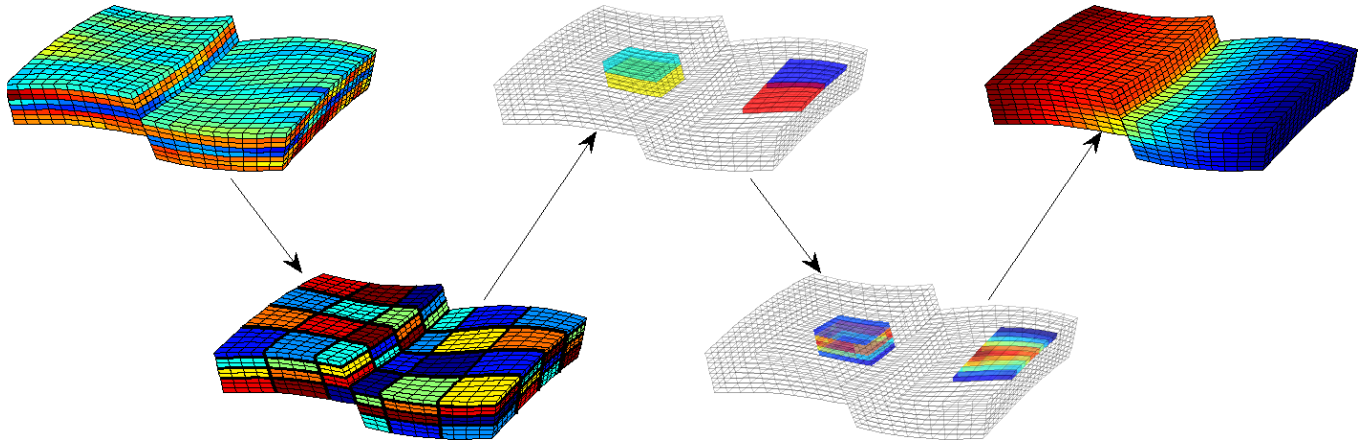
Upgridding by Amalgamation: Flow-Adapted Grids for Multiscale Simulations

Knut–Andreas Lie and Jostein R. Natvig, SINTEF, Norway



SIAM Conference on
Mathematical and Computational Issues in the Geosciences
Long Beach, CA, March 21–24, 2011

What is multiscale simulation?

Generally:

Methods that incorporate fine-scale information into a set of coarse scale equations in a way which is consistent with the local property of the differential operator

Herein:

Multiscale pressure solver (upscaling + downscaling in one step)

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

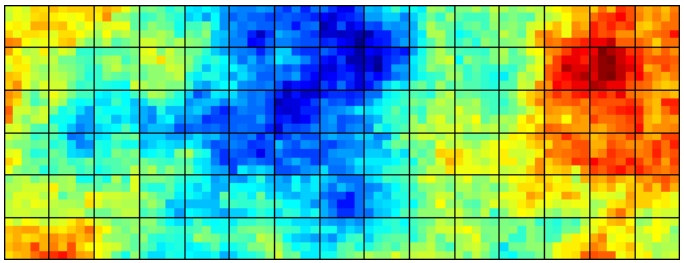
\+ Transport solver (on fine, intermediate, or coarse grid)

$$\phi \frac{\partial S}{\partial t} + \nabla\Big(\vec{v} f(S)\Big) = q$$

= Multiscale simulation of models with higher detail

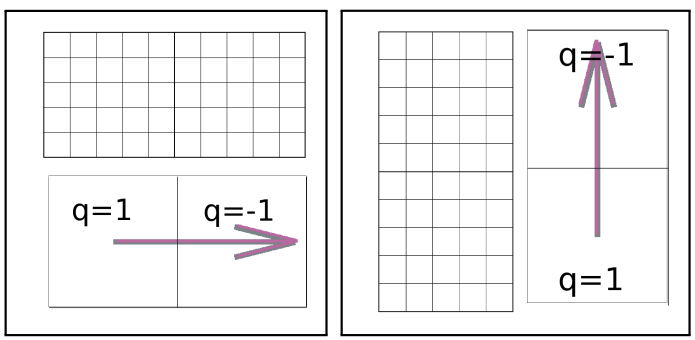
What is multiscale simulation?

Coarse partitioning:



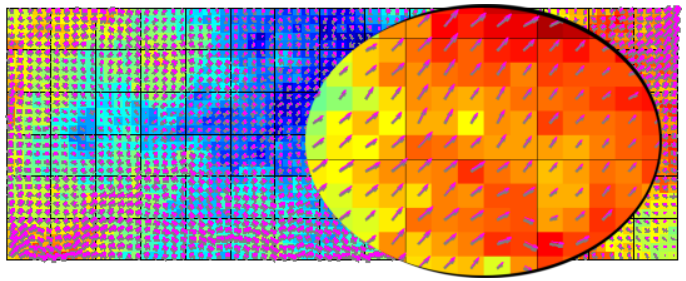
$\Downarrow$

Local flow problems:



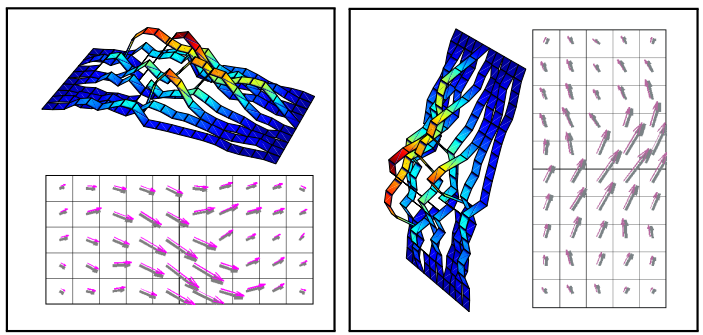
$\Rightarrow$

Flow field with subresolution:



$\Uparrow$

Flow solutions $\longrightarrow$ basis functions:



Flow-adapted coarse grids

Goal:

Given the ability to model velocity on geomodels and transport on coarse grids: Find a suitable coarse grid that best resolves fluid transport and minimizes loss of accuracy.

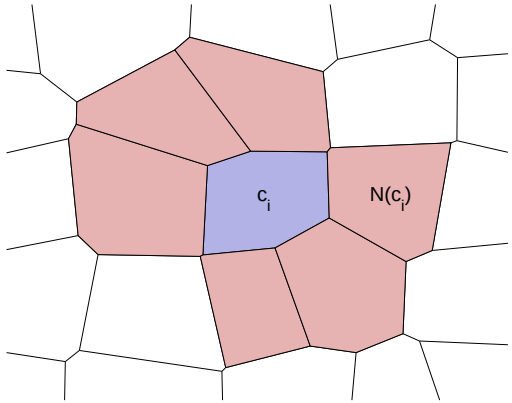
Formulated as the minimization of two measures:

1. the *projection error* between fine and coarse grid
2. the *evolution error* on the coarse grid

Flow-adapted coarsening

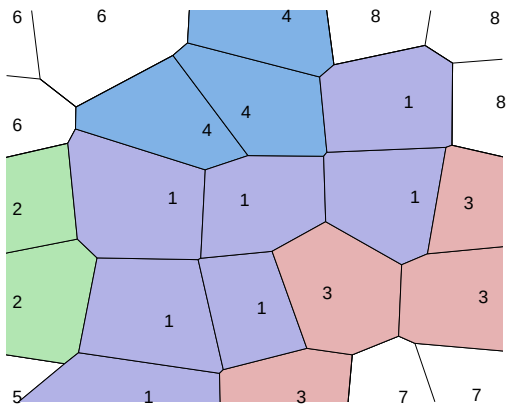
Assumptions:

- a matching polyhedral grid with n cells c_i,
- a mapping $\mathcal{N}(c)$ between cell c and its nearest neighbours,
- a set of flow indicators $I(c_i)$ in cell c_i



We seek a coarse grid that:

- adapts to the flow pattern predicted by indicator I,
- is formed by grouping cells into N blocks B_ℓ,
- is described by a partition vector p with n elements, in which element p_i assumes the value ℓ if cell c_i is member of block B_ℓ.



Coarsening principles

- Minimize heterogeneity of flow field inside each block

$$\min_{B_j}\Big(\sum_{p_i=j}|I_1(c_i)-I_1(B_j)|^p\,|c_i|\Big)^{\frac{1}{p}},\qquad 1\le p\le\infty,$$

- Equilibrate indicator values over grid blocks

$$\min\Big(\sum_{j=1}^{N}|I_2(B_j)-\bar{I}_2(\Omega)|^p\,|B_j|\Big)^{\frac{1}{p}},\qquad 1\le p\le\infty,$$

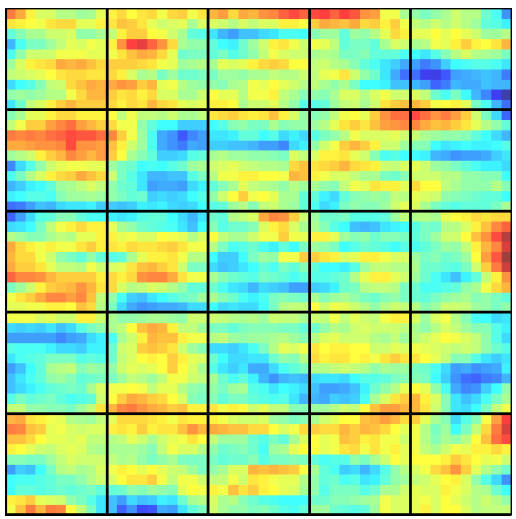
- Keep block sizes within prescribed lower and upper bounds

Amalgamation algorithm

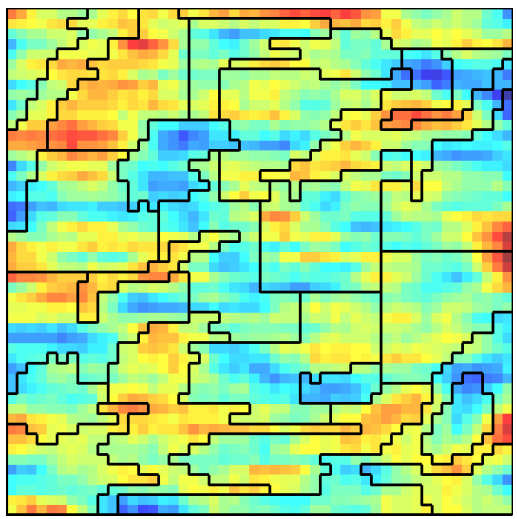
Amalgamation of cells:

- Difficult to formulate a practical and well-posed minimization problem for optimal coarsening $\longrightarrow$ ad hoc algorithms
- Coarsening process steered by a set of admissible and feasible amalgamation directions

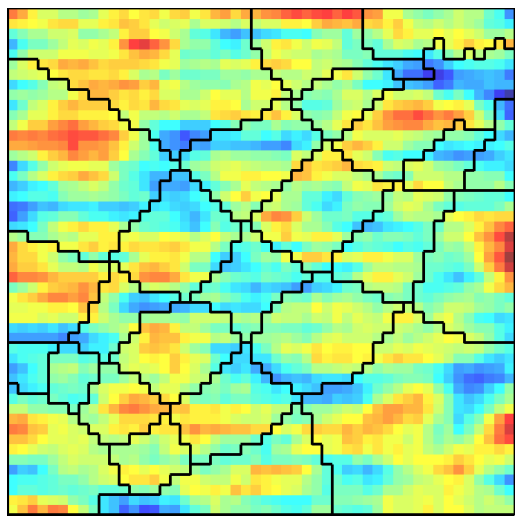
50×50 **lognormal permeability:**



regular: 25 blocks



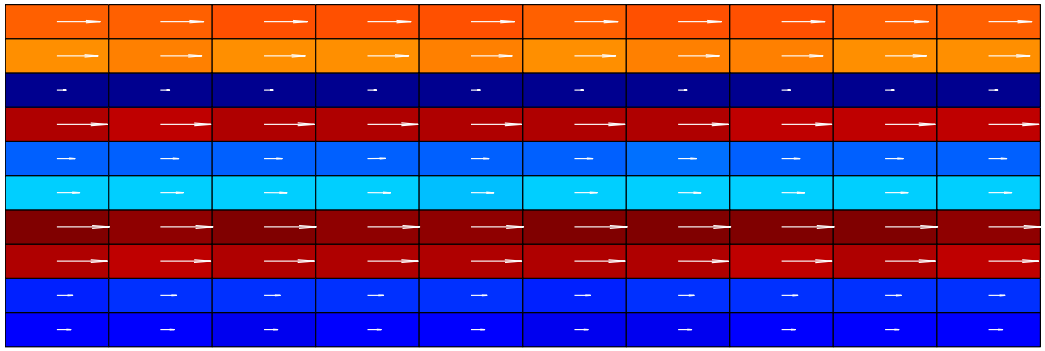
flow magnitude: 26 blocks



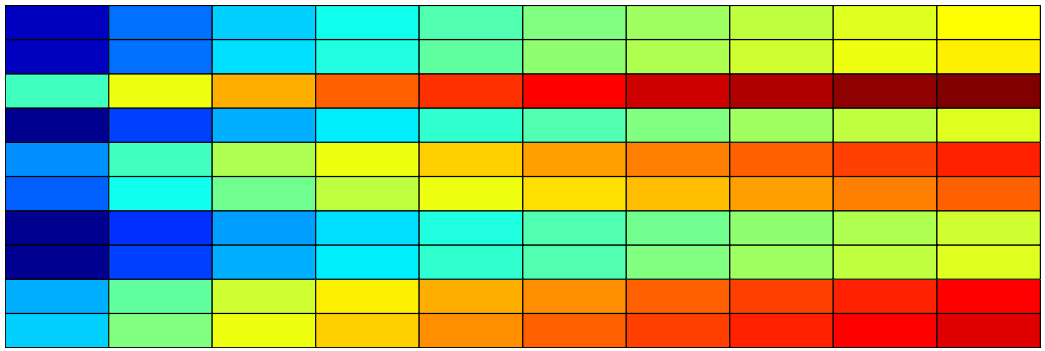
isocontours [p]: 26 blocks

Motiviation: layered reservoir

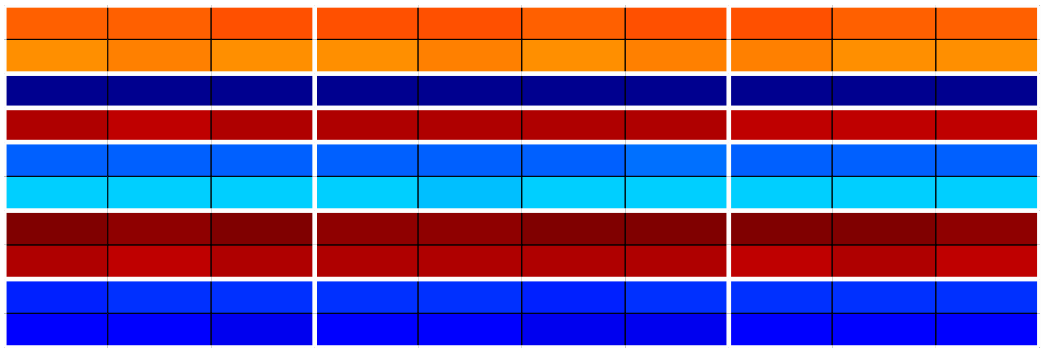
Permeability and velocity



Time−of−flight



Partition



Heuristic minimization: algorithmic components

Formulated using a set of:

sources that create a partition vector based upon grid topology, geometry, flow-based indicator functions, error estimates, or expert knowledge supplied by the user, thereby introducing the feasible amalgamation directions

`filters` that take a set of partition vectors as input and create a new partition as output, by

- combining/intersecting different partitions
- performing sanity checks, ensuring connected partitions according to admissible directions, etc
- modifying partition by merging small blocks or splitting large blocks

Heuristic minimization: algorithmic components

Partition:

- prescribed topology or predefined shapes
- segmentation of cells c_i into bins $\tilde{B}_\ell$

$$c_i \subset \tilde{B}_\ell \text{ if } I(c_i) \in [\ell, \ell + 1).$$

assuming indicator function I scaled to the interval $[1, M + 1]$

Intersection:

- intersect two or more partitions to produce a new partition
- split multiply connected blocks into sets of singly connected cells

Heuristic minimization: algorithmic components

Merging:

If block B violates the condition

$$I(B)\,|B| \geq \frac{N_L}{n}\,\bar{I}(\Omega)\,|\Omega|,$$

for a prescribed constant N_L, the block is merged with the neighbouring block B' that has the closest indicator value, i.e.,

$$B' = \operatorname{argmin}_{B'' \subset \mathcal{N}(B)} |I(B) - I(B'')|.$$

Refinement:

Refine blocks B that violate the condition

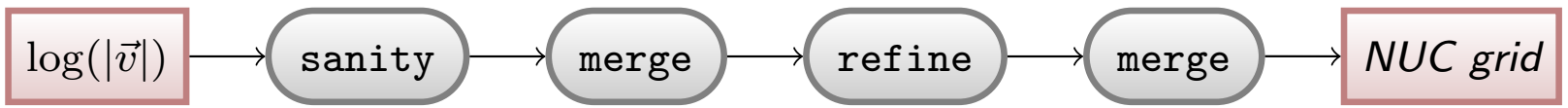
$$I(B)\,|B| \leq \frac{N_U}{n}\,\bar{I}(\Omega)\,|\Omega|,$$

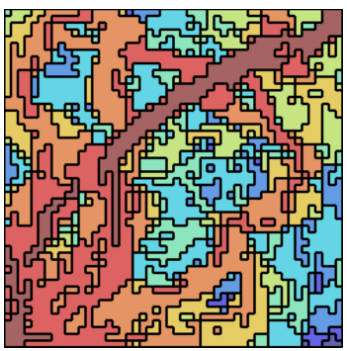
for a prescribed constant N_U

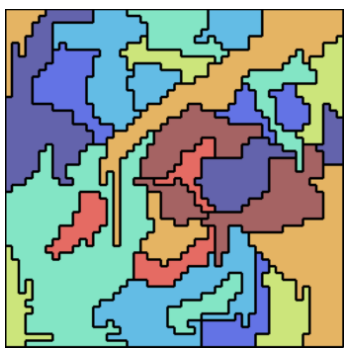
Example: non-uniform coarsening

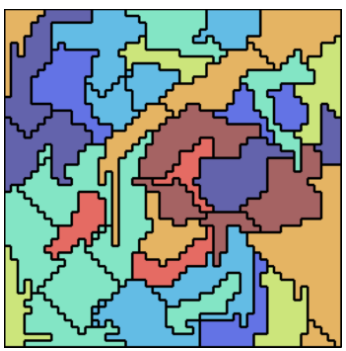
Aarnes, Efendiev & Hauge (2007):

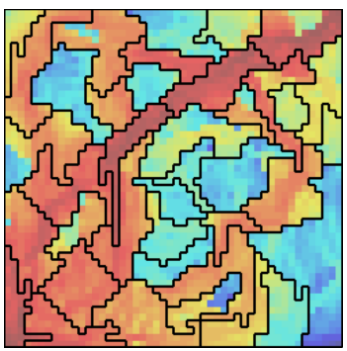
Use flow velocities to make a nonuniform grid in which each coarse block admits approximately the same total flow.



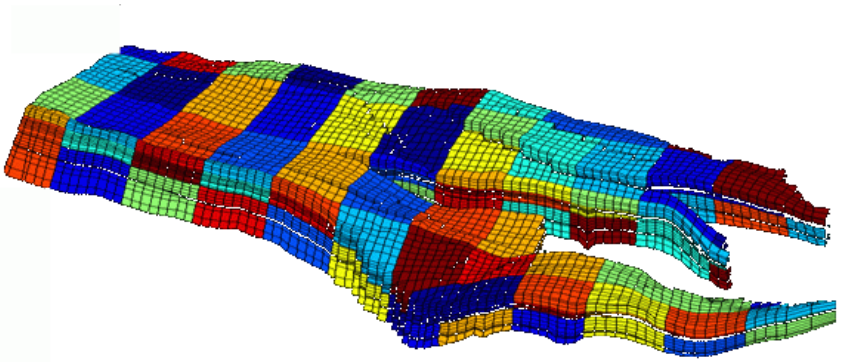

Partition: 304 blocks

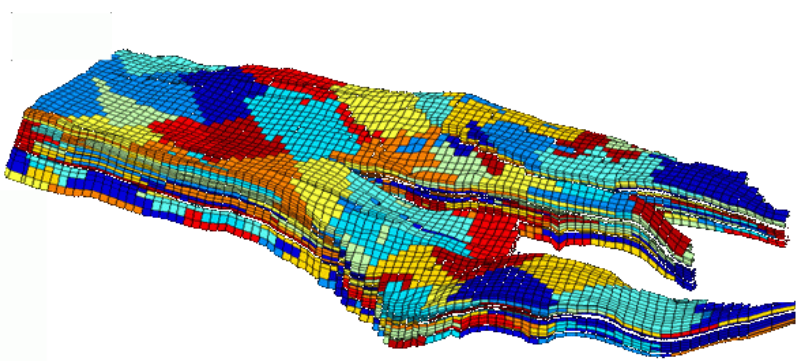

Merging: 29 blocks

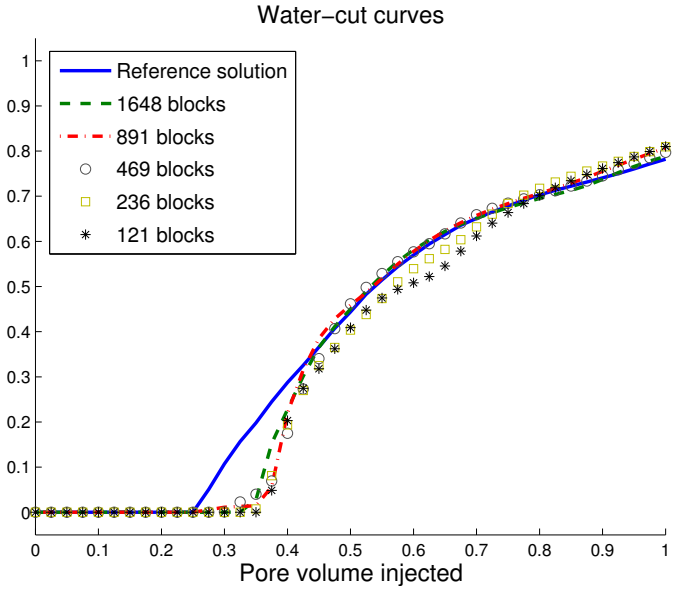

Refinement: 47 blocks

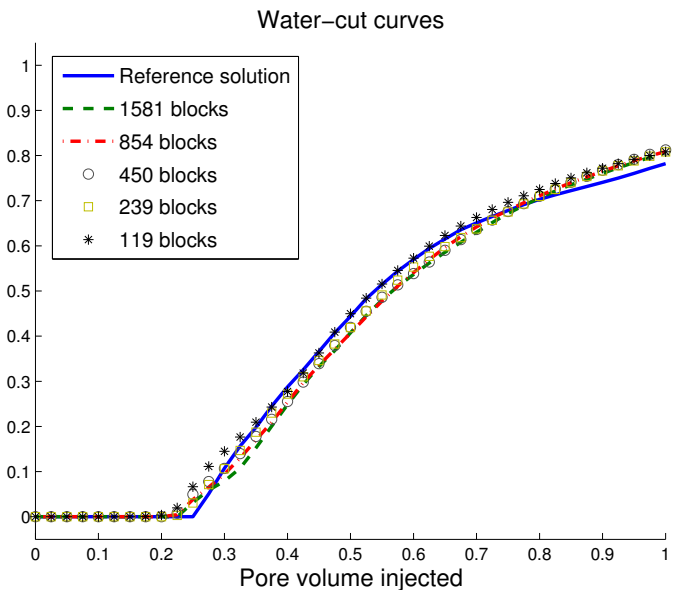

Merging: 39 blocks

Example: non-uniform coarsening

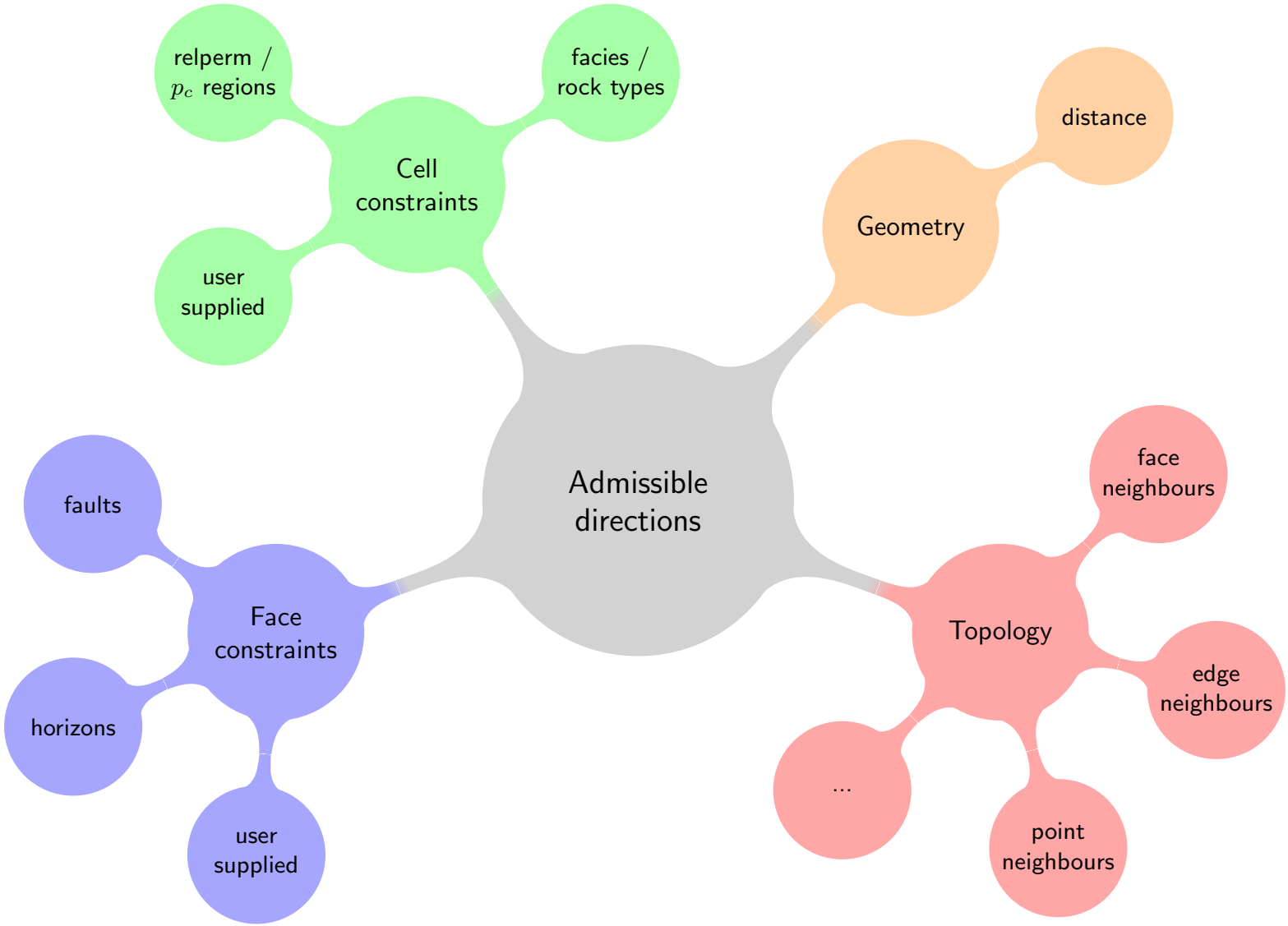




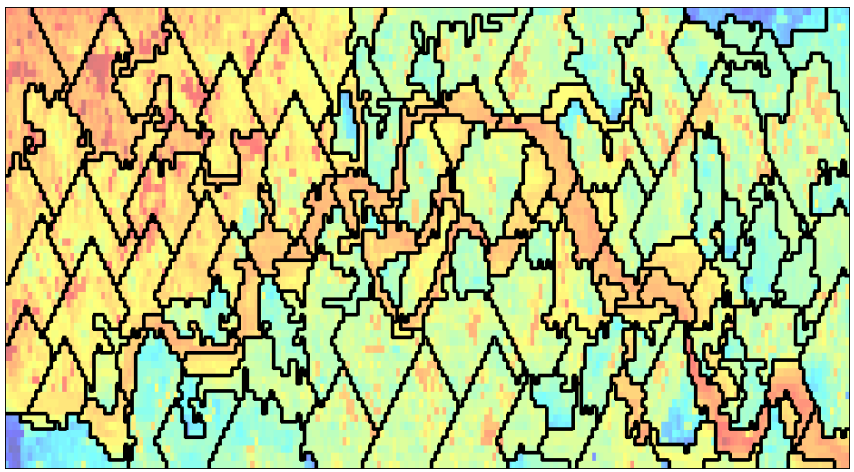


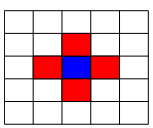


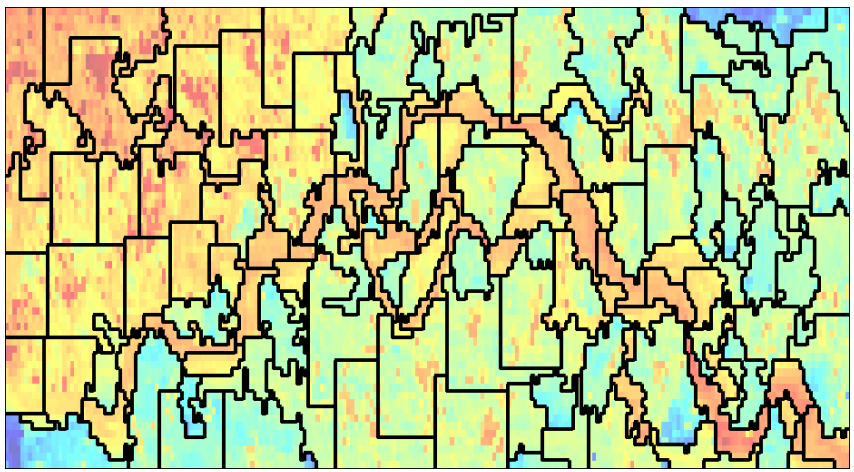
Amalgamation: admissible directions (neighbourship)

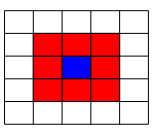


Amalgamation: extended neighbourship (topology)

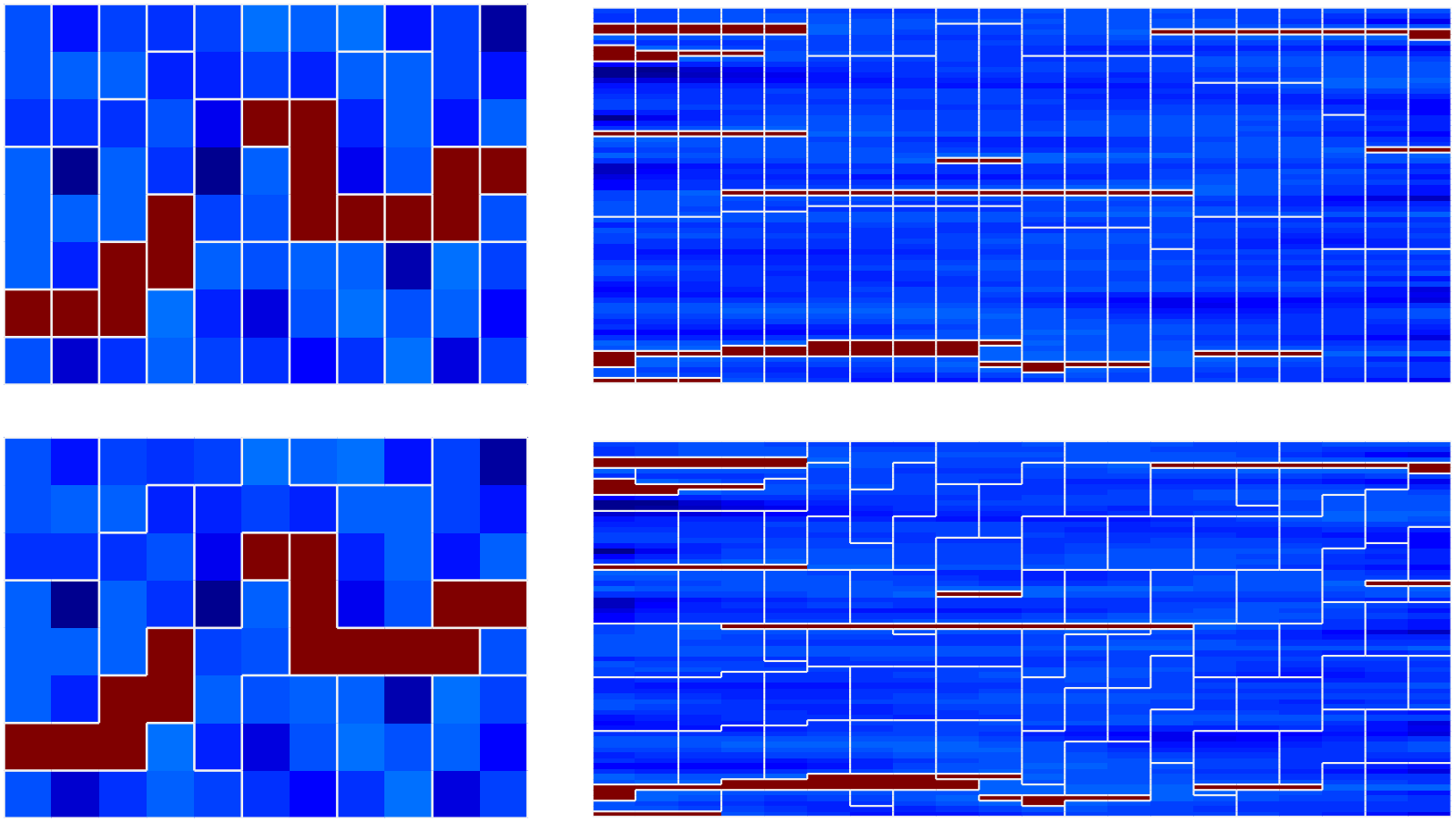


5-neighborhood 



9-neighborhood 

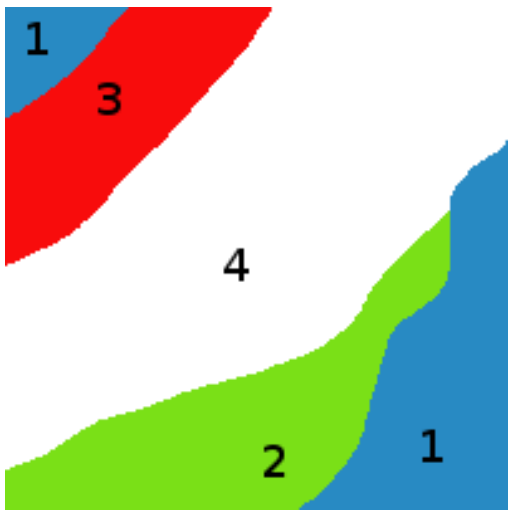
Amalgamation: restricted neighbourship (topology)



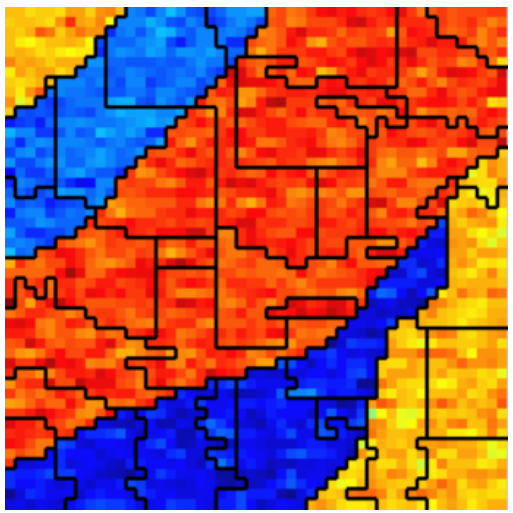
Upper row: $\mathcal{N}(c_{ij}) = \{c_{i,j\pm1}\}$

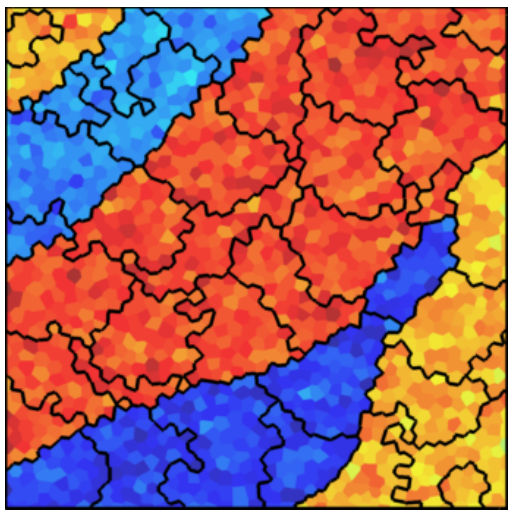
Lower row: $\mathcal{N}(c_{ij}) = \{c_{i,j\pm1}, c_{i\pm1,j}, c_{i\pm1,j\pm1}\}$

Amalgamation: restricted neighbourship (facies)



Facies distribution

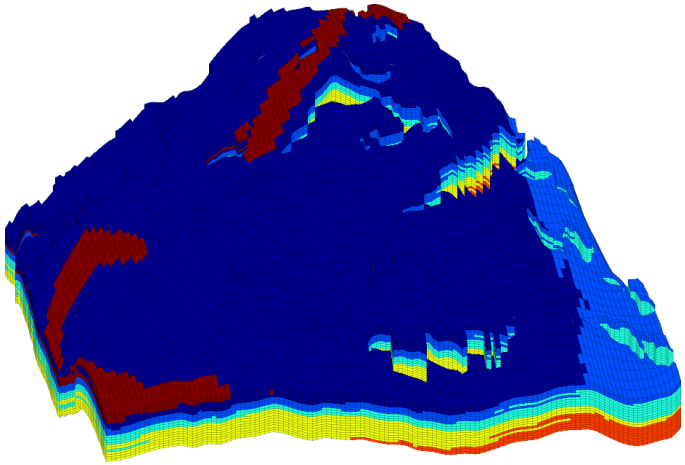

Cartesian


PEBI

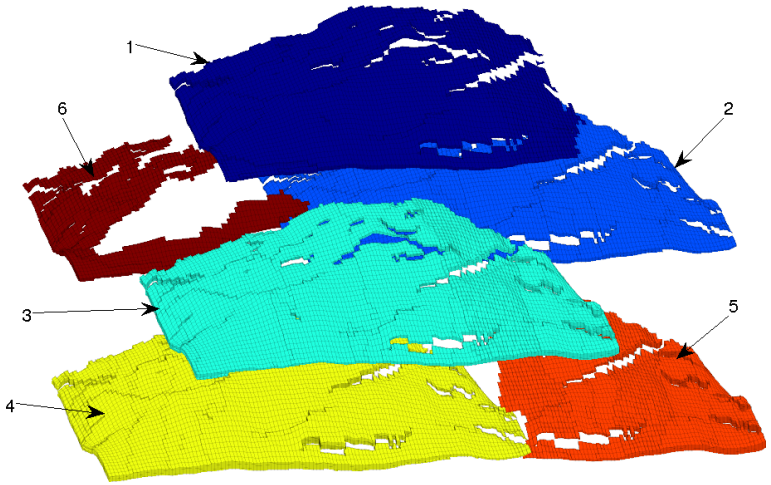
Constraining to facies / saturation regions:

- useful to preserve heterogeneity
- useful to avoid upscaling k_r and p_c curves

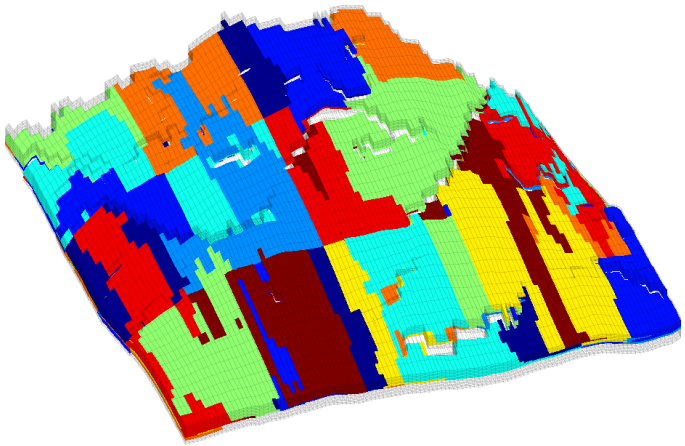
Amalgamation: restricted neighbourship (satnum)



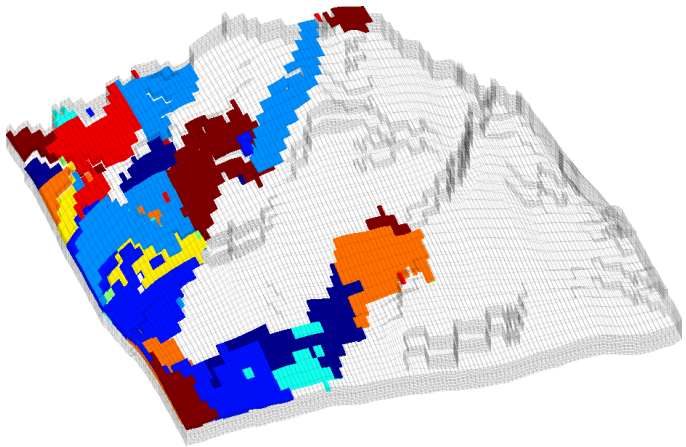
saturation regions



regions separated



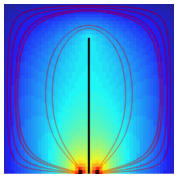
region # 3



region #6

Realization from SAIGUP study, coarsening within six different saturation regions

Amalgamation: restricted neighbourship (faults)

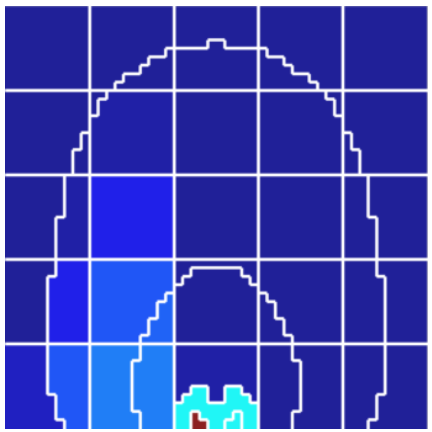


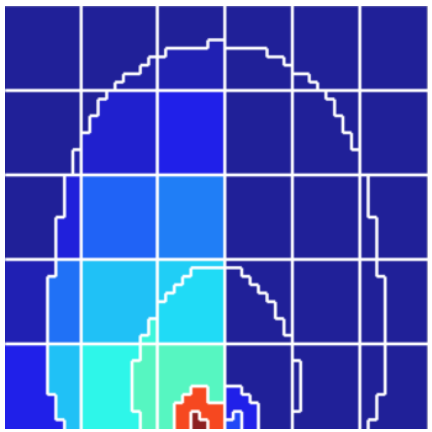
5×5 partition

6×5 partition

water-cut curves

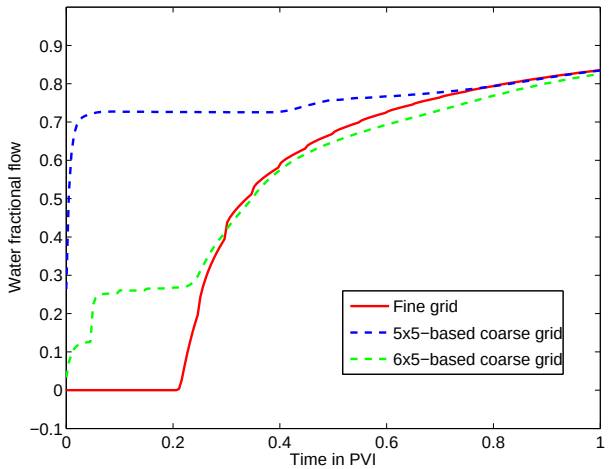
unconstrained



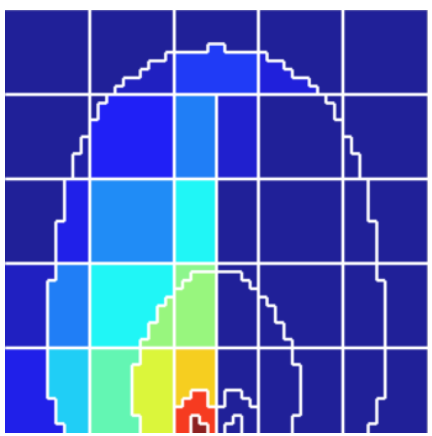


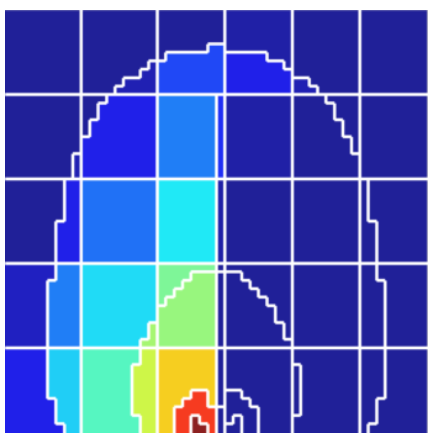
46 blocks

52 blocks



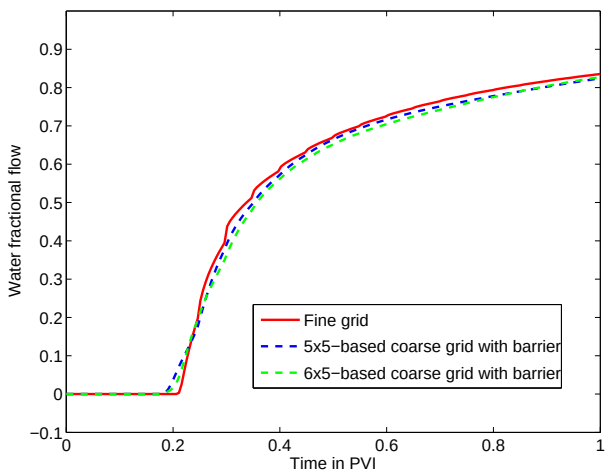
constrained



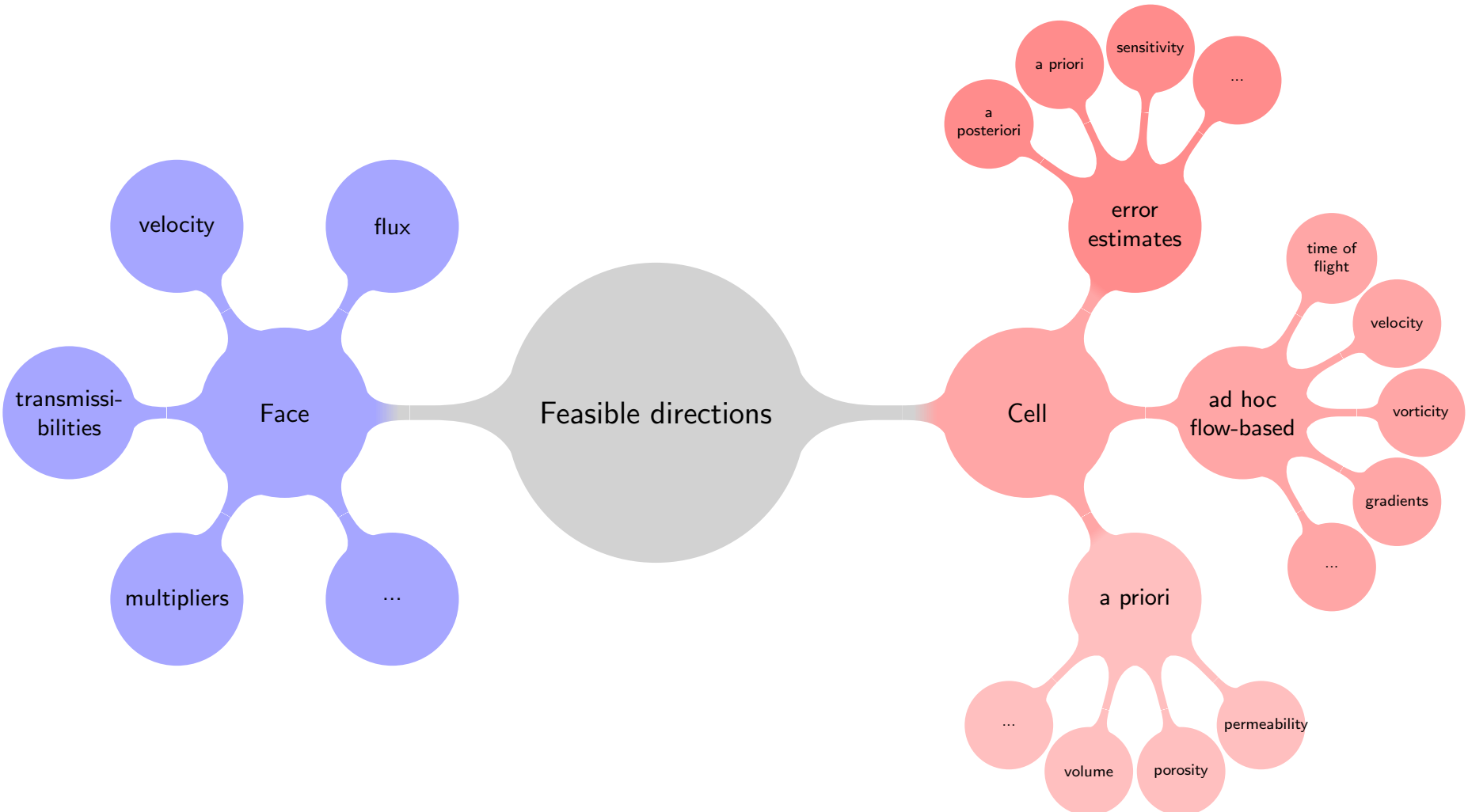


52 blocks

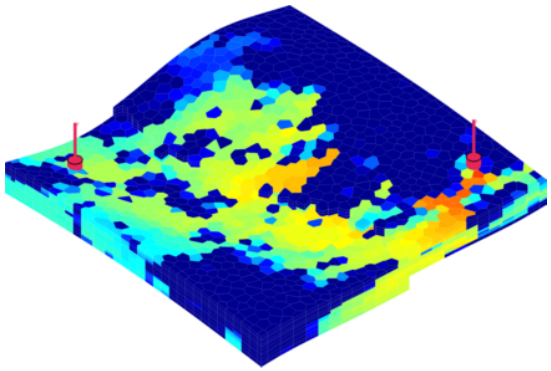
58 blocks



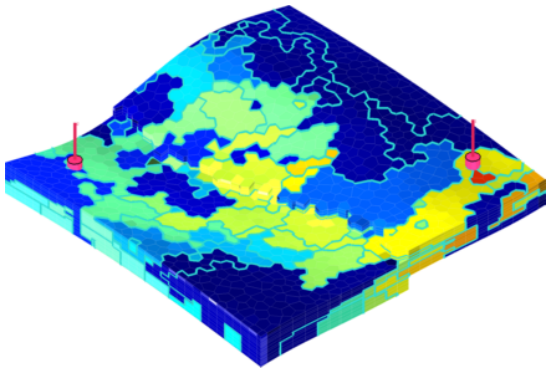
Amalgamation: feasible directions (indicators)



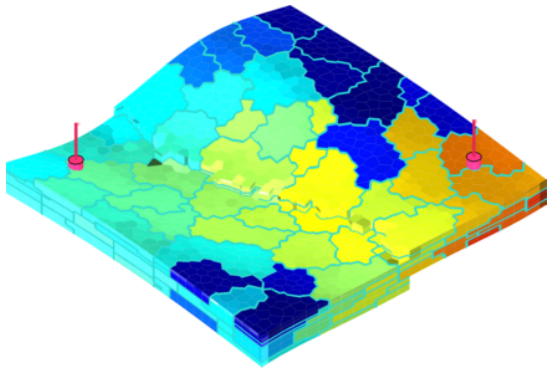
Example: flow-based indicators



reference solution
11 864 cells



time-of-flight grid
127 blocks



METIS grid
175 blocks

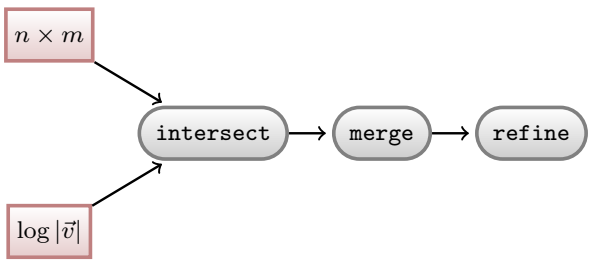
General observations:

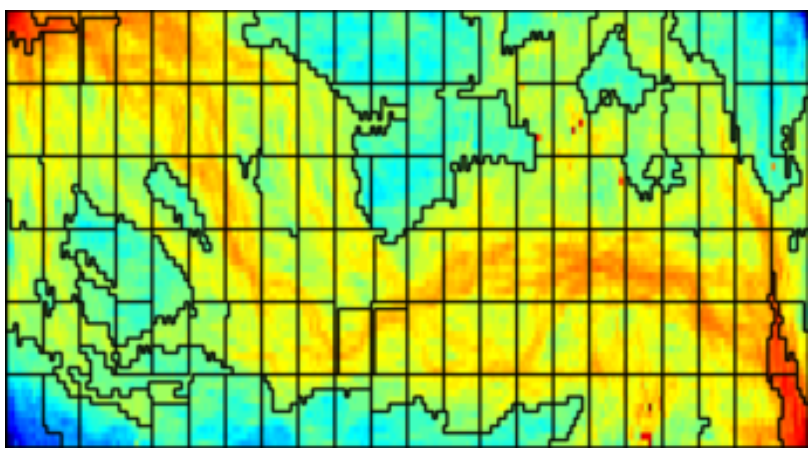
- Time-of-flight is typically a better indicator than velocity
- Velocity is a better indicator than vorticity
- Vorticity is a better indicator than permeability
- ...

However, for smooth heterogeneities, the indicators tend to overestimate the importance of flow.

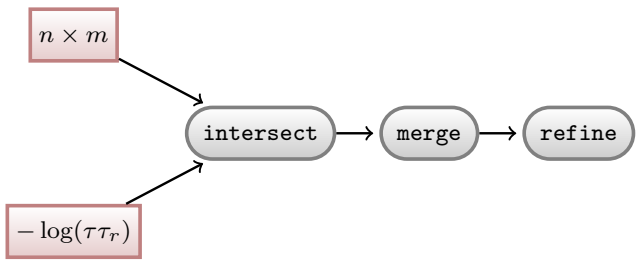
Example: hybrid methods

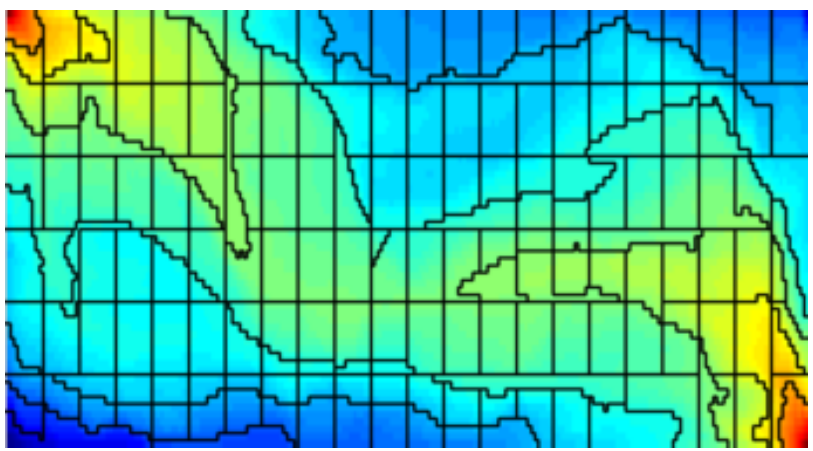
Velocity + Cartesian partition:





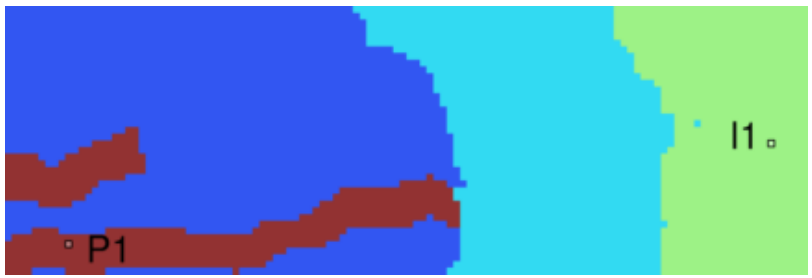
Time-of-flight + Cartesian partition:

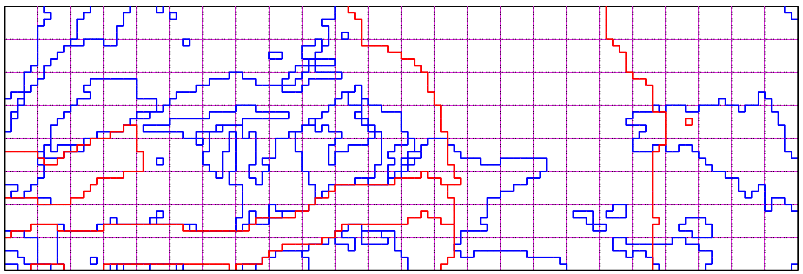


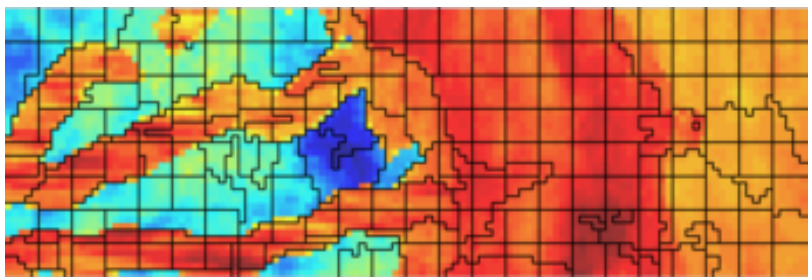


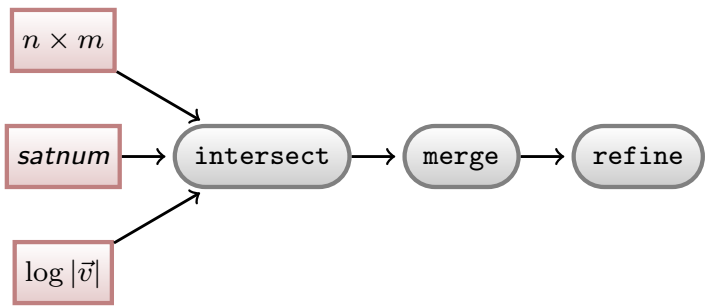
Example: hybrid methods

Satnum + velocity + Cartesian:









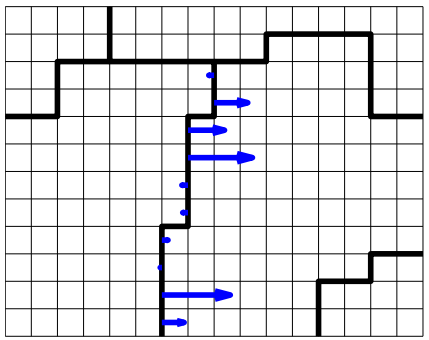
Flow-adapted coarsening: summary

- Developed a general and flexible framework
  - Heuristic algorithms: good rather than optimal grid
  - Algorithmic components: partition, intersection, merging, refinement
  - Key concepts: flow indicator, admissible and feasible directions
- Systematic way of generating fit-for-purpose grids
  - Several existing methods appear as special cases
- Inclusion of geological information and expert knowledge important
  - Facies, saturation regions, surfaces, faults, etc.
  - Predefined shapes and topologies

Coarse-grid discretisation

Bi-directional fluxes (upwind on fine scale):

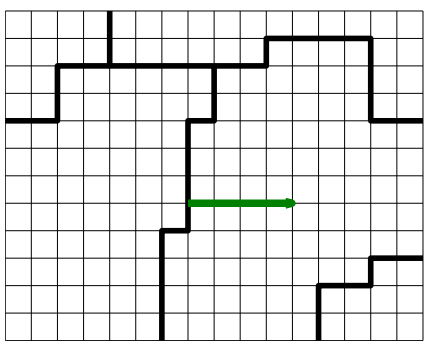
$$S_\ell^{n+1} = S_\ell^n - \frac{\Delta t}{\phi_\ell |B_\ell|}\Big[f(S_\ell^{n+1}) \sum_{\partial B_\ell} \max(v_{ij}, 0) - \sum_{k \neq \ell} \Big(f(S_k^{n+1}) \sum_{\Gamma_{k\ell}} \min(v_{ij}, 0) \Big) \Big].$$



This gives a centred scheme on the coarse scale

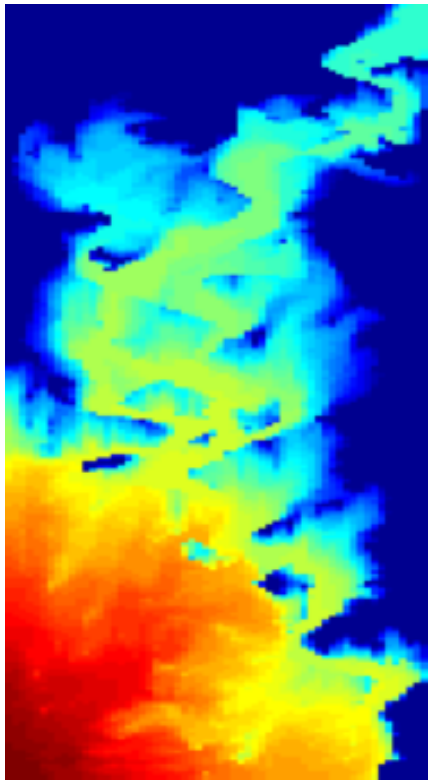
Net fluxes:

$$S_\ell^{n+1} = S_\ell^n - \frac{\Delta t}{\phi_\ell |B_\ell|} \sum_{k \neq \ell} \max\Big(f(S_\ell^{n+1}) \sum_{\Gamma_{k\ell}} v_{ij}, - f(S_k^{n+1}) \sum_{\Gamma_{k\ell}} v_{ij} \Big).$$

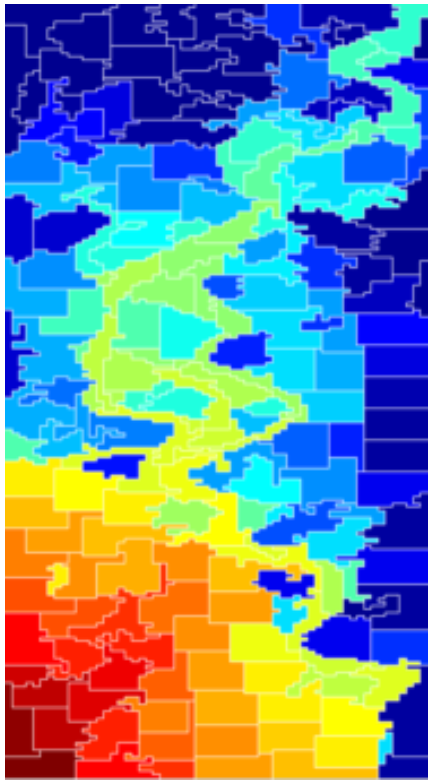


This gives an upwind scheme on the coarse scale

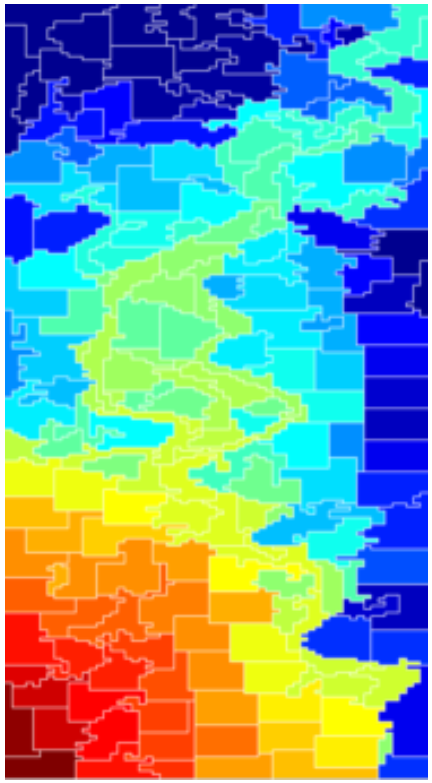
Coarse-grid discretisation: numerical diffusion



reference



net fluxes

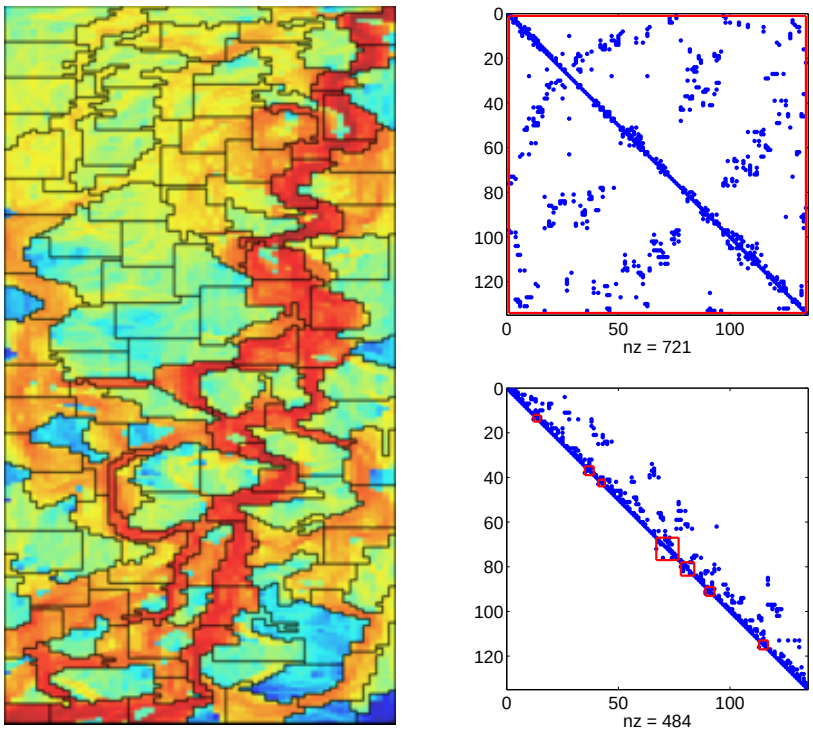


bi-directional fluxes

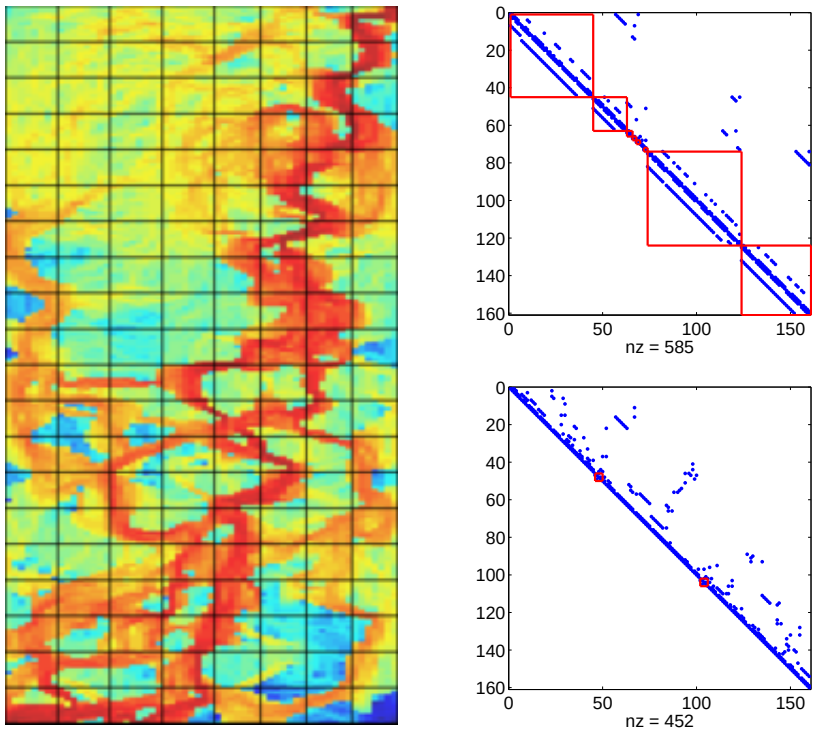
Layer 37 from SPE10

Coarse-grid discretisation: matrix structure

Flow-adapted grid

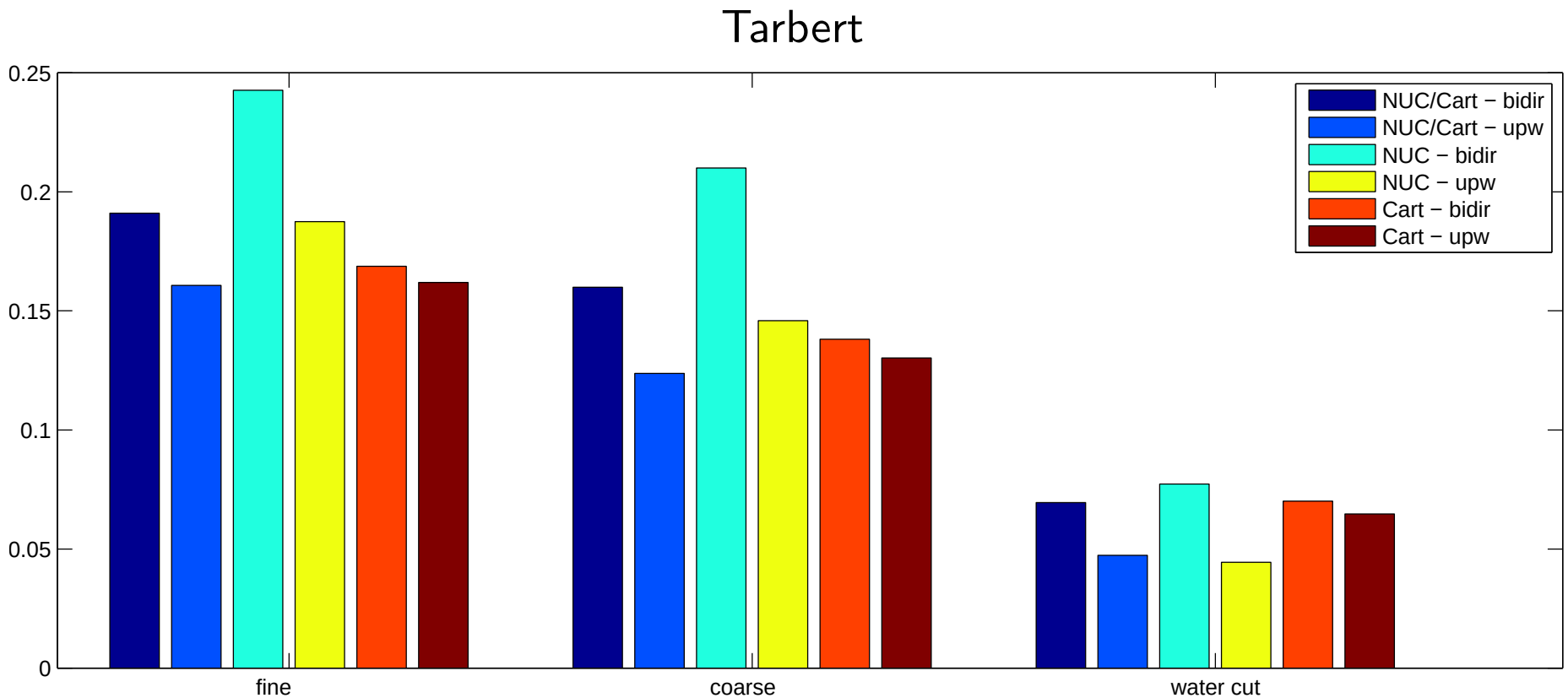


Cartesian grid



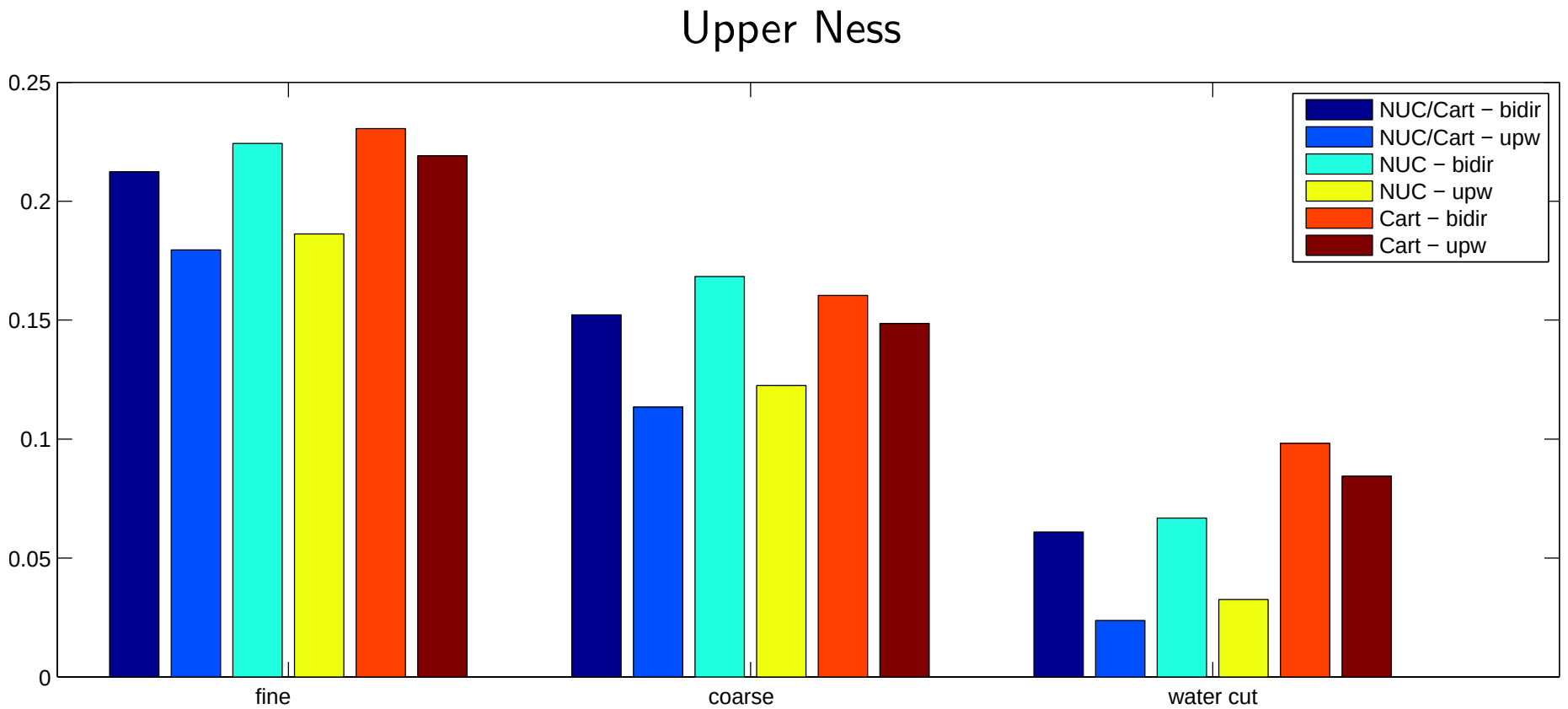
Layer 68 from SPE10. Top: bi-directional fluxes. Bottom: net fluxes

Coarse-grid discretisation: numerical errors



Average errors over all layers of the two formations in SPE10

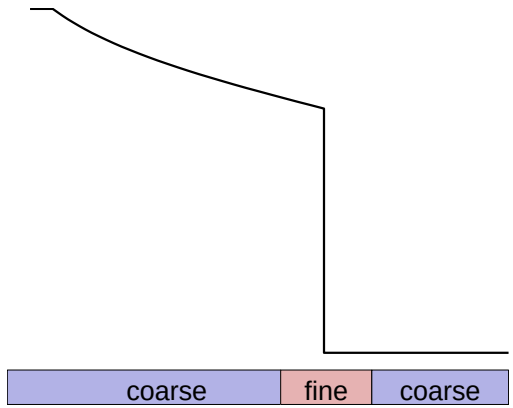
Coarse-grid discretisation: numerical errors



Average errors over all layers of the two formations in SPE10

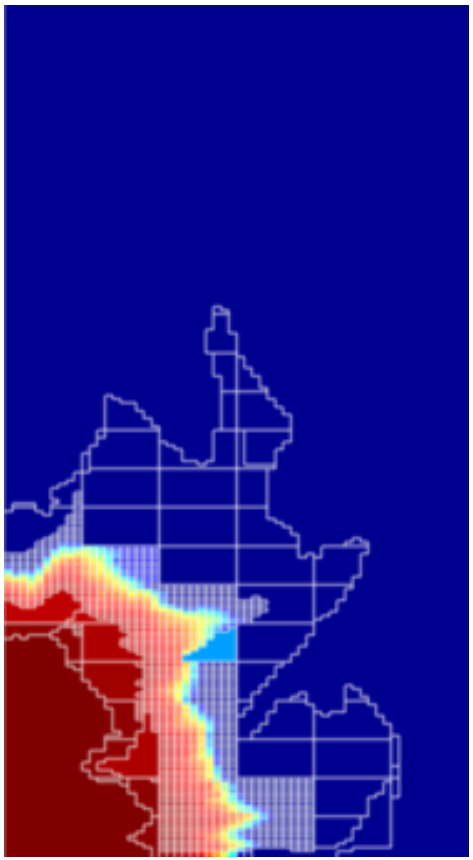
Dynamical adaption

- Inaccurate representation of strong displacement fronts can lead to significant errors.
- **Idea:** Refine dynamically around strong fronts.
- For a Buckley–Leverett displacement:
  - Unswept region ahead of the displacement: coarse grid.
  - Swept region behind the front: coarse grid.
  - At the front: fine or intermediate grid.

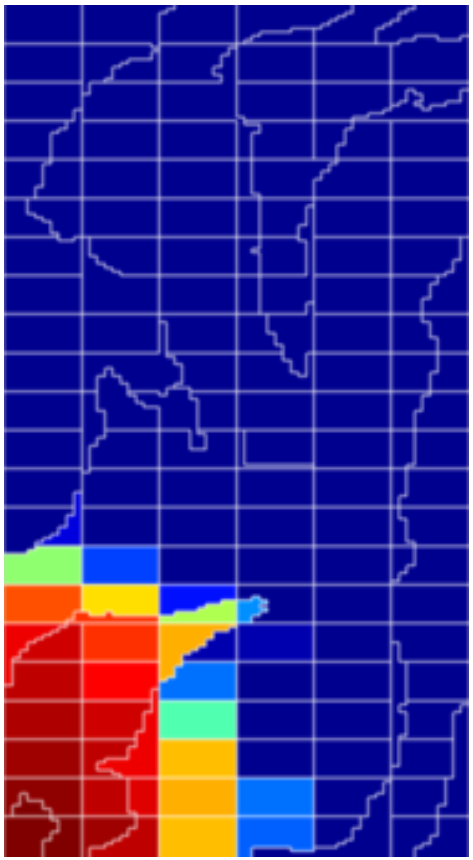


Example: layers from SPE10

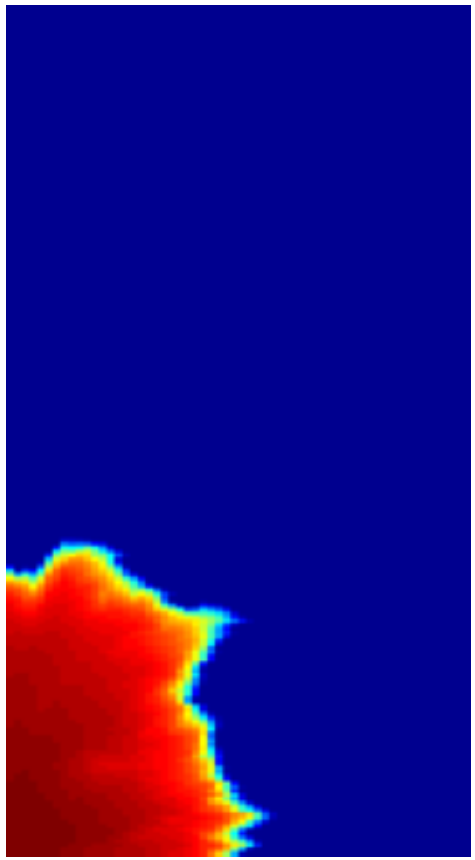
After injection of 0.1 PVI



adaptive



coarse

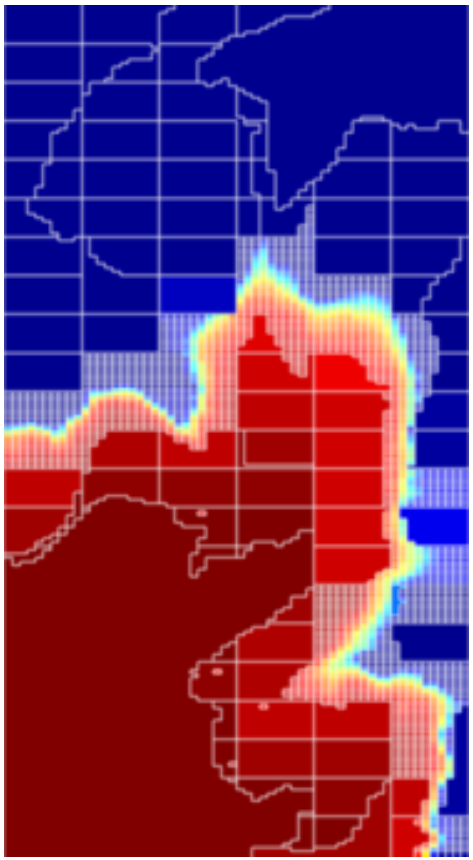


fine grid

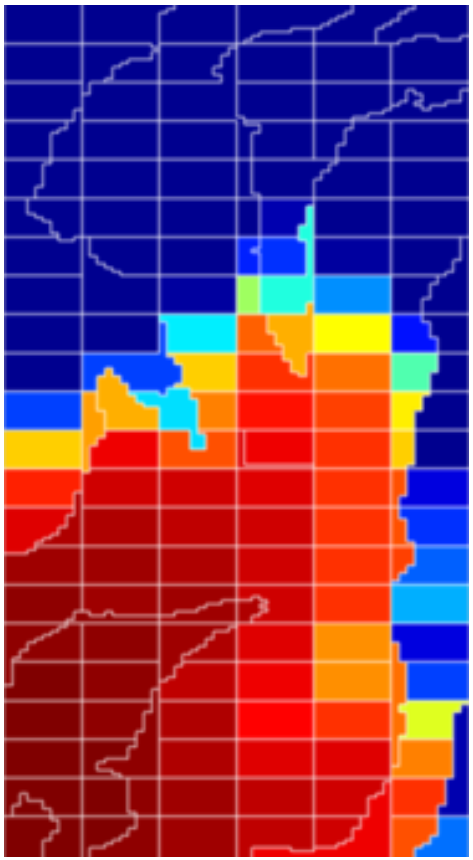
Layer 22 from SPE10

Example: layers from SPE10

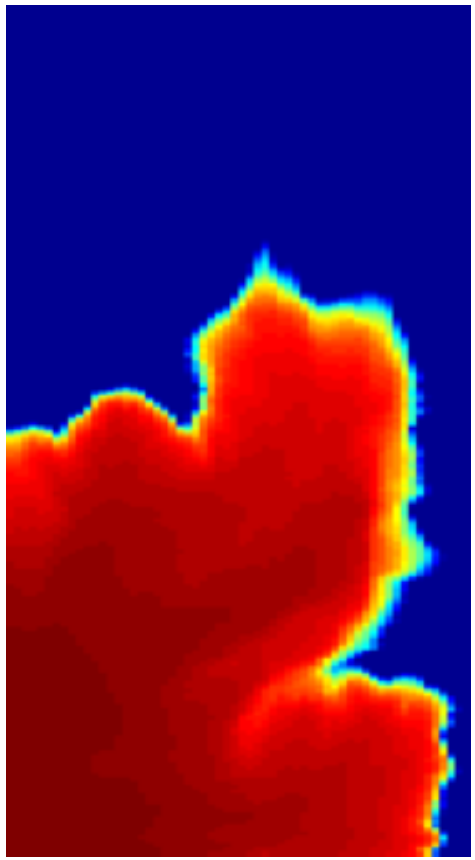
After injection of 0.5 PVI



adaptive



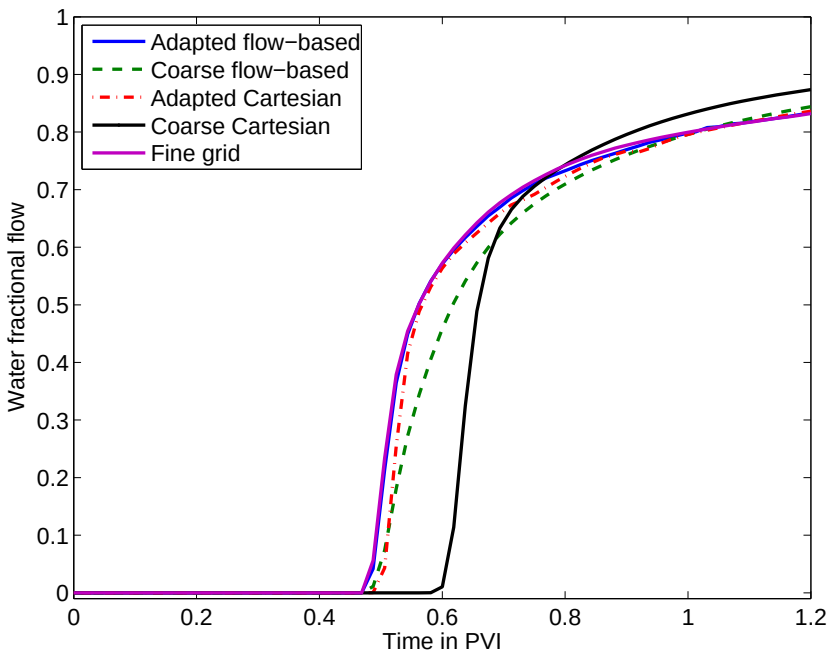
coarse

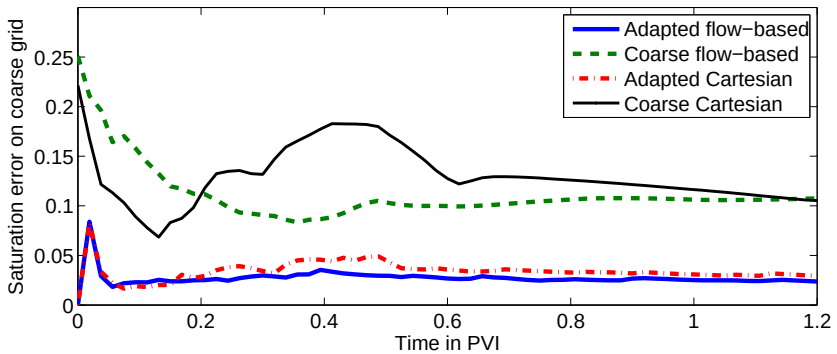


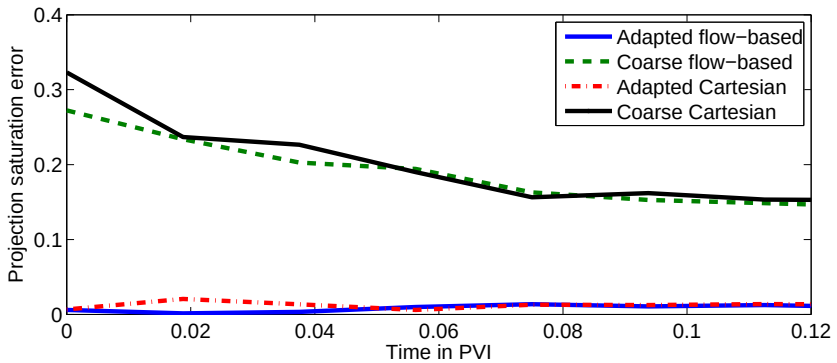
fine grid

Layer 22 from SPE10

Example: layers from SPE10







Layer 37 from SPE10