The MATLAB Reservoir Simulation Toolbox (MRST)

Knut-Andreas Lie

SINTEF Digital, Oslo, Norway

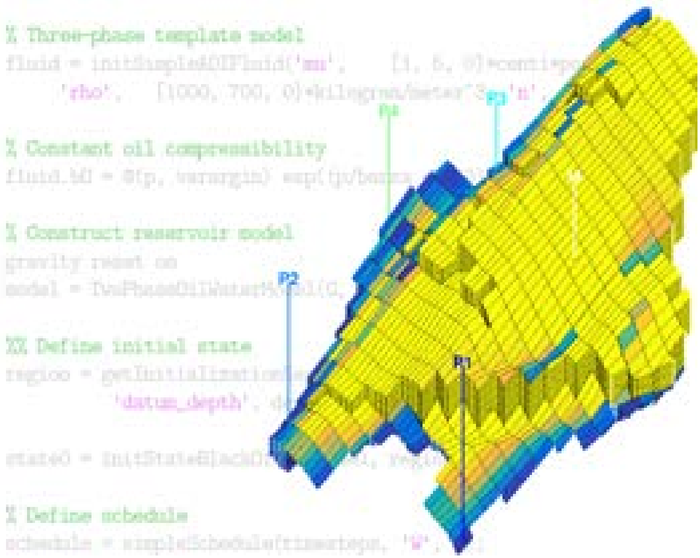
DARTS workshop, TU Delft, March 7–8 2023

MATLAB Reservoir Simulation Toolbox (MRST)

Transforming research on reservoir modelling

Unique prototyping platform:

- Standard data formats
- Data structures/library routines
- Fully unstructured grids
- Rapid prototyping:
  - differentiation operators
  - automatic differentiation
  - object-oriented framework
  - state functions
- Industry-standard simulation



```
% Three-phase template model
fluid = initSimpleADIFluid('mu',    [1, 5, 0]*centipo
    'rho',   [1000, 700, 0]*kilogram/meter^3, 'n',

% Constant oil compressibility
fluid.bO = @(p, varargin) exp((p/bar

% Construct reservoir model
gravity reset on
model = TwoPhaseOilWaterModel(G,

%% Define initial state
region = getInitializationRe
           'datum_depth', d

state0 = initStateBlackOi

% Define schedule
schedule = simpleSchedule(timesteps, 'W',
```

http://www.mrst.no

MATLAB Reservoir Simulation Toolbox (MRST)

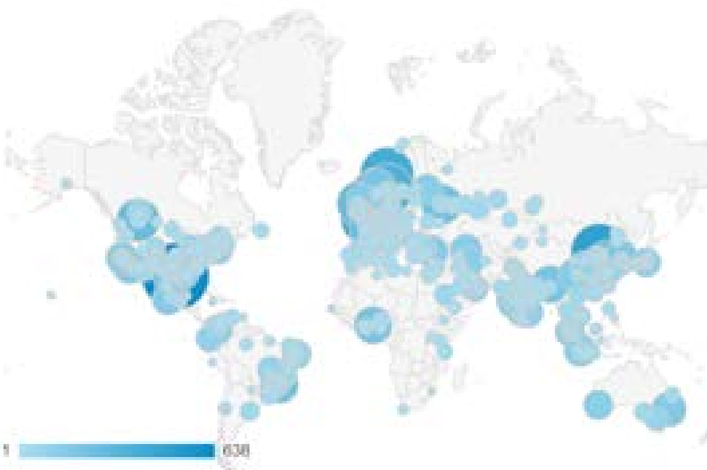
Transforming research on
reservoir modelling

Large international user base:

- downloads from the whole world
- 136 master theses
- 78 PhD theses
- 367 journal papers (not by SINTEF)
- 205 proceedings papers

Numbers are from Google Scholar notifications

Used both by academia and industry



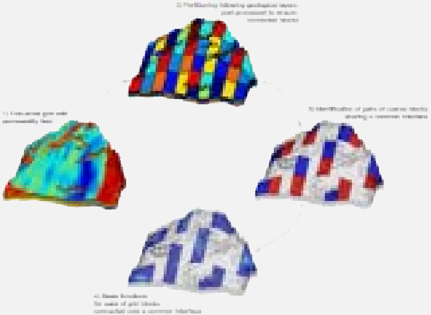
Google Analytics: access pattern for `www.mrst.no`
Period: 1 July 2018 to 31 December 2019

Why did we start developing MRST?

Like many other things, MRST grew out of frustrations:

- 2004: started building a group working on multiscale methods and consistent discretizations
- Experimental numerics: new codes for each new paper
- Time-consuming to code other methods for comparisons

⟶ systematic development of internal research toolbox (first commit to svn in October 2007)


2) Partitioning following geological layers,
post-processed to ensure
connected blocks
1) Fine-scale grid with
permeability field
3) Identification of pairs of coarse blocks
sharing a common interface
4) Basis functions
for pairs of grid blocks
connected over a common interface

Multiscale mixed finite-element methods: focus on handling realistic grids, and corner-point and PEBI grids in particular

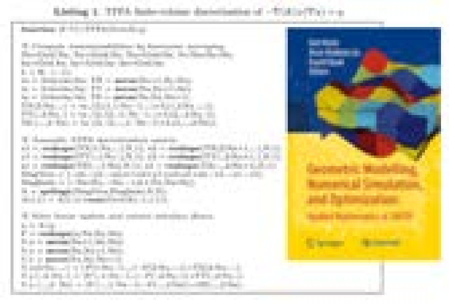
Why did we start developing MRST?

Like many other things, MRST grew out of frustrations:

- 2004: started building a group working on multiscale methods and consistent discretizations
- Experimental numerics: new codes for each new paper
- Time-consuming to code other methods for comparisons

⟶ systematic development of internal research toolbox (first commit to svn in October 2007)

Meanwhile: educational paper on simulating porous media flow in 50 lines of MATLAB became surprisingly popular



Aarnes, Gimse and Lie. An introduction to the numerics of flow in porous media using Matlab. In "Geometrical Modeling, Numerical Simulation, and Optimization: Industrial Mathematics at SINTEF", Springer Verlag, pp. 265-306, 2007.

Why did we start developing MRST?

Like many other things, MRST grew out of frustrations:

- 2004: started building a group working on multiscale methods and consistent discretizations
- Experimental numerics: new codes for each new paper
- Time-consuming to code other methods for comparisons

⟶ systematic development of internal research toolbox (first commit to svn in October 2007)

Meanwhile: educational paper on simulating porous media flow in 50 lines of MATLAB became surprisingly popular

Why not simply release our toolbox for free?

Partial motivation:

- SINTEF pioneered C++ for numerics. Our experience: prototyping in MATLAB much more efficient
- Frustration as PhD student: need software pieces from outside domain of expertise
- Hard to get industry interest if results are not part of commercial simulator
- Sharing code forces you to polish/generalize ⟶ easier to maintain know-how

Reservoir simulation in MATLAB...?

MATLAB/GNU Octave was/is widely used in academic institutions

Different development process:

- Use abstractions to express your ideas in a form close to the underlying mathematics
- Build your program using an interactive environment:
  - try out each operation and build program as you go
- Dynamic type checking lets you prototype while you test an existing program:
  - run code line by line, inspect and change variables at any point
  - step back and rerun parts of code with changed parameters
  - add new behavior and data members while executing program

- MATLAB is fairly efficient using vectorization, logical indexing, external iterative solvers, etc.
- Avoids build process, linking libraries, cross-platform problems
- Builtin mathematical abstractions, numerics, data analysis, visualization, debugging/profiling,
- Use scripting language as a wrapper when you develop solvers in compiled languages

Community code: software organization

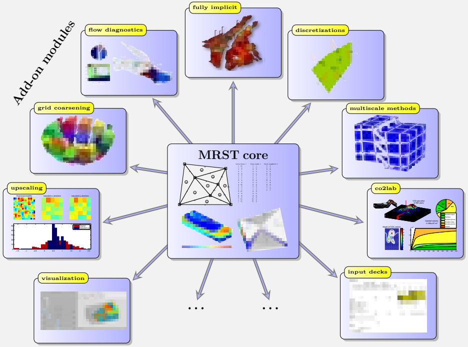
Modular design:

- **small core** with mature and well-tested functionality used in *many* programs or modules
- **semi-independent modules** extend core functionality
- in-source documentation like in MATLAB
- all modules must have code examples and/or tutorials

This simplifies how we distinguish public and in-house or client-specific functionality

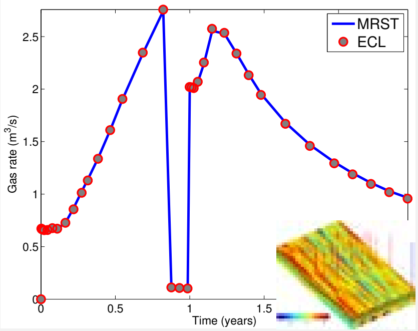
Latest release: MRST 2022b has 65 modules

3 967 files, 275 000 code lines, 131 000 comment lines
Also: 23 000 lines of C/C++



Key to large user-base: high code integrity/quality

- Strive to maintain professional code quality
- Use standard MATLAB as much as possible
- Try to be compatible with GNU Octave
- Document all functionality a user might utilize
- Follow consistent name conventions (camelCase)
- Use distributed version control (git)
- Utilize code review and automated testing
- Perform extensive prerelease testing
- Verify and validate new functionality
- . . .



Example: validation of MRST against ECLIPSE100 for the SPE9 benchmark case.

Downloading the software

MRST - Transforming research on reservoir simulation

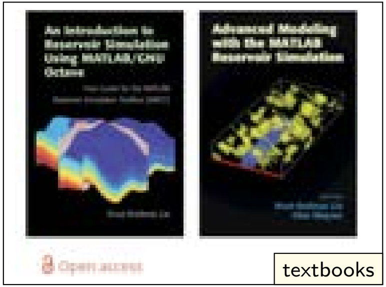
[illegible]

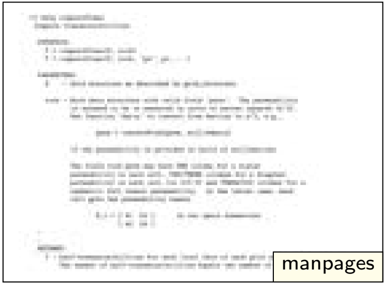
- URL: `www.mrst.no`
- Free software with GNU GPL license
- Released twice per year
- Latest release: MRST 2022b, from 30 September 2022
- Provided as a self-contained archive file (e.g., `mrst-2022b.zip`)
- We recommend that you join the `MRST-announce` Google group

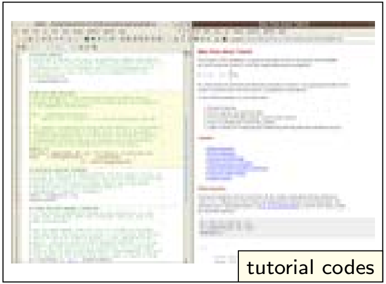
Learning more and getting help

website

user forum


An Introduction to Reservoir Simulation Using MATLAB/GNU Octave
Advanced Modeling with the MATLAB Reservoir Simulation
Open access
textbooks

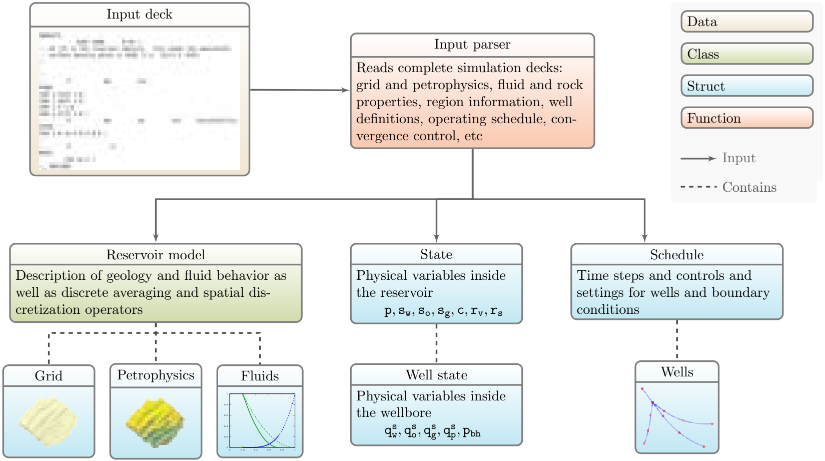

manpages


tutorial codes

online tutorials

Feature highlights

Capabilities as in commercial simulators



Input deck

Input parser

Reads complete simulation decks: grid and petrophysics, fluid and rock properties, region information, well definitions, operating schedule, convergence control, etc

Reservoir model

Description of geology and fluid behavior as well as discrete averaging and spatial discretization operators

State

Physical variables inside the reservoir

$p, s_w, s_o, s_g, c, r_v, r_s$

Schedule

Time steps and controls and settings for wells and boundary conditions

Grid

Petrophysics

Fluids

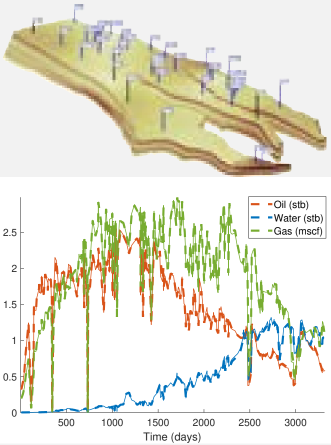
Well state

Physical variables inside the wellbore

$q^s_w, q^s_o, q^s_g, q^s_p, p_{bh}$

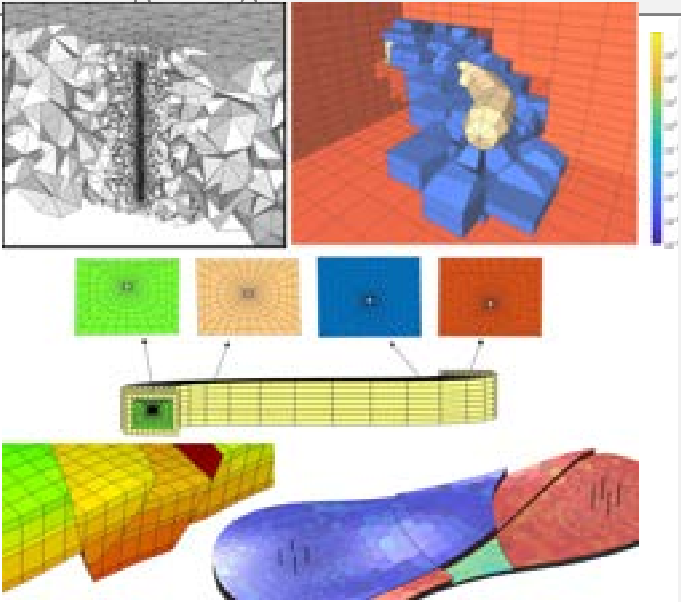
Wells

Example:



Field production compared with OPM Flow for the Norne field

Advanced gridding schemes



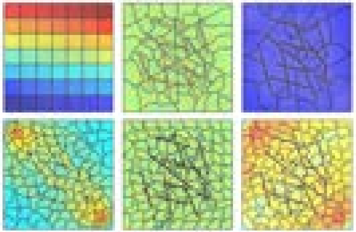
MRST grids are chosen to always be **fully unstructured** ⟶ can implement algorithms without knowing the specifics of the grid

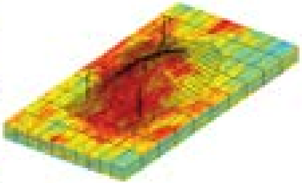
- Cartesian and rectilinear
- Corner-point
- Tetrahedral, prismatic, PEBI
- General polyhedral/polytopal
- Hybrid, cut-cell, or depogrids
- Local refinements . . .

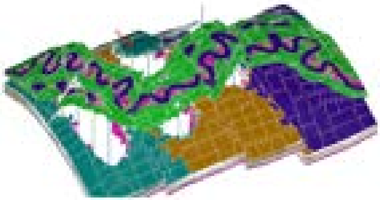
Examples:

- `deckformat` – corner-point grids
- `upr` module – constrained Voronoi
- `nwm` module – near-well gridding
- `gmsh` module – converts gmsh grids to MRST

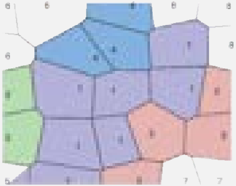
Flexible grid partitioning schemes







All grids in MRST can be partitioned into a coarse grid using integer partition vectors



Enables wide variety of partition schemes:

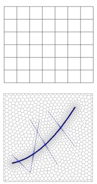
- structured/logical partitions
- aggregated partitions
- feature-based partitions
- flow-based partitions
- hybrid partitions
- dynamic partitions

The resulting coarse grids can be used seamlessly with most solvers

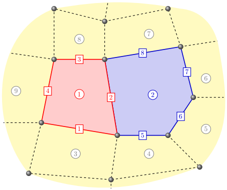
Rapid prototyping (of fully differentiable simulators)

Abstract discretization operators

Idealized models



Grid structure in MRST



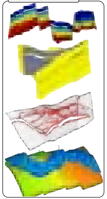
c	F(c)
1	1
1	2
1	3
1	4
2	5
2	6
2	7
2	8
2	2
3	1
⋮	⋮
⋮	⋮

Map: cell → faces

f	C_1	C_2
1	3	1
2	1	2
3	1	8
4	9	1
5	4	2
6	2	5
7	2	6
8	2	7
⋮	⋮	⋮
⋮	⋮	⋮

Map: face → cells

Industry models



Continuous

Incompressible flow:

$$\nabla \cdot (\mathbf{K}\nabla p) + q = 0$$

Compressible flow:

$$\frac{\partial(\phi\rho)}{\partial t} + \nabla \cdot (\rho\mathbf{K}\nabla p) + q = 0$$

Discrete in MATLAB

Incompressible flow:

```
eq = div(T .* grad(p)) + q;
```

Compressible flow:

```
eq = (pv(p).*rho(p)-pv(p0).*rho(p0))/dt ...
   + div(avg(rho(p)).*T.*grad(p))+q;
```

Automatic differentiation:

- you implement residual equations
- AD library automatically linearizes and assembles Jacobians
- simplifies coupling of submodels
- memory-optimized backends

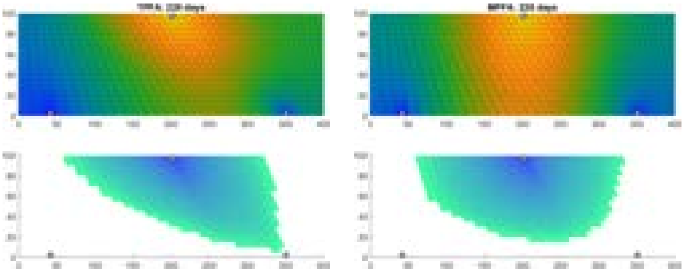
State functions:

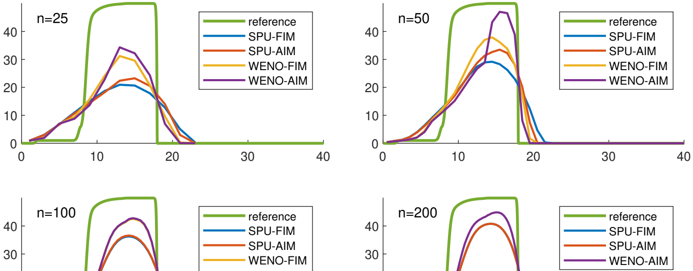
- dependency management
- generic interfaces: plug and play
- compute cache and lazy evaluation

Adjoints:

- gradients and Jacobians with respect to input parameters
- simplifies gradient-based calibration/optimization

A variety of discretization schemes





Industry-standard

- two-point + upstream mobility weighting

Consistent discretizations:

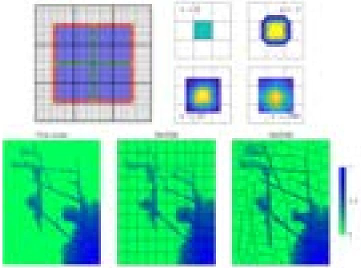
- mimetic methods (MFD)
- multipoint flux approximation (MPFA)
- nonlinear two-point and AvgMPFA
- vertex approximated gradient (VAG)
- virtual element methods (VEM)

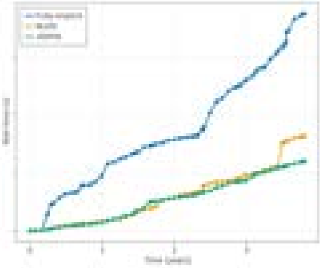
High-resolution methods:

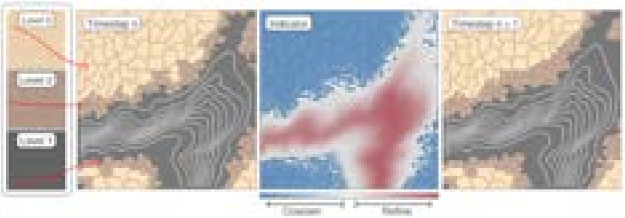
- WENO methods on polytopal grids
- Discontinuous Galerkin methods

All cell-centered schemes can be used in the AD-OO simulator framework

Advanced solvers and solution methods







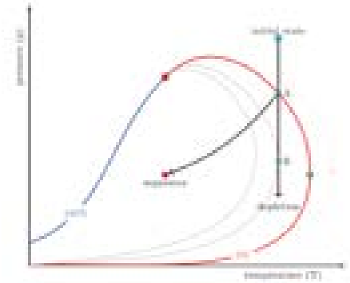
Linear solvers:

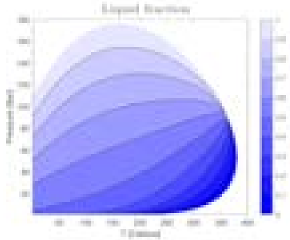
- MATLAB's standard direct solvers
- Preconditioned Krylov and multilevel solvers from AMGCL (GMRES, ILU, smoothed aggregation, etc)
- Multiscale methods (as in SLB's next-generation IX MS SFI)

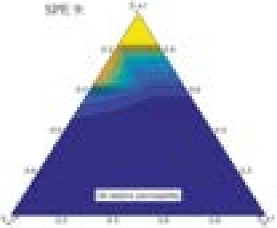
Nonlinear solvers/formulations:

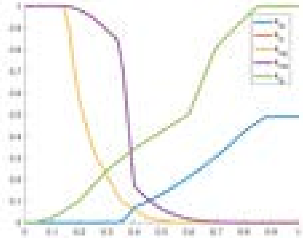
- Fully implicit and adaptive implicit
- Sequential (fully) implicit
- Nonlinear domain-decomposition preconditioning
- Trust-region methods

Different types of flow physics









Incompressible flow:

- Corey-type relative permeability
- linear or Leverett–J capillary pressure

Black-oil models:

- one to three fluid phases
- dissolved gas and vaporized oil
- input primarily from ECLIPSE decks

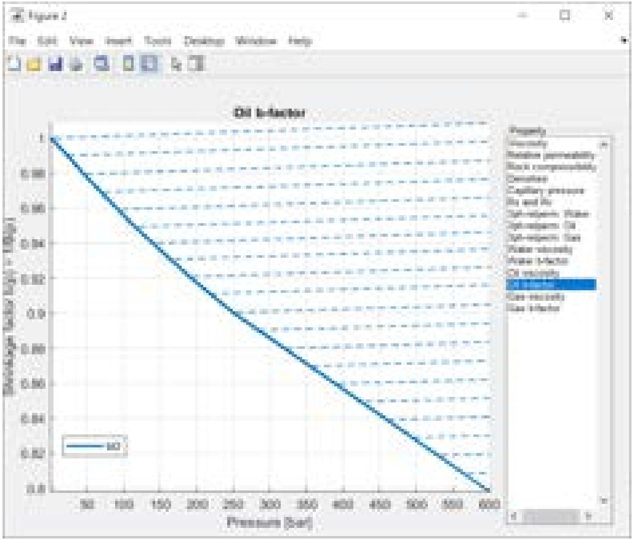
Enhanced oil recovery:

- surfactant, polymer, surfactant–polymer
- solvent

Compositional flow:

- general equation-of-state models
- overall component and natural variable formulations

Interactive visualization

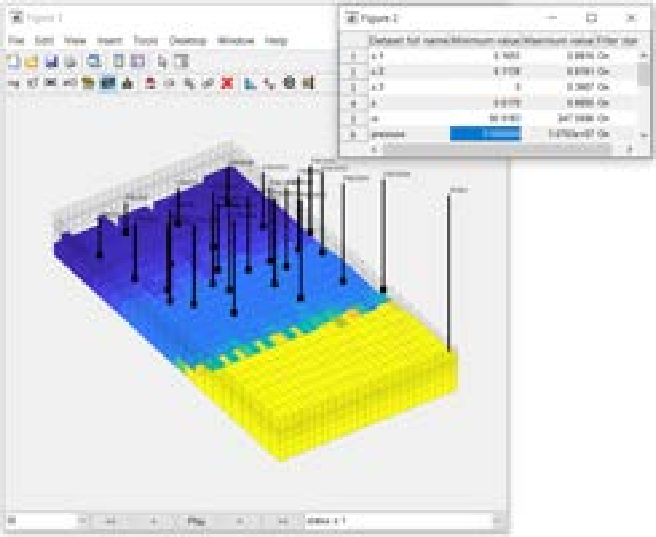


MRST has many functions for script-based plotting of cell and face data, wells, histograms, etc. Tutorial scripts offer plenty of examples

In addition, there are specialized GUIs for interactive

- inspection of functional relationships in fluid models
- volumetric visualization of data sets
- visualization of well responses

Interactive visualization



MRST has many functions for script-based plotting of cell and face data, wells, histograms, etc. Tutorial scripts offer plenty of examples

In addition, there are specialized GUIs for interactive

- inspection of functional relationships in fluid models
- volumetric visualization of data sets
- visualization of well responses

Interactive visualization

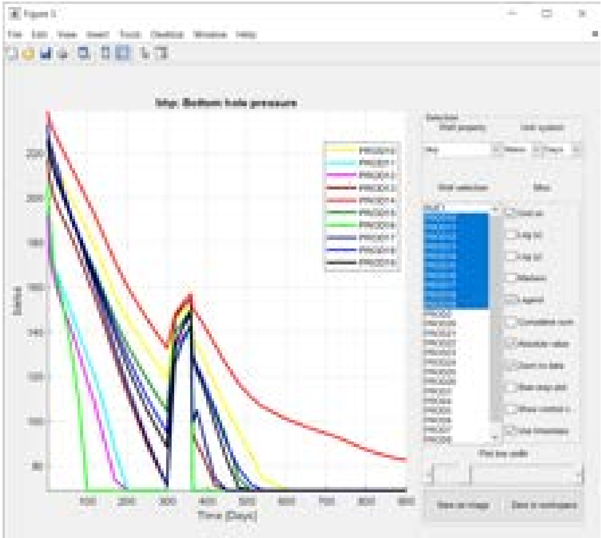


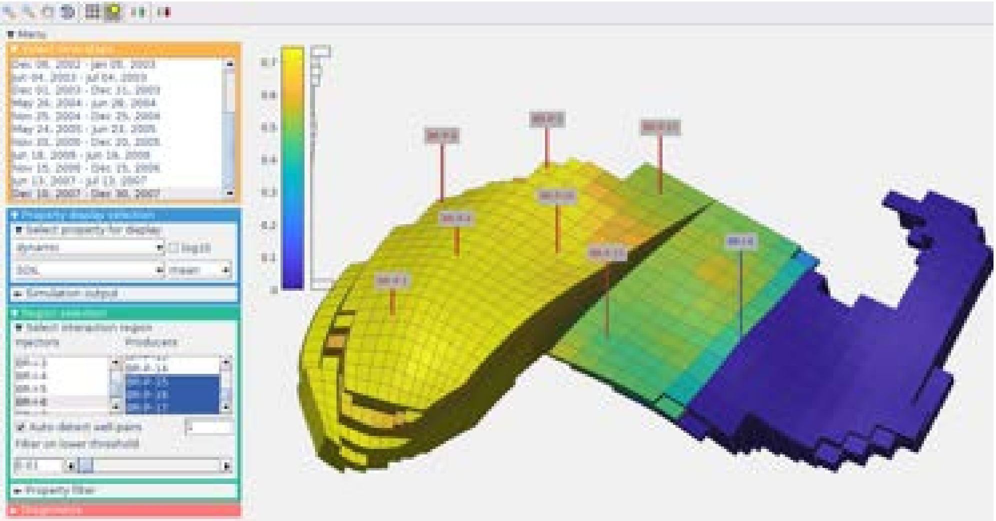
Figure 1
File Edit View Insert Tools Desktop Window Help
bhp: Bottom hole pressure
100
200
300
400
500
600
700
800
900
Time [Days]

MRST has many functions for script-based plotting of cell and face data, wells, histograms, etc. Tutorial scripts offer plenty of examples

In addition, there are specialized GUIs for interactive

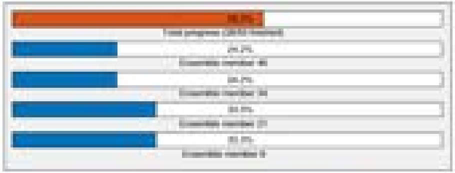
- inspection of functional relationships in fluid models
- volumetric visualization of data sets
- visualization of well responses

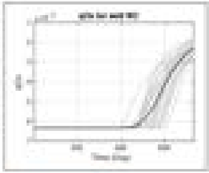
Example: post-processing multiphase simulations

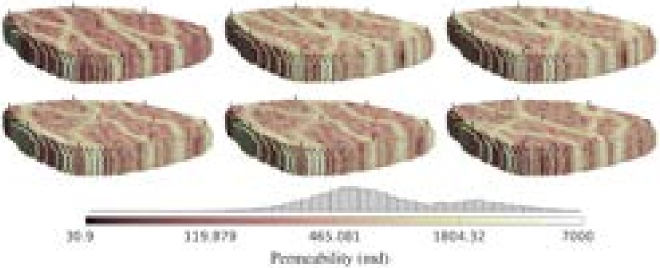

Menu
Dec 28, 2002 - Jan 05, 2003
Jun 04, 2003 - Jul 04, 2003
Dec 01, 2003 - Dec 31, 2003
May 24, 2004 - Jun 28, 2004
Nov 25, 2004 - Dec 25, 2004
May 24, 2005 - Jun 23, 2005
Nov 20, 2005 - Dec 20, 2005
Jun 18, 2006 - Jun 18, 2006
Nov 15, 2006 - Dec 15, 2006
Jun 13, 2007 - Jul 13, 2007
Dec 18, 2007 - Dec 30, 2007
Select property for display
dynamic
□ log10
SOIL
mean
Simulation output
Select interaction region
Injectors
Producers
Auto detect well pairs
Filter on lower threshold
0.01
Property filter
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0

User guide to flow diagnostics postprocessing – Simulations in MRST and ECLIPSE output format. URL: hdl.handle.net/11250/2687032

Systematic management of simulation cases







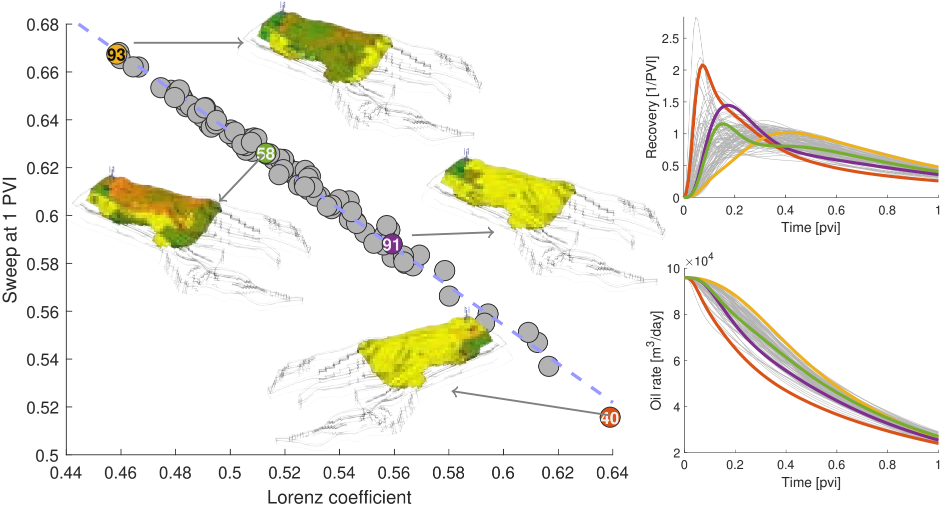
"Packed simulations":

- systematic method of setting up cases
- library of standard test cases
- automatic storage of intermediate simulation results to disk
- automatic restart of aborted simulations
- simple and seamless retrieval of previous simulations

The `ensemble` module

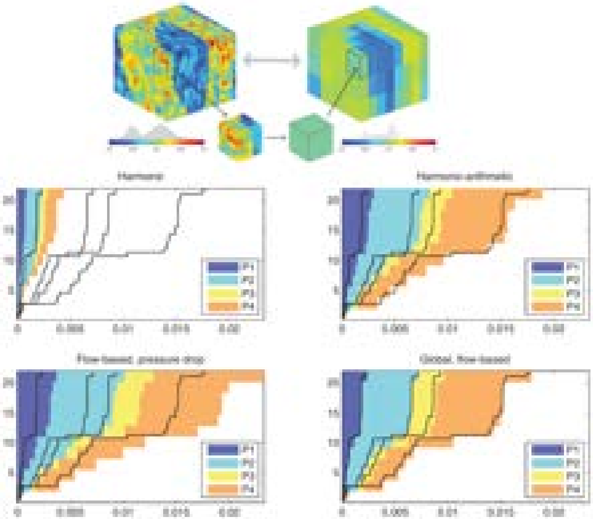
- (multilevel) Monte Carlo simulations
- base problem and stochastic samples
- defines quantities of interest
- functionality for parallel/batch processing

Example: ranking ensemble realizations before simulation



User guide to flow diagnostics in MRST – flow diagnostics preprocessors for model ensembles.
URL: hdl.handle.net/11250/2687239

Model upscaling and model reduction



MRST contains many utility functions and tutorials for upscaling

Single-phase upscaling

- Volume averaging of additive quantities
- Averaging methods: arithmetic, geometric, harmonic, hybrid
- Local flow-based methods: lab-conditions (axial no-flow), periodic
- Global flow-based: generic and specific

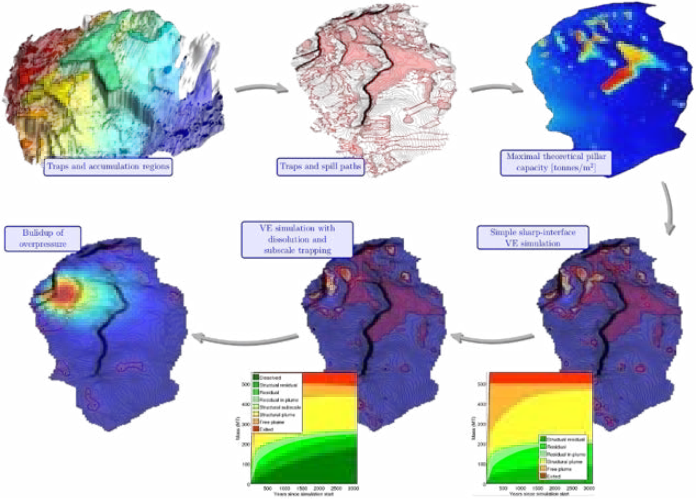
Multiphase upscaling:

- Steady-state methods: general state, capillary- and viscous dominated flow
- Standard flow-based
- Vertical-equilibrium models

MRST CO_2 laboratory

- Rapid simulation of long-term storage (VE modelling)
- Easy to use, intuitive toolchain
- Backed by decades of academic research and development
- Interactively explore geological formations from the Norwegian Continental Shelf
- Visualize migration paths
- Trapping capacities for structural, residual, and solubility trapping
- Pressure build-up/plume migration
- Optimize the placement of injection points and injection schedules

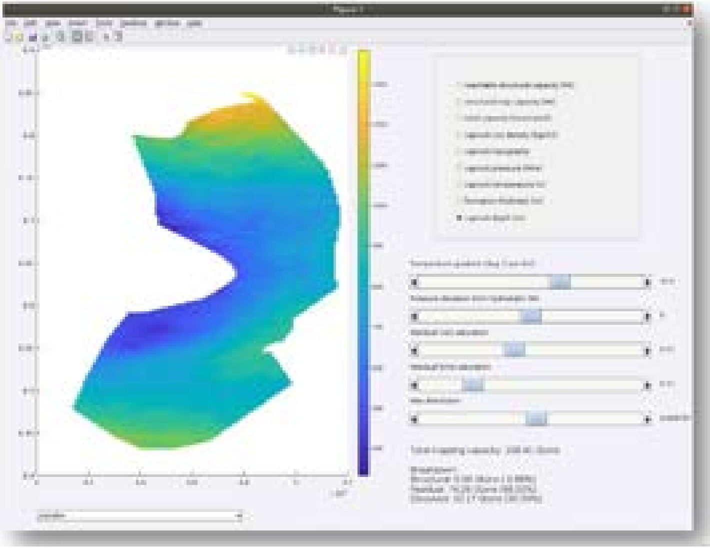
`sintef.no/co2lab`



MRST CO_2 laboratory

- Rapid simulation of long-term storage (VE modelling)
- Easy to use, intuitive toolchain
- Backed by decades of academic research and development
- Interactively explore geological formations from the Norwegian Continental Shelf
- Visualize migration paths
- Trapping capacities for structural, residual, and solubility trapping
- Pressure build-up/plume migration
- Optimize the placement of injection points and injection schedules

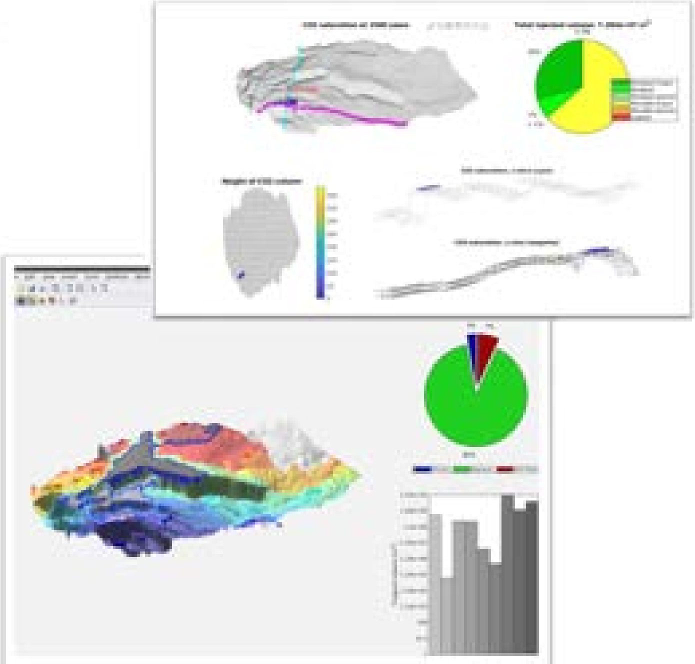
`sintef.no/co2lab`



MRST CO_2 laboratory

- Rapid simulation of long-term storage (VE modelling)
- Easy to use, intuitive toolchain
- Backed by decades of academic research and development
- Interactively explore geological formations from the Norwegian Continental Shelf
- Visualize migration paths
- Trapping capacities for structural, residual, and solubility trapping
- Pressure build-up/plume migration
- Optimize the placement of injection points and injection schedules

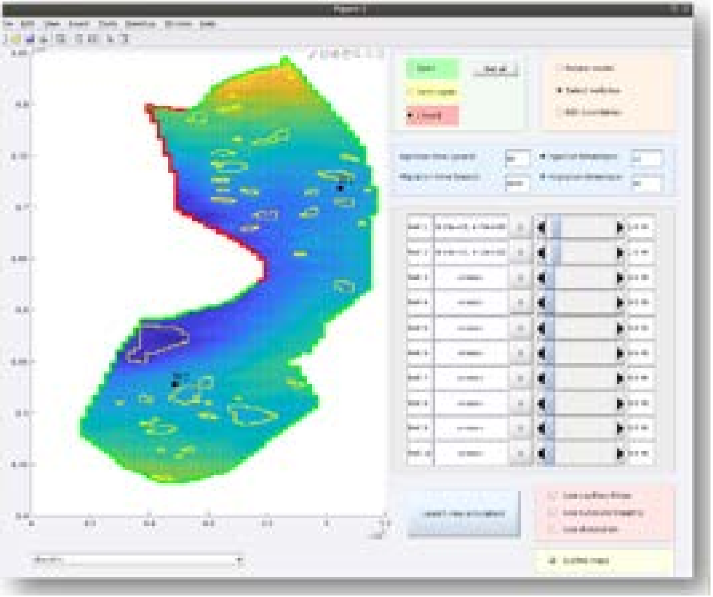
`sintef.no/co2lab`

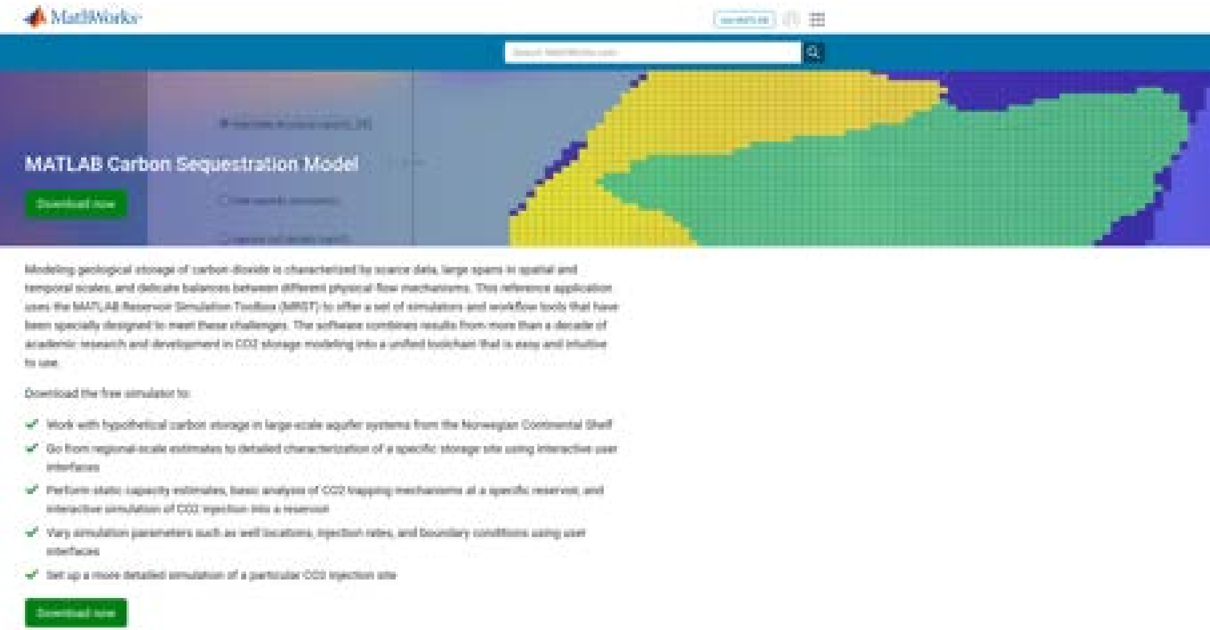


MRST CO_2 laboratory

- Rapid simulation of long-term storage (VE modelling)
- Easy to use, intuitive toolchain
- Backed by decades of academic research and development
- Interactively explore geological formations from the Norwegian Continental Shelf
- Visualize migration paths
- Trapping capacities for structural, residual, and solubility trapping
- Pressure build-up/plume migration
- Optimize the placement of injection points and injection schedules

`sintef.no/co2lab`





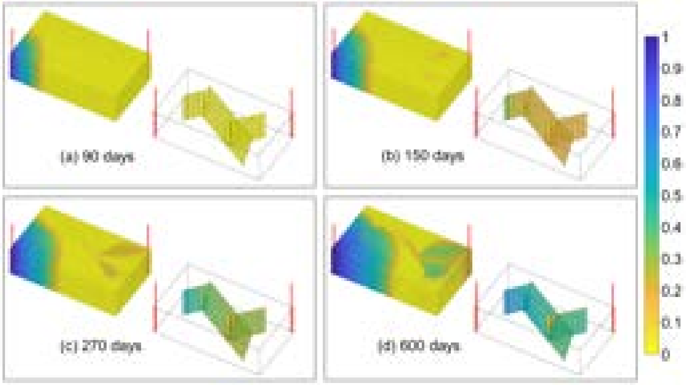
MATLAB Carbon Sequestration Model

Modeling geological storage of carbon dioxide is characterized by scarce data, large spans in spatial and temporal scales, and delicate balances between different physical flow mechanisms. This reference application uses the MATLAB Reservoir Simulation Toolbox (MRST) to offer a set of simulators and workflow tools that have been specially designed to meet these challenges. The software combines results from more than a decade of academic research and development in CO2 storage modeling into a unified toolchain that is easy and intuitive to use.

Download the free simulator to:

- ✓ Work with hypothetical carbon storage in large-scale aquifer systems from the Norwegian Continental Shelf
- ✓ Go from regional-scale estimates to detailed characterization of a specific storage site using interactive user interfaces
- ✓ Perform static capacity estimates, basic analysis of CO2 trapping mechanisms at a specific reservoir, and interactive simulation of CO2 injection into a reservoir
- ✓ Vary simulation parameters such as well locations, injection rates, and boundary conditions using user interfaces
- ✓ Set up a more detailed simulation of a particular CO2 injection site

Embedded discrete fracture models



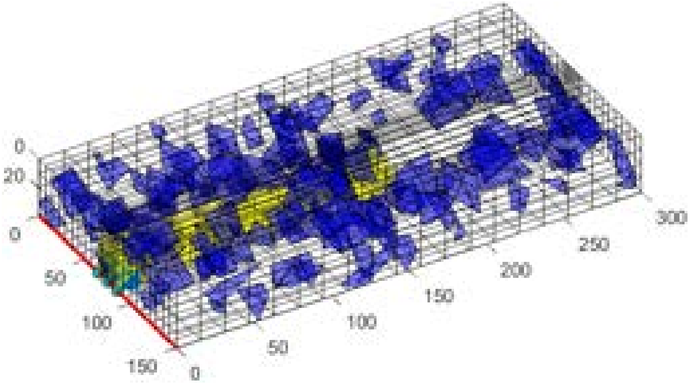
`hfm` module developed jointly with TU Delft and Heriot–Watt University

Very popular method to describe fractured media:

- Fracture network is represented by a separate lower-dimensional grid system embedded inside the volumetric grid
- Conservation equations posted on both the matrix and the fracture grid systems
- Mass exchange among fractures and between matrix and fractures is modelled using a two-point flux type model
- Current implementation does not support non-conductive fractures

The `hfm` module is described in detail in the 2nd MRST textbook

Unconventional oil and gas reservoirs



3rd-party module developed by O. Olorode, B. Wang, and H. U. Rashid, Lousiana State University

Module developed to simulate the special physics of unconventional oil and gas resources

Extends the `compositional` module with

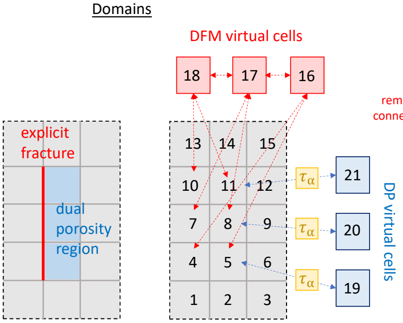
- sorption – adsorption of gas molecules on internal surfaces of the source rock
- molecular diffusion
- stress-sensitive permeability to model closing of fractures

Extends the `hfm` module with

- projection-based embedded discrete fractures (pEDFM) to accurately model fractures with low conductivities

The `shale` module is described in detail in the 2nd MRST textbook

Unified fracture modelling



Connections

removed connections

original connections

1	2
2	3
~~4~~	~~5~~
5	6
~~7~~	~~8~~
8	9
~~10~~	~~11~~
11	12
13	14
14	15
1	4
2	5
3	6
4	7
5	8
6	9
7	10
8	11
9	12
10	13
11	14
12	15

DFM connections

4	16
7	17
10	18
16	5
17	8
18	11
16	17
17	18

DP connections

5	19
8	20
11	21

3rd-party module developed by R. March, C. Meier, F. Doster, and S. Geiger, from Heriot–Watt University

Unified treatment of different modeling approaches

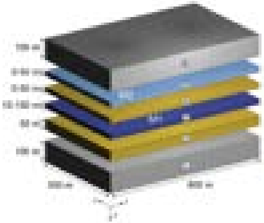
- discrete fracture and matrix methods (DFM)
- multicontinuum approaches relying on transfer functions

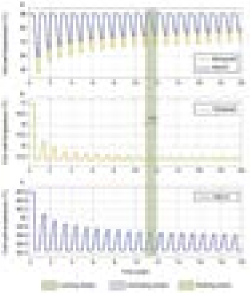
Suitable abstractions enable simple specification of hybrid models

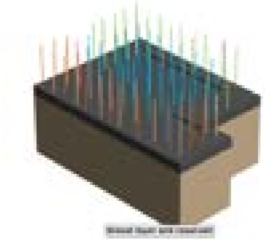
Built on top of the general AD-OO simulator framework

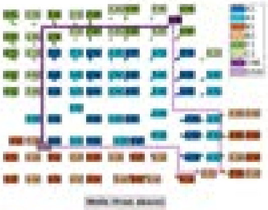
The `hwu-fractures` module is described in detail in the 2nd MRST textbook

Geothermal energy









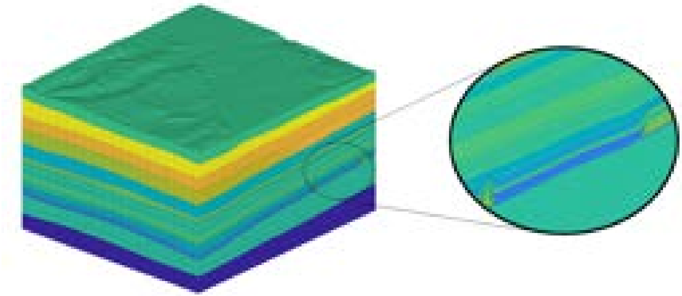
The module combines

- fluid flow
- heat transfer
- tracking of salts

Builds upon the AD-OO framework. Jointly developed with the University of Geneva for high-temperature aquifer thermal storage

Also supports low-temperature, shallow geothermal storage. Scripts are so far only in-house

Geomechanics



Model with separation between flow layers (reservoir) and over/underburden. Layers in reservoir are very thin and contain pinch-outs.

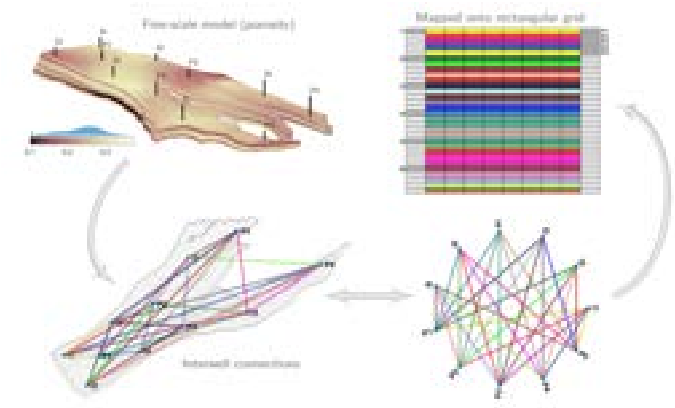
Three modules:

- `vemmech` – virtual elements for linear mechanics
- `ad-mechanics` – coupled flow and mechanics in AD-OO: fully coupled and fixed-stress split scheme
- `dual-continuum-mech` – extension to dual-continuum models, by M. Ashworth, Heriot–Watt University

Also: unsaturated porous media with Richards equation

- by J. Varela et al., University of Bergen
- Richards' equation for partially saturated media)
- Unsaturated poroelasticity (`fv-unsat`)
- Multipoint discretizations: MPFA/MPSA from `re-mpfa`/`fvbiot` or `mpsaw`

Network models: reduced-order/data-driven modelling



Two types of network models:

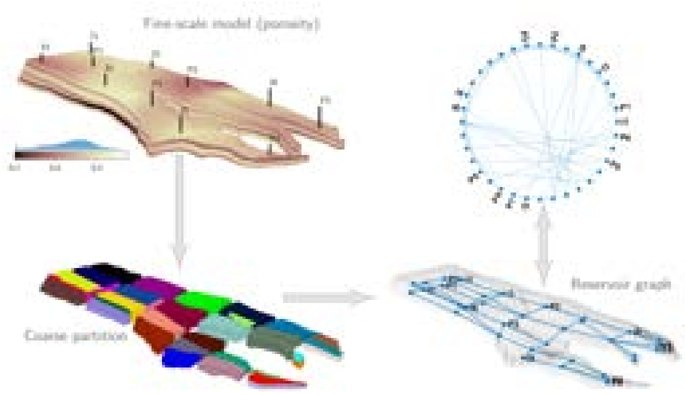
- <span style="color:red">GPSNet – interwell network models</span>
- CGNet – network from volumetric grid

→ computational graph with pore volumes, transmissibilities, well indices, and fluid properties as tunable parameters

Calibration = misfit minimization:

- Classic Gauss–Newton (Levenberg–Marquardt) with misfit Jacobian computed using adjoints
- Quasi-Newton (L-BFGS-B) with gradients computed using adjoints
- Ensemble smoother with multiple data assimilation (ES-MDA)

Network models: reduced-order/data-driven modelling



Two types of network models:

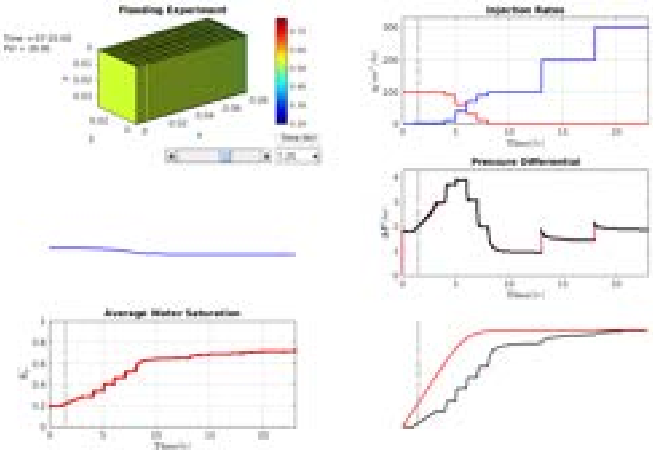
- GPSNet – interwell network models
- CGNet – network from volumetric grid

→ computational graph with pore volumes, transmissibilities, well indices, and fluid properties as tunable parameters

Calibration = misfit minimization:

- Classic Gauss–Newton (Levenberg–Marquardt) with misfit Jacobian computed using adjoints
- Quasi-Newton (L-BFGS-B) with gradients computed using adjoints
- Ensemble smoother with multiple data assimilation (ES-MDA)

Special core analysis (SCAL)



3rd-party module developed by the research group of Prof. Holger Ott, Montanuniversität Leoben

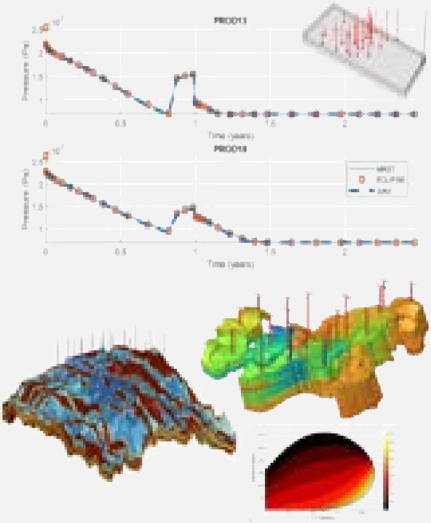
Functionality to simulate SCAL experiments: drainage and imbibition steady-state, unsteady-state, and centrifuge.

Using this module:

- SCAL experiments are designed before they are performed using forward simulations
- relative permeability and capillary pressure/saturation functions are calculated from measured SCAL data using history matching simulations

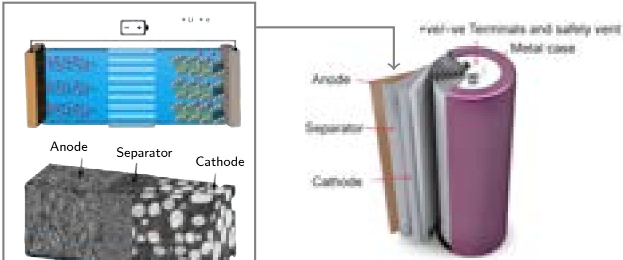
Jutul: new high-performance simulator written in Julia

- Based on ideas from MRST and OPM Flow
- Aimed at high-computational performance and support of both multicore and many-core (GPU) architectures
- Relies on MRST for preprocessing simulation cases
- Run from MRST via files or via a daemon process
- Flow physics: immiscible, black-oil, EOS compositional multiphase flow, multisegment wells
- Significantly faster assembly than OPM, AD-GPRS, etc.
- Reservoir simulator:
  `github.com/sintefmath/JutulDarcy.jl`
- Examples:
  `github.com/sintefmath/JutulExamples.jl`

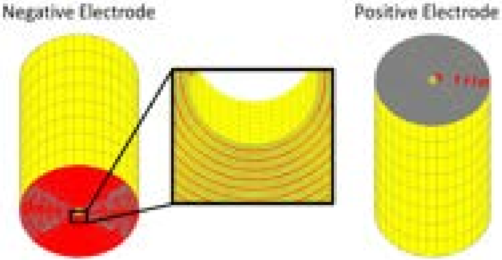


Verified against simulators on various cases

BattMo: battery modelling

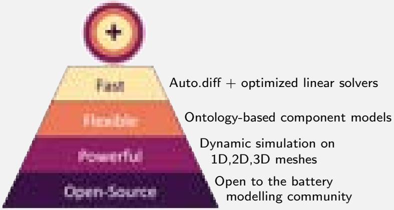


0.1 mm × 50 mm × 5 m



BattMo – continuum modelling of batteries and electrochemical devices:

- Conservation of charge, mass, and energy
- First-order finite-volume methods



Auto.diff + optimized linear solvers

Ontology-based component models

Dynamic simulation on 1D,2D,3D meshes

Open to the battery modelling community

MRST

github.com/BattMoTeam/BattMo

How do we see MRST in 5 to 10 years?

Our goal is to

- keep MRST a community tool for rapid prototyping
- support MRST as easy-go-to black-box simulators
- increase focus on differentiable programming
- encourage more content contributed by others

Plan to move to GitHub for visibility and reorganize the module structure for clarity:

- basic functionality used by everybody
- advanced simulation tools
- historic modules (only partially maintained)
- third-party modules
- contributed content (not modules)

Funding:

- Research Council of Norway
- SINTEF
- Chevron, Ecopetrol, Eni, Equinor, ExxonMobil, Ruden, Shell, SLB, TotalEnergies, Wintershall DEA, . . .
- Equinor: gold open access for the MRST textbook