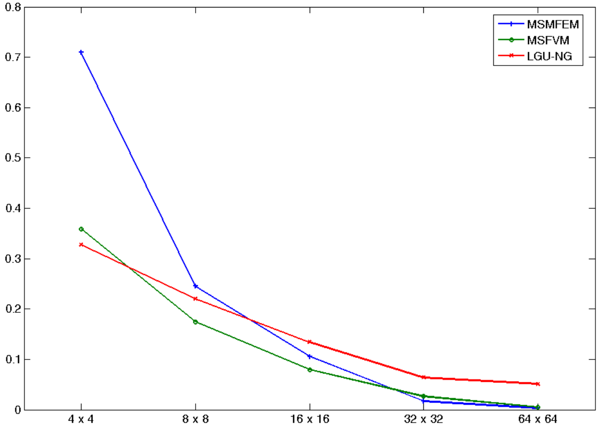


$$k_{\text{red}} = 10^4$$
$$k_{\text{yellow}} = 1$$
$$k_{\text{blue}} = 10^{-8}$$

Coarse grid $= 8 \times 8$.

Shale Barriers and High-Permeable Channels (cont'd)

Error in saturation field as function of coarse grid size:



MsMFEM: almost perfect solution by adaptive grid or global boundary conditions

Conclusions

- All three reasonably accurate on typical problems
- LGU-NG least efficient, convergence problems on barrier case, no apparent way of utilizing a fine-grid solution
- MsFVM most accurate on barrier case, completely off on the anisotropic case. Problem fixed by NG, but still less accurate than MsMFEM
- MsMFEM simpler to implement, no particular weakness (except possibly barrier case, as discussed by Krogstad), generally best of the three for random media.
- Global BC give nearly perfect results for MsFVM and MsMFEM