Discretisation on Complex Grids – Open-Source MATLAB Implementation

K.–A. Lie, S. Krogstad, I. S. Ligaarden, J. R. Natvig, H. M. Nilsen, and B. Skaflestad
Department of Applied Mathematics, SINTEF ICT, Oslo, Norway

Summary

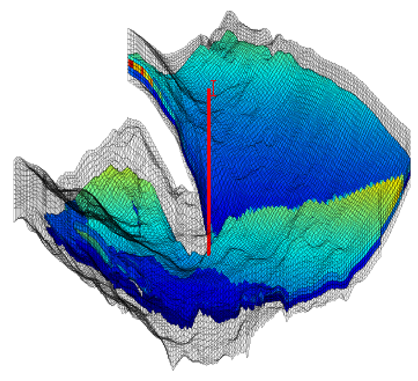
Accurate geological modelling of features such as faults, fractures or erosion requires grids that are flexible with respect to geometry. Such grids generally contain polyhedral cells and complex grid cell connectivities. The grid representation for polyhedral grids in turn affects the efficient implementation of numerical methods for subsurface flow simulations.

We give an overview of an open-source MATLAB toolbox that has been developed to support our research on new, consistent and convergent, computational methodologies. The toolbox offers flexibility and efficiency with respect to different grid formats, and in particular hierarchical grids used in multiscale methods.

The MATLAB Reservoir Simulation Toolbox (MRST)

The toolbox has the following functionality for rapid prototyping of solvers for flow and transport:

- grid structure, grid factory routines, input/processing of industry-standard formats, real-life and synthetic example grids
- petrophysical parameters and incompressible fluid models, conversion routines to/from SI and common field units, very simplified geostatistical routines
- routines for setting and manipulating boundary conditions, sources/sinks, and well models
- reservoir state (pressure, fluxes, saturations, compositions, . . .)
- visualisation routines for cell and face data (scalars)



Solvers: mimetic and multiscale pressure solvers, TPFA, MPFA-O, explicit and implicit upwind for transport

Inhouse prototypes: compressible black-oil, (multiscale) adjoint methods, fault multipliers, flow-based coarsening, streamlines, vertically averaged models, . . .

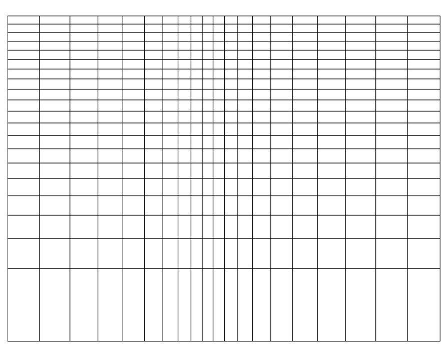
Version 2010a was released in March 2010 and can be downloaded under the terms of the GNU General Public License (GPL).

Flexible Gridding

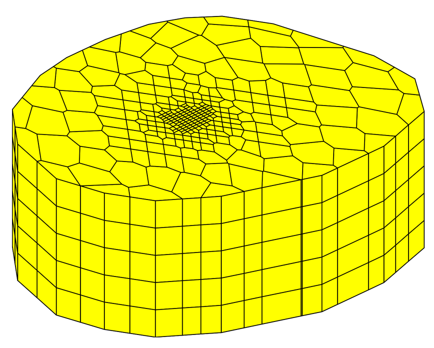
All grids in MRST are considered as fully unstructured:

- non-overlapping polyhedral cells with matching planar faces
- non-matching grids are converted to matching by subdividing faces
- stored in a general unstructured format describing cells, faces, and nodes and how they are connected

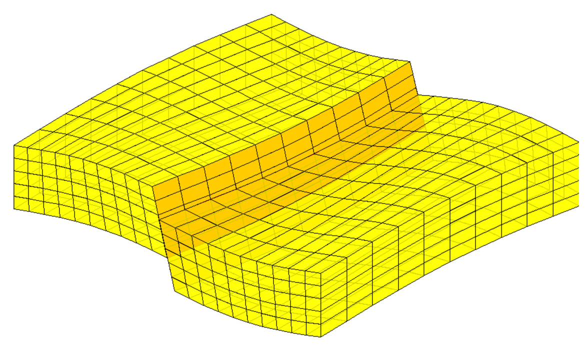
Examples: basic grid types and how to create them



```
% Rectillinear grid
dx = 1−0.5*cos((−1:0.1:1)*pi);
x = −1.15+0.1*cumsum(dx);
G = tensorGrid(x, sqrt(0:0.05:1));
plotGrid(G);
```

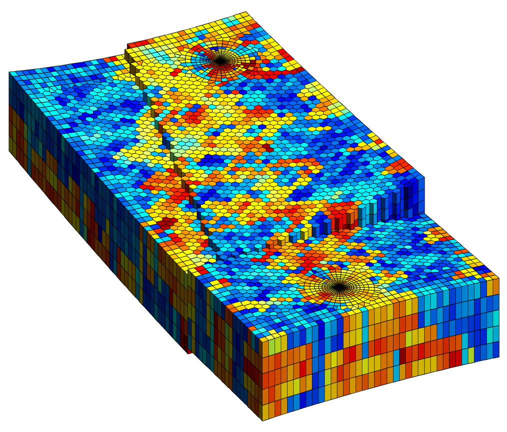


```
% Extrude a standard MATLAB dataset
load seamount
V = makeLayeredGrid(pebi(tri2grid(...
    delaunay(x,y),[x(:) y (:)])), 5);
plotGrid(V), view(−40, 60), axis off
```

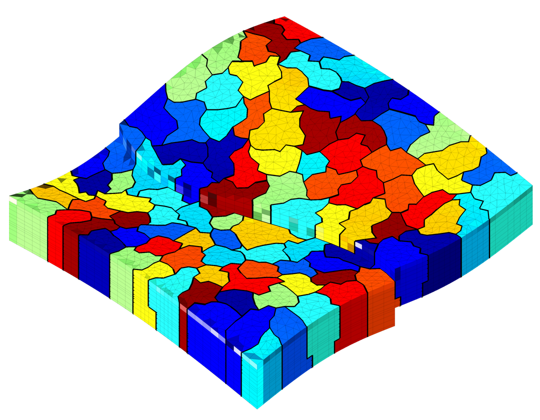


```
% Make and read a simple Eclipse input file
G = processGRDECL(simpleGrdecl([20, 10, 5], 0.12));
plotGrid(G,'FaceAlpha',0.8);
plotFaces(G,find(G.faces.tag>0), 'FaceColor','red');
view(40,40), axis off
```

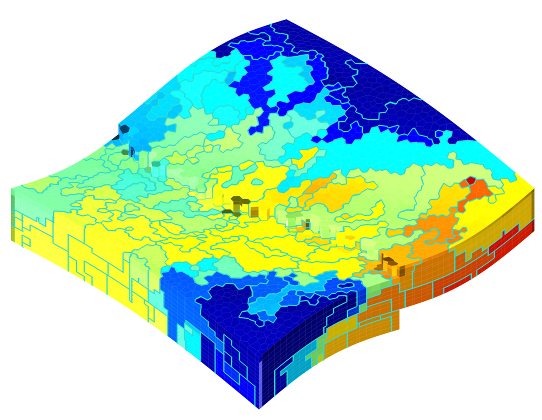
Examples: hybrid and hierarchical grids



Hybrid grid: extrusion of an areal grid consisting of structured and unstructured parts: (i) radially refined grid at wells, (ii) Cartesian grid along boundary, (iii) hexahedral in the interior, and (iv) polyhedral transition cells



Hierarchical grid: prismatic fine grid constructed by extrusion of a triangular grid along vertical pillars, and a coarse grid constructed by collecting cells from the fine grid into blocks [1]



Flow-based grid: the fine grid is a PEBI grid constructed as the dual of the triangular grid in the middle plot, the coarse grid is constructed by grouping PEBI cells into blocks according to flow magnitude [2]

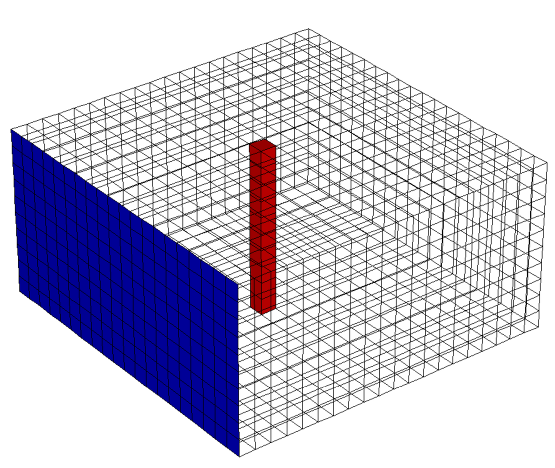
Example of an MRST Solver

Solve the single-phase flow equation

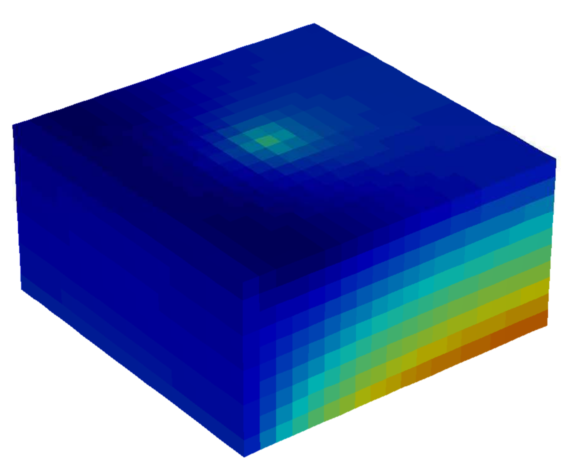
$$\nabla \cdot \vec{v} = q, \qquad \vec{v} = -\frac{\mathbf{K}}{\mu}[\nabla p + \rho g \nabla z]$$

inside a $20 \times 20 \times 10$ m domain, with a vertical well in the centre and a prescribed boundary pressure of 10 bar at the left-hand side.

```
% Grid and rock parameters
nx = 20; ny = 20; nz = 10;
G = computeGeometry(cartGrid([nx, ny, nz]));
rock.perm = repmat(100 * milli*darcy, [G.cells.num, 1]);
fluid = initSingleFluid('mu', 1*centi*poise, ...
                        'rho', 1014*kilogram/meter^3);
gravity reset on

% Fluid sources and boundary conditions
c   = (nx/2*ny+nx/2 : nx*ny : nx*ny*nz) .';
src = addSource([], c, ones(size(c)) ./ day());
bc = pside([], G, 'LEFT', 10*barsa());

% Construct components for mimetic system
S = computeMimeticIP(G, rock, 'Verbose', true);

% Solve the system and convert to bars
rSol = initResSol(G, 0);
rSol = solveIncompFlow(rSol, G, S, fluid,'src', src, 'bc', bc);
p = convertTo(rSol.pressure(1:G.cells.num), barsa() );
```



Source term and boundary condition

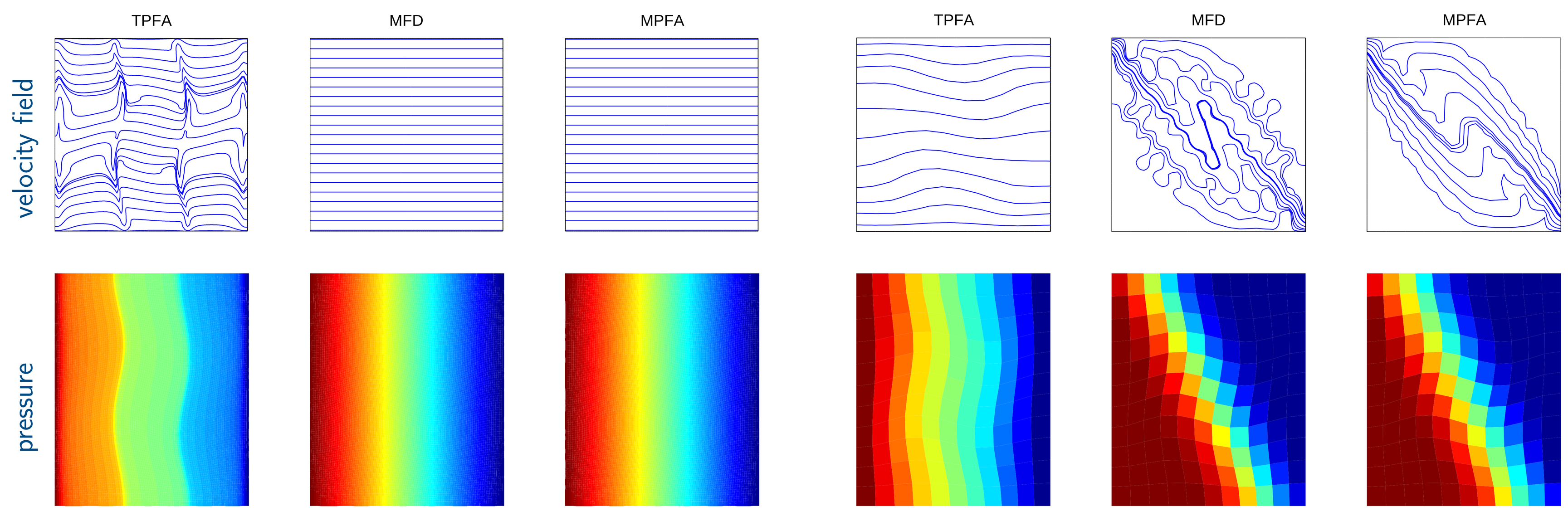


Pressure distribution

Grid-Orientation Effects and Monotonicity

We consider three different flow solvers: TPFA, MPFA-O, and a mimetic finite-difference (MFD) scheme. Whereas TPFA is convergent only on K-orthogonal grids, the two multipoint schemes are constructed to be convergent for rough grids and full-tensor permeabilities.

Example:



Grid-orientation: 100×100 grid with anisotropic permeability tensor with anisotropy (ratio 1 : 1000 aligned with the x-axis), horizontal unit pressure drop, and no-flow on top and bottom

Monotonicity: 11×11 grid with anisotropic permeability tensor (ratio 1 : 1000) rotated by $\pi/6$, horizontal pressure drop, no-flow on top and bottom

Rough grid: normalise a uniform Cartesian grid to the unit square, add/subtract $0.03 \sin(\pi x) \sin(3\pi(y - 1/2))$ to the x- and y-coordinates, and transform back

Modelling Faults in Multipoint, Mimetic, and Mixed Methods

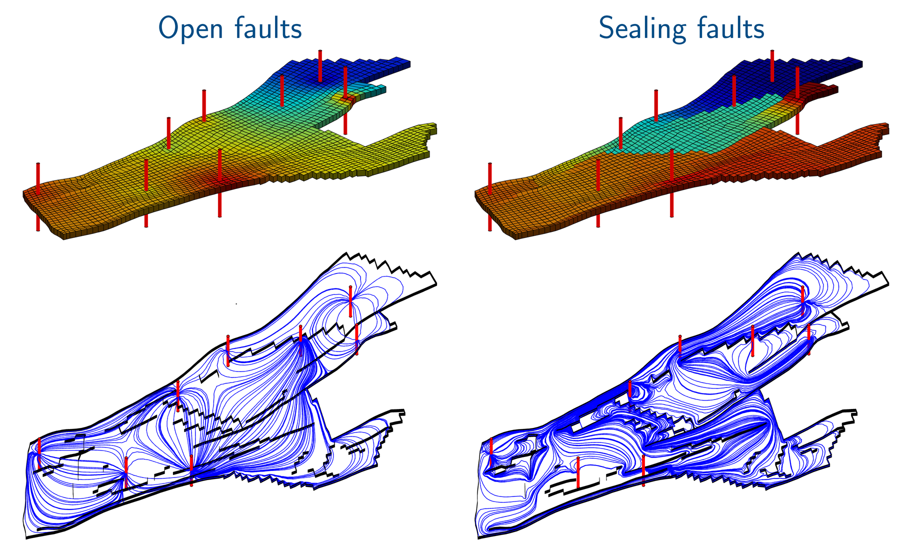
The industry standard to represent flow barriers is to use non-negative transmissibility multipliers less than unity. Such multipliers are highly grid dependent and strictly associated with a connection between two grid cells rather than with the fault itself (and hence tied to a particular discretisation).

In MRST, flow barriers are represented as internal boundaries [3]:

- unique pressure assigned to each fault face
- continuity explicitly imposed across the fault

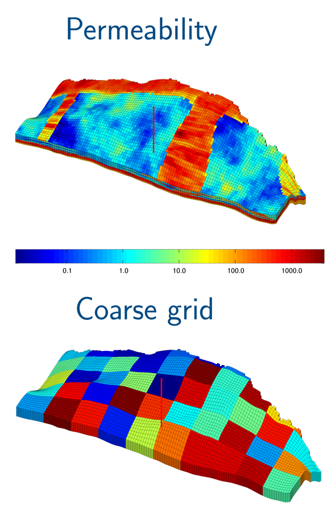
Example: we demonstrate the effects of fault and shale-layer barriers on a 2D cartoon model derived from a real-field model from the Norwegian Sea. The faults, shown in black, are modelled as internal boundaries and are set to be either open or fully sealing resulting in very different flow patterns, as shown by the streamlines.

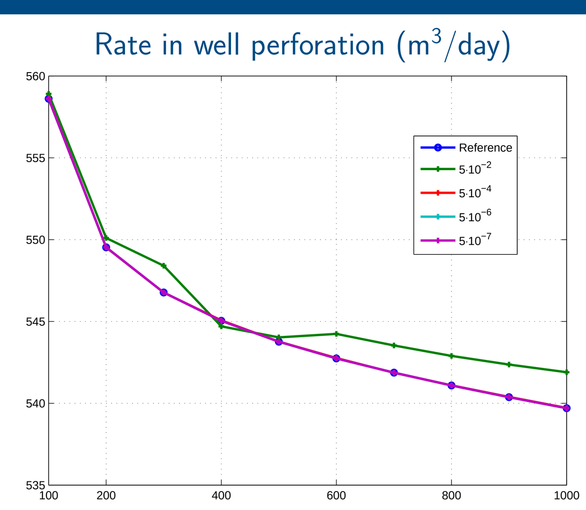
(Simulations for the full model are reported in the proceedings paper).



Primary Production from a Gas Reservoir

- Shallow-marine reservoir (realisation from SAIGUP)
- Model size: $40 \times 120 \times 20$
- Initially filled with gas, 200 bar
- Single producer, bhp=150 bar
- Multiscale solution for different tolerances compared with fine-scale reference solution.





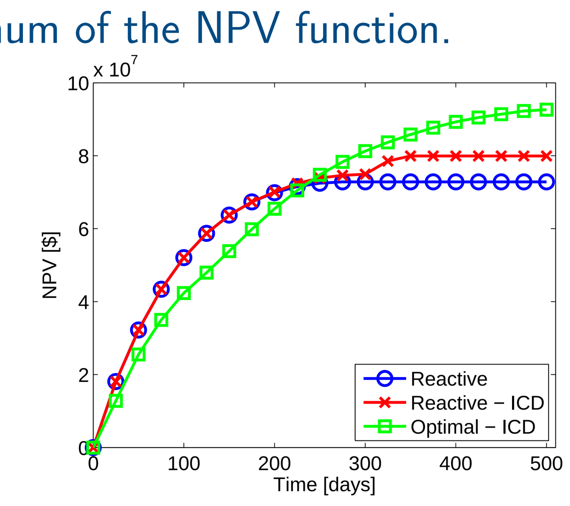
Production Optimisation

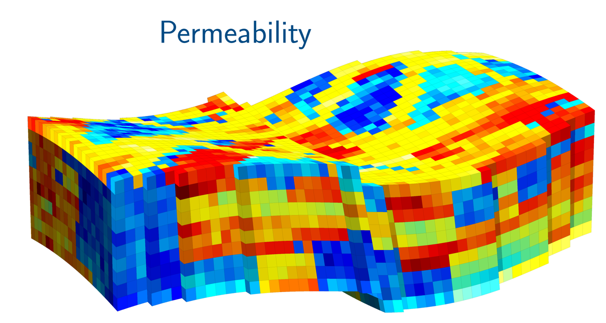
Many reservoir management challenges can be cast as mathematical optimisation problems. To accelerate gradient-based methods, we have developed an efficient multiscale simulator that combines an adjoint multiscale pressure solver with a transport solver formulated on flow-adapted grids [3].

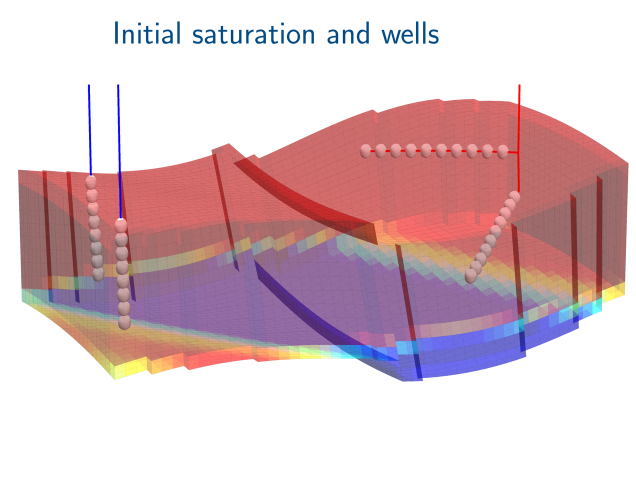
Example: Optimising net present value (NPV)

Three production strategies:

1. Constant BHP until water cut of 0.82, when the well shuts in.
2. Constant BHP with ICD shutting down well segments in which the water cut has reached 0.82.
3. Optimal segment rates corresponding to local maximum of the NPV function.







References

[1] J. E. Aarnes, S. Krogstad, and K.-A. Lie. Multiscale mixed/mimetic methods on corner-point grids. *Comput. Geosci.*, 12(3):297–315, 2008. DOI: 10.1007/s10596-007-9072-8

[2] V. L. Hauge, K.-A. Lie, and J. R. Natvig. Flow-based grid coarsening for transport simulations. Proceedings of ECMOR XII.

[3] S. Krogstad, V. L. Hauge, and A. F. Gulbransen. Adjoint multiscale mixed finite elements. *SPE J.*, to appear. SPE119112.

[4] H. M. Nilsen, K.-A. Lie, J. R. Natvig, and S. Krogstad. Accurate modelling of faults by multipoint, mimetic, and mixed methods. Preprint 2010. http://www.sintef.no/Geoscale/

Bard.Skaflestad@sintef.no