Multiscale Methods for Capturing Geological Heterogeneity

Stein Krogstad and Knut–Andreas Lie

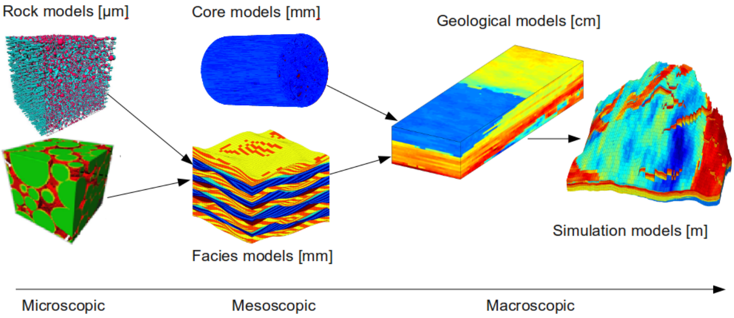
SINTEF ICT, Dept. Applied Mathematics

Rijswijk, May 3 2010

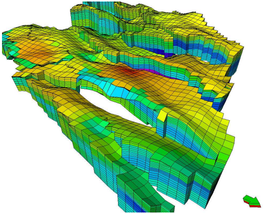
Physical Scales in Subsurface Modelling

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



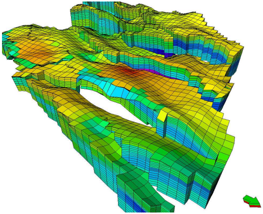
Expressing the geologists' preception of the reservoir:

- here: geo-cellular models
- describe the reservoir geometry (horizons, faults, etc)
- typically generated using geostatistics (or process simulation)
- give rock parameters (permeability and porosity)

Geological Models

Expressing the geologists' preception of the reservoir:

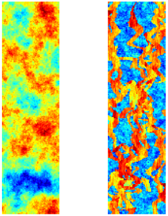
- here: geo-cellular models
- describe the reservoir geometry (horizons, faults, etc)
- typically generated using geostatistics (or process simulation)
- give rock parameters (permeability and porosity)



Rock parameters:

- have a multiscale structure
- details on all scales impact flow
- permeability spans many orders of magnitude

Ex: Brent sequence

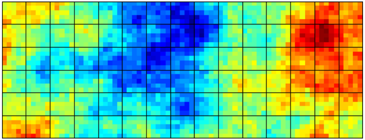


Tarbert Upper Ness

Heterogeneity versus Flow Modelling

Gap in resolution:

- Geomodels: $10^7 - 10^9$ cells
- Simulators: $10^5 - 10^6$ cells

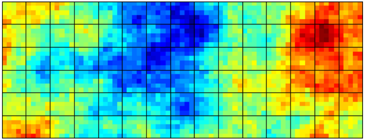


Heterogeneity versus Flow Modelling

Gap in resolution:

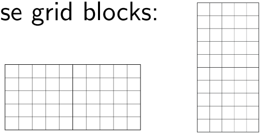
- Geomodels: $10^7 - 10^9$ cells
- Simulators: $10^5 - 10^6$ cells

$\longrightarrow$ sector models and/or
upscaling of parameters



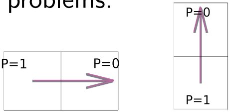
$\Downarrow$

Coarse grid blocks:



$\Downarrow$

Flow problems:

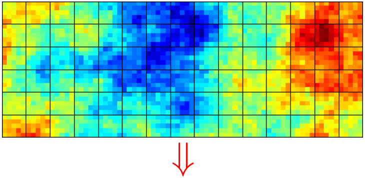


Heterogeneity versus Flow Modelling

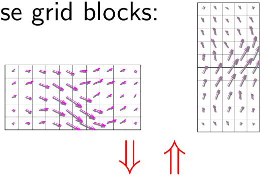
Gap in resolution:

- Geomodels: $10^7 - 10^9$ cells
- Simulators: $10^5 - 10^6$ cells

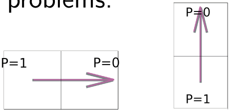
⟶ sector models and/or
upscaling of parameters



Coarse grid blocks:



Flow problems:

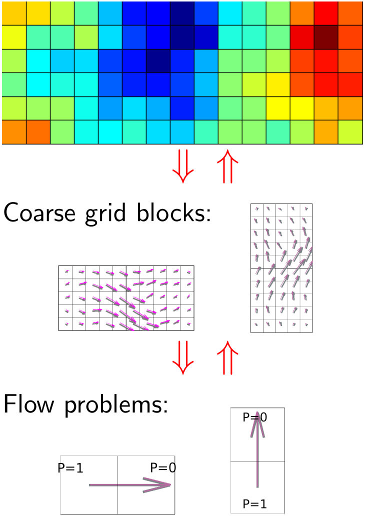


Heterogeneity versus Flow Modelling

Gap in resolution:

- Geomodels: $10^7 - 10^9$ cells
- Simulators: $10^5 - 10^6$ cells

$\longrightarrow$ sector models and/or
upscaling of parameters



Coarse grid blocks:

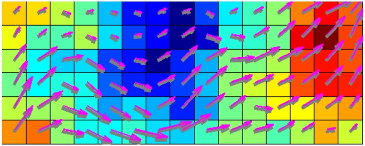
Flow problems:

Heterogeneity versus Flow Modelling

Gap in resolution:

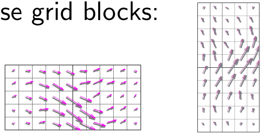
- Geomodels: $10^7 - 10^9$ cells
- Simulators: $10^5 - 10^6$ cells

$\longrightarrow$ sector models and/or
upscaling of parameters



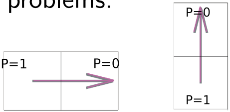
$\Downarrow \quad \Uparrow$

Coarse grid blocks:



$\Downarrow \quad \Uparrow$

Flow problems:



Heterogeneity versus Flow Modelling

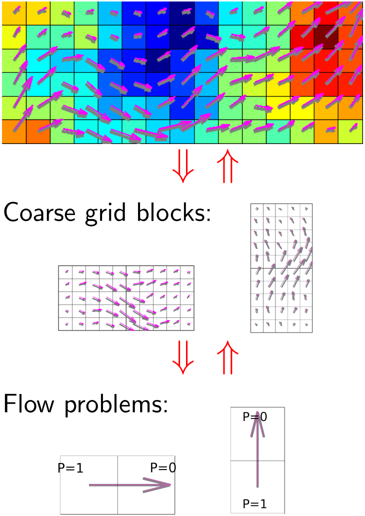
Gap in resolution:

- Geomodels: $10^7 - 10^9$ cells
- Simulators: $10^5 - 10^6$ cells

$\longrightarrow$ sector models and/or
upscaling of parameters

Many alternatives:

- Harmonic, arithmetic, geometric, ...
- Local methods (K or T)
- Global methods
- Local-global methods
- Pseudo methods
- Ensemble methods
- Steady-state methods



Simulation on Seismic/Geologic Grid

Why do we want/need it?

- Upscaling is a bottleneck in workflow,
- gives loss of information/accuracy,
- is not sufficiently robust,
- extensions to multiphase flow are somewhat shaky

Simulation on Seismic/Geologic Grid

Why do we want/need it?

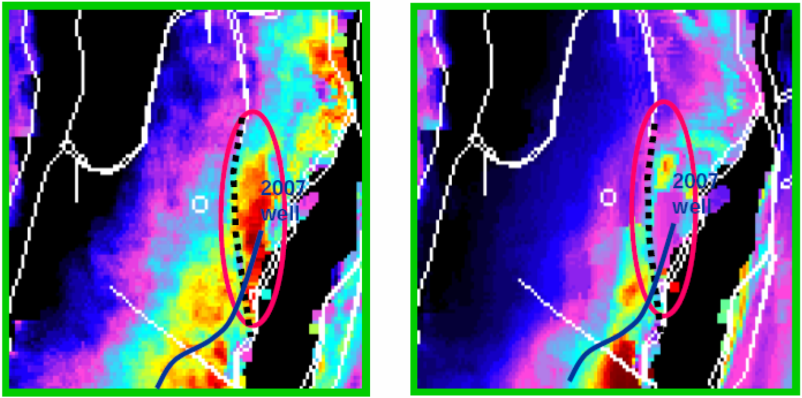
- Upscaling is a bottleneck in workflow,
- gives loss of information/accuracy,
- is not sufficiently robust,
- extensions to multiphase flow are somewhat shaky

Simulation on seismic/geologic grid:

- best possible resolution of the physical processes,
- faster model building and history matching,
- makes inversion a better instrument to find remaining oil,
- better estimation of uncertainty by running alternative models

Example: Gullfaks Field (North Sea)

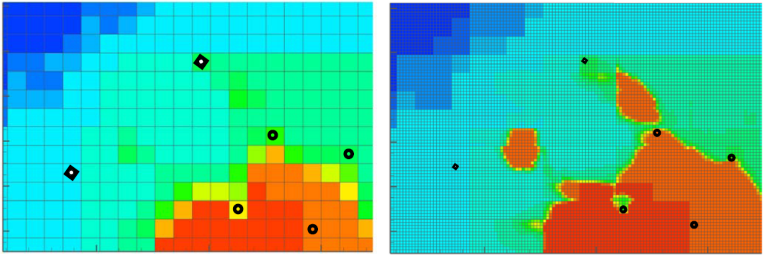
Bypassed oil (4D inversion vs simulation):



Arnesen, WPC, Madrid, 2008

Example: Giant Middle-East Field

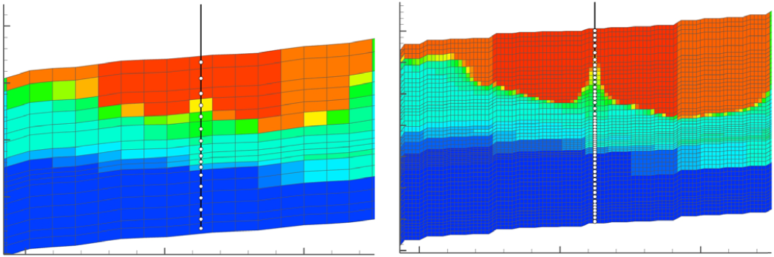
Difference in resolution (10 million vs 1 billion cells):



From Dogru et al., SPE 119272

Example: Giant Middle-East Field

Difference in resolution (10 million vs 1 billion cells):



From Dogru et al., SPE 119272

How to Close the Resolution Gap. . . ?

Simplified flow physics:

Can often tell a lot about the fluid movement. “Full physics” is typically only required towards the end of a workflow

How to Close the Resolution Gap. . . ?

Simplified flow physics:

Can often tell a lot about the fluid movement. “Full physics” is typically only required towards the end of a workflow

Operator splitting:

- fully coupled solution is slow..
- subequations often have different time scales
- splitting opens up for tailor-made methods

How to Close the Resolution Gap. . . ?

Simplified flow physics:

Can often tell a lot about the fluid movement. “Full physics” is typically only required towards the end of a workflow

Operator splitting:

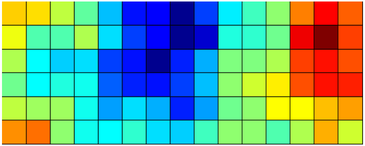
- fully coupled solution is slow..
- subequations often have different time scales
- splitting opens up for tailor-made methods

Use of sparsity / (multiscale) structure:

- effects resolved on different scales
- small changes from one step to next
- small changes from one simulation to next

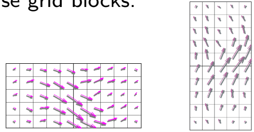
From Upscaling to Multiscale Pressure Solvers

Standard upscaling:



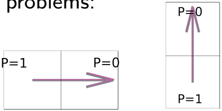
⇓⇑

Coarse grid blocks:

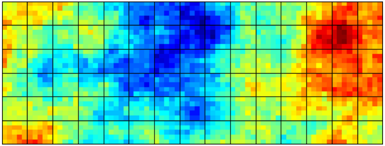


⇓⇑

Flow problems:

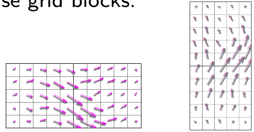


Multiscale method:



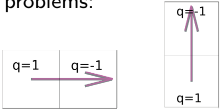
⇓

Coarse grid blocks:



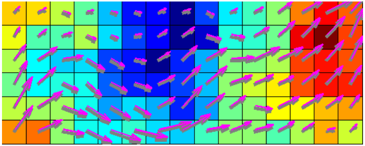
⇓⇑

Flow problems:



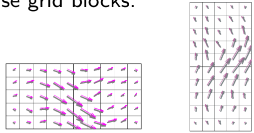
From Upscaling to Multiscale Pressure Solvers

Standard upscaling:



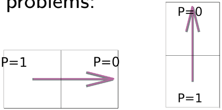
⇓⇑

Coarse grid blocks:

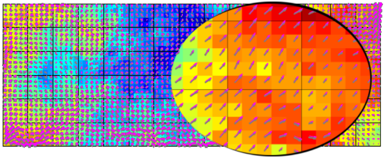


⇓⇑

Flow problems:

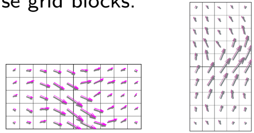


Multiscale method:



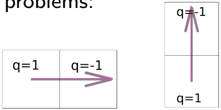
⇓ ⇑

Coarse grid blocks:



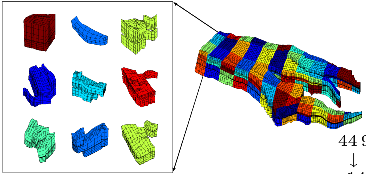
⇓⇑

Flow problems:



Workflow with Automated Upgridding in 3D

1) Automated coarsening: uniform partition in index space for corner-point grids

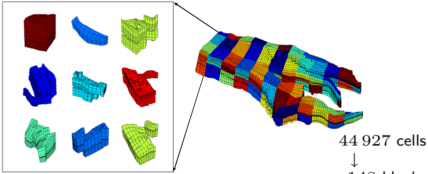


44 927 cells
↓
148 blocks

9 different coarse blocks

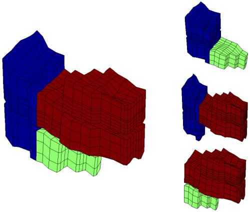
Workflow with Automated Upgridding in 3D

1) Automated coarsening: uniform partition in index space for corner-point grids



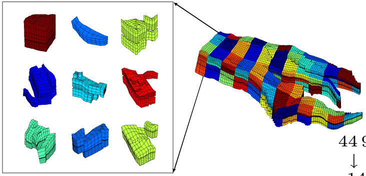
9 different coarse blocks

2) Detect all adjacent blocks



Workflow with Automated Upgridding in 3D

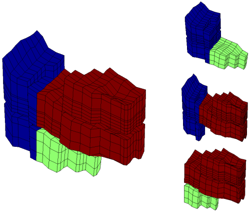
1) Automated coarsening: uniform partition in index space for corner-point grids



44 927 cells
↓
148 blocks

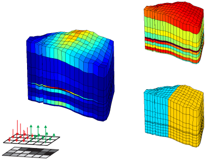
9 different coarse blocks

2) Detect all adjacent blocks



3) Compute basis functions

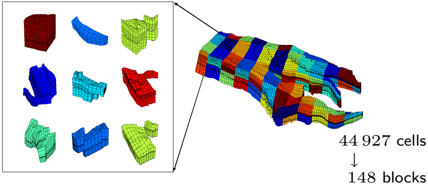
$$\nabla\cdot\psi_{ij} = \begin{cases} w_i(x), \\ -w_j(x), \end{cases}$$



for all pairs of blocks

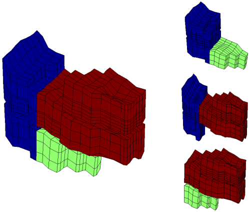
Workflow with Automated Upgridding in 3D

1) Automated coarsening: uniform partition in index space for corner-point grids



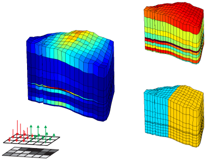
9 different coarse blocks

2) Detect all adjacent blocks



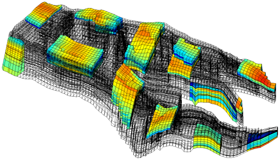
3) Compute basis functions

$$\nabla \cdot \psi_{ij} = \begin{cases} w_i(x), \\ -w_j(x), \end{cases}$$



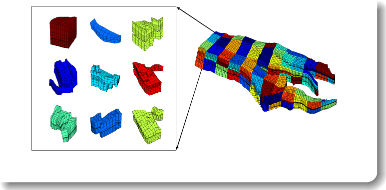
for all pairs of blocks

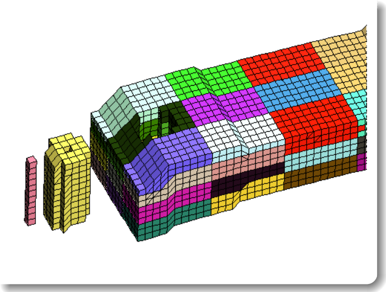
4) Block in coarse grid: component for building global solution



Multiscale Methods: Potential

- More flexible wrt grids than standard upscaling methods: automatic coarsening

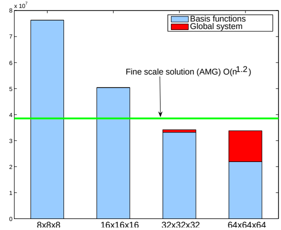




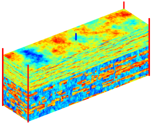
Multiscale Methods: Potential

- More flexible wrt grids than standard upscaling methods: automatic coarsening
- Reuse of computations, key to computational efficiency

Operations vs. upscaling factor:



SPE10: 1.1 mill cells

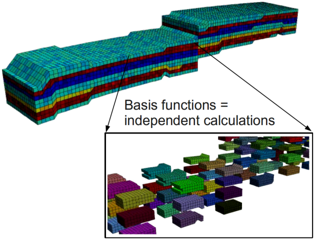


Inhouse code from 2005:

Multiscale: 2 min and 20 sec
Multigrid: 8 min and 36 sec

Multiscale Methods: Potential

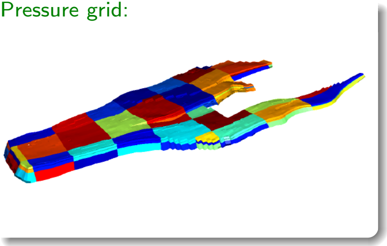
- More flexible wrt grids than standard upscaling methods: automatic coarsening
- Reuse of computations, key to computational efficiency
- Natural (elliptic) parallelism:
  - giga-cell simulations
  - multicore and heterogeneous computing


Basis functions = independent calculations

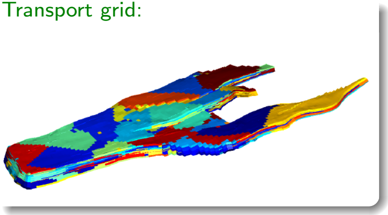
Multiscale Methods: Potential

- More flexible wrt grids than standard upscaling methods: automatic coarsening
- Reuse of computations, key to computational efficiency
- Natural (elliptic) parallelism:
  - giga-cell simulations
  - multicore and heterogeneous computing
- Fine-scale velocity ⟶ different grid for flow and transport ⟶ dynamical adaptivity

Pressure grid:



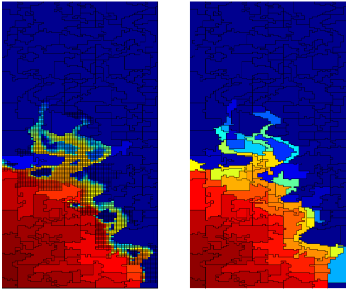
Transport grid:



Multiscale Methods: Potential

- More flexible wrt grids than standard upscaling methods: automatic coarsening
- Reuse of computations, key to computational efficiency
- Natural (elliptic) parallelism:
    - giga-cell simulations
    - multicore and heterogeneous computing
- Fine-scale velocity ⟶ different grid for flow and transport ⟶ dynamical adaptivity

Flow-based gridding:



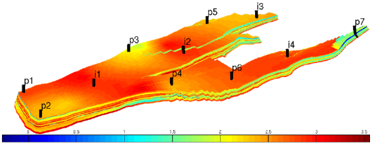
with and without dynamic Cartesian refinement

Research by: Vera Louise Hauge, Shell scholarship

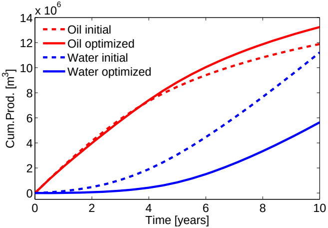
Multiscale Methods: Potential

- More flexible wrt grids than standard upscaling methods: automatic coarsening
- Reuse of computations, key to computational efficiency
- Natural (elliptic) parallelism:
  - giga-cell simulations
  - multicore and heterogeneous computing
- Fine-scale velocity ⟶ different grid for flow and transport ⟶ dynamical adaptivity
- Method for model reduction:
  - adjoint simulations ⟶ approximate gradients
  - ensemble simulations with representative basis functions

Water-flood optimization:



Reservoir geometry from a Norwegian Sea field

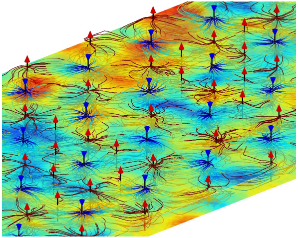


Forward simulations:
44 927 cells, 20 time steps, $<$ 5 sec in Matlab

Multiscale Methods: Potential

- More flexible wrt grids than standard upscaling methods: automatic coarsening
- Reuse of computations, key to computational efficiency
- Natural (elliptic) parallelism:
  - giga-cell simulations
  - multicore and heterogeneous computing
- Fine-scale velocity ⟶ different grid for flow and transport ⟶ dynamical adaptivity
- Method for model reduction:
  - adjoint simulations ⟶ approximate gradients
  - ensemble simulations with representative basis functions

History matching 1 million cells:



7 years: 32 injectors, 69 producers

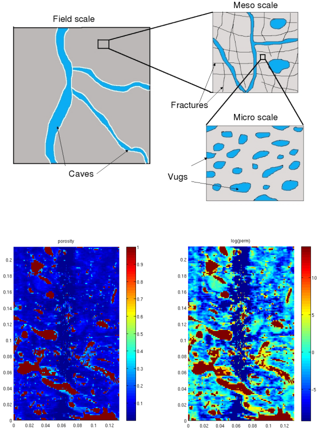
Generalized travel-time inversion + multiscale:
7 forward simulations, 6 inversions

	CPU-time (wall clock)		
Solver	Total	Pres.	Transp.
Multigrid	39 min	30 min	5 min
Multiscale	17 min	7 min	6 min

Multiscale Methods: Potential

- More flexible wrt grids than standard upscaling methods: automatic coarsening
- Reuse of computations, key to computational efficiency
- Natural (elliptic) parallelism:
    - giga-cell simulations
    - multicore and heterogeneous computing
- Fine-scale velocity ⟶ different grid for flow and transport ⟶ dynamical adaptivity
- Method for model reduction:
    - adjoint simulations ⟶ approximate gradients
    - ensemble simulations with representative basis functions
- Multiphysics applications

Stokes–Brinkmann:



Multiscale Methods: Limitations

How well do these methods handle complex physics?

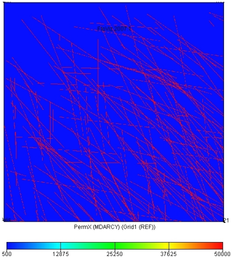
- No fully-implicit formulation available
- Compressibility, gravity, . . . ⟶ correction functions
- Strong coupling ⟶ more iterations and updates of basis and correction functions
- To force residual to zero, multiscale methods start to look like multigrid/domain decomposition
- Not yet applied to compositional/thermal/...

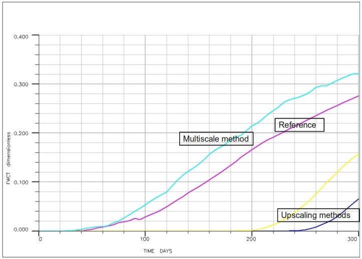
Multiscale Methods: Limitations

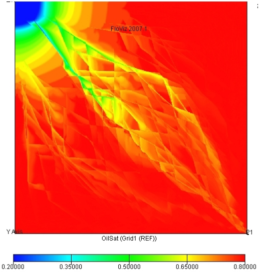
Other issues:

- How to choose good coarse grids for unstructured grids?
- Need for global information or iterative procedures?
- A posteriori error analysis (resolution or fine-scale junk)?
- More than two levels in hierarchical grid?
- How to include pore-scale models?

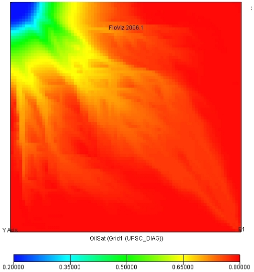
Example: Fracture Corridors



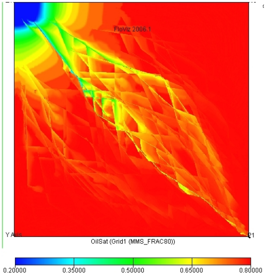




800 × 800

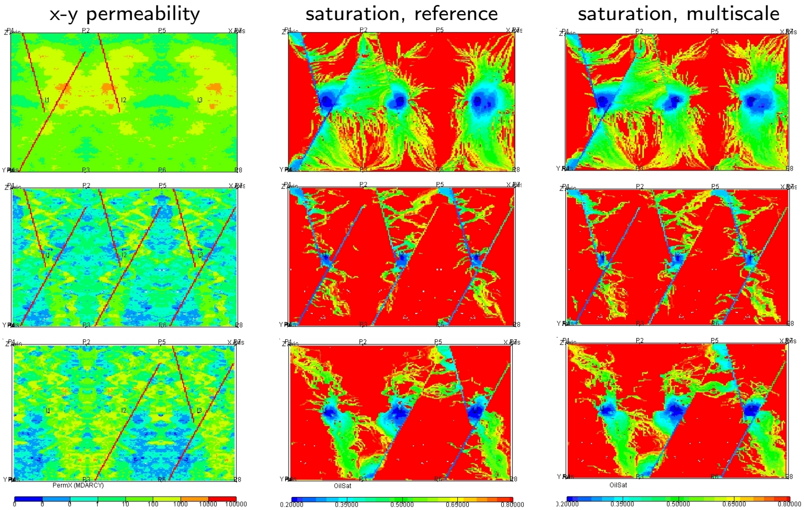


80 × 80 upscaled



80 × 80 multiscale

Example: SPE10 with Fracture Corridors

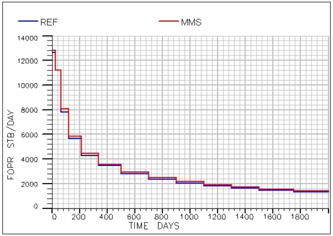


x-y permeability

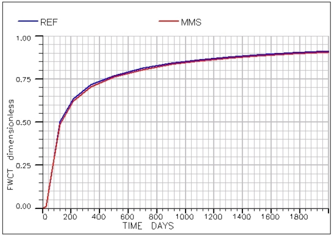
saturation, reference

saturation, multiscale

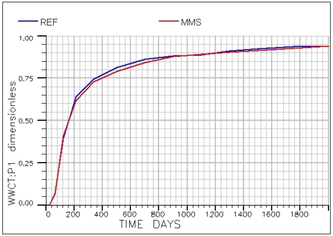
Example: SPE10 with Fracture Corridors



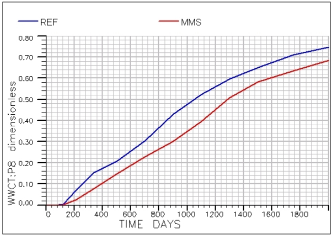
field oil-production rate



field water cut



best well: water cut in P1

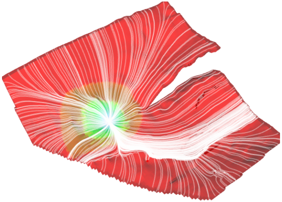


worst well: water cut in P8

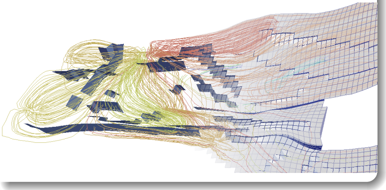
Efficient Transport Solvers

- Streamline methods
    - intuitive visualization + new data
    - subscale resolution
    - good scaling, known to be efficient

Flow pattern (CO2 injection):



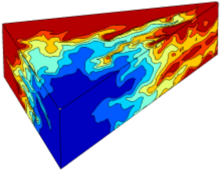
Connections across faults:



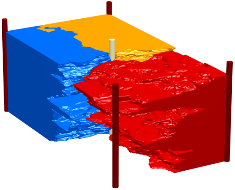
Efficient Transport Solvers

- Streamline methods
  - intuitive visualization + new data
  - subscale resolution
  - good scaling, known to be efficient

Time-of-flight (timelines):



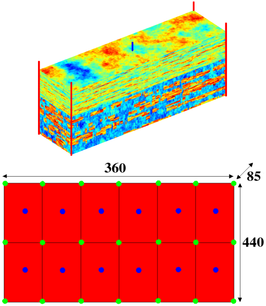
Flooded volumes (stationary tracer):



Efficient Transport Solvers

- Streamline methods
  - intuitive visualization + new data
  - subscale resolution
  - good scaling, known to be efficient

Synthetic example in FrontSim:

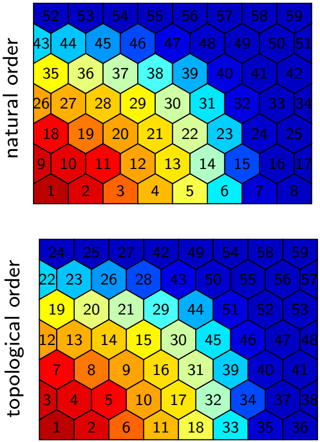


- 13.5 million cells
- Intel Xeon, 64 GiB, 3.2 GHz
- Single thread, 13.5 GiB RAM
- Runtime: 1 h 55 min

Efficient Transport Solvers

- Streamline methods
  - intuitive visualization + new data
  - subscale resolution
  - good scaling, known to be efficient
- Optimal ordering
  - same assumptions as for streamlines
  - utilize causality $\longrightarrow \mathcal{O}(n)$ algorithm, cell-by-cell solution
  - local control over (non)linear iterations

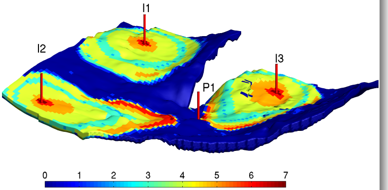
Topological sorting



Efficient Transport Solvers

- Streamline methods
  - intuitive visualization + new data
  - subscale resolution
  - good scaling, known to be efficient
- Optimal ordering
  - same assumptions as for streamlines
  - utilize causality $\longrightarrow \mathcal{O}(n)$ algorithm, cell-by-cell solution
  - local control over (non)linear iterations

Local iterations:



Johansen formation: 27 437 active cells

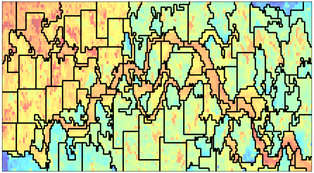
Global vs local Newton–Raphson solver

Δt	global		local	
days	time	iter	time (sec)	iter
125	2.26	12.69	0.044	0.93
250	2.35	12.62	0.047	1.10
500	2.38	13.25	0.042	1.41
1000	2.50	13.50	0.042	1.99

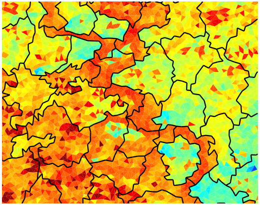
Efficient Transport Solvers

- Streamline methods
  - intuitive visualization + new data
  - subscale resolution
  - good scaling, known to be efficient
- Optimal ordering
  - same assumptions as for streamlines
  - utilize causality $\longrightarrow \mathcal{O}(n)$ algorithm, cell-by-cell solution
  - local control over (non)linear iterations
- Flow-based coarsening
  - agglomeration of cells $\longrightarrow$ simple and flexible coarsening
  - hybrid griding schemes
  - heterogeneous multiscale method?
  - efficient model reduction

Cartesian grid:



Triangular grids:

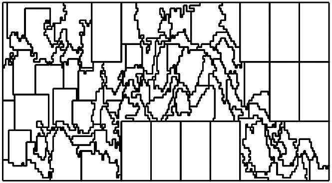


Efficient Transport Solvers

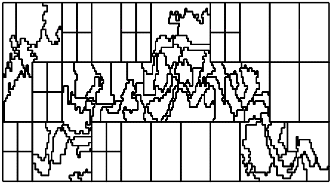
- Streamline methods
  - intuitive visualization + new data
  - subscale resolution
  - good scaling, known to be efficient
- Optimal ordering
  - same assumptions as for streamlines
  - utilize causality $\longrightarrow \mathcal{O}(n)$ algorithm, cell-by-cell solution
  - local control over (non)linear iterations
- Flow-based coarsening
  - agglomeration of cells $\longrightarrow$ simple and flexible coarsening
  - hybrid griding schemes
  - heterogeneous multiscale method?
  - efficient model reduction

Different partitioning:

Uniform coarsening + NUC refinement



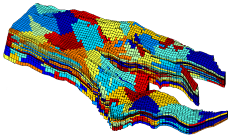
Uniform coarsening + Cartesian/NUC refinement

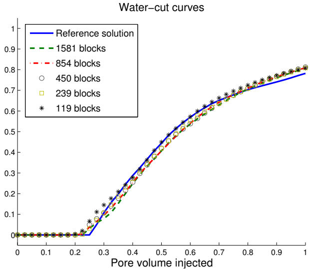


Efficient Transport Solvers

- Streamline methods
  - intuitive visualization + new data
  - subscale resolution
  - good scaling, known to be efficient
- Optimal ordering
  - same assumptions as for streamlines
  - utilize causality $\longrightarrow \mathcal{O}(n)$ algorithm, cell-by-cell solution
  - local control over (non)linear iterations
- Flow-based coarsening
  - agglomeration of cells $\longrightarrow$ simple and flexible coarsening
  - hybrid griding schemes
  - heterogeneous multiscale method?
  - efficient model reduction

Model reduction by coarsening:





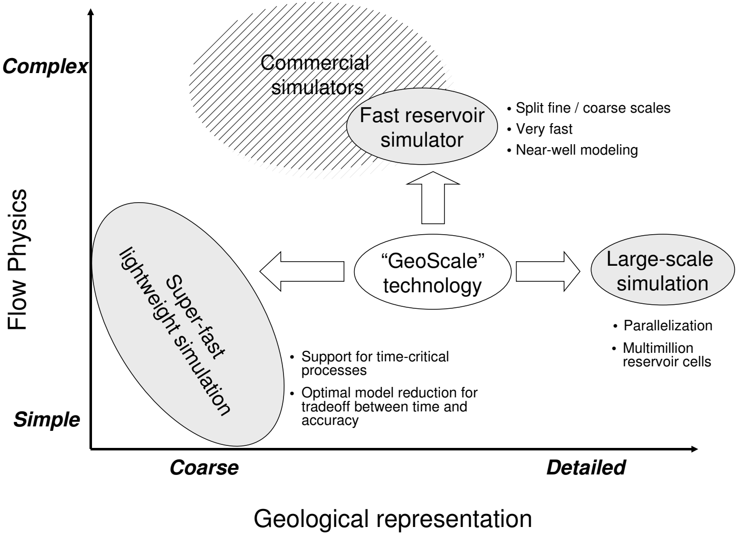
Summary

Keys to enable fast simulation on seismic/geological grids:

- Simplified physics
- Operator splitting
- Sparsity / (multiscale) structure

In the future: fit-for-purpose rather than one-simulator-solves-all..?

Current and Future Research



- Split fine / coarse scales
- Very fast
- Near-well modeling

- Support for time-critical processes
- Optimal model reduction for tradeoff between time and accuracy

- Parallelization
- Multimillion reservoir cells