Reservoir Simulation
What and why?

Reservoir simulation is the means by which a numerical model of the petrophysical characteristics of a hydrocarbon reservoir is used to analyze and predict fluid behavior in the reservoir over time.

Reservoir simulation is used as a basis for decisions regarding development of reservoirs and management during production. To this end, one needs to

- predict reservoir performance from geological descriptions and constraints,
- fit geological descriptions to static and dynamic data,
- assess uncertainty in predictions,
- optimize production strategies,
Reservoir modelling is a true *multiscale* discipline:
- Measurements and models on a large number of scales
- Large number of models
- Complex grids with a large number of parameters
- High degree of uncertainty

There is always a need for faster and more accurate simulators that use all available geological information.

Physical Scales in Porous Media Flow
One cannot resolve them all at once

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.
Physical Scales in Porous Media Flow

Microscopic: the scale of individual sand grains

Flow in individual pores between sand grains

Geological: the meter scale of layers, depositional beds, etc

Porous sandstones often have repetitive layered structures, but faults and fractures caused by stresses in the rock disrupt flow patterns
Physical Scales in Porous Media Flow

Choosing a scale for modelling

Scales in reservoirs

- Pore scale
- Core scale
- Simulation model
- Geological model
Geomodels:
- are articulations of the experts’ perception of the reservoir
- describe the reservoir geometry (horizons, faults, etc)
- give rock parameters (e.g., permeability $K$ and porosity $\phi$) that determine the flow

In the following: the term “geomodel” will designate a grid model where rock properties have been assigned to each cell.

---

Flow Simulation
Model problem: incompressible, single phase

Consider the following model problem

Darcy’s law: $v = -K(\nabla p - \rho g \nabla D)$,

Mass balance: $\nabla \cdot v = q$ in $\Omega$,

Boundary conditions: $v \cdot n = 0$ on $\partial \Omega$.

The multiscale structure of porous media enters the equations through the absolute permeability $K$, which is a symmetric and positive definite tensor with uniform upper and lower bounds.

We will refer to $p$ as pressure and $v$ as velocity.
Flow Simulation
The impact of rock properties

Rock properties are used as parameters in flow models
- Permeability $K$ spans many length scales and have multiscale structure
  \[
  \max K / \min K \sim 10^3 - 10^{10}
  \]
- Details on all scales impact flow

Challenges:
- How much details should one use?
- Need for good linear solvers, preconditioners, etc.

Gap in resolution and model sizes

Gap in resolution:
- High-resolution geomodels may have $10^6 - 10^{10}$ cells
- Conventional simulators are capable of about $10^5 - 10^6$ cells

Traditional solution: upscaling of parameters
Assume that $u$ satisfies the elliptic PDE:
\[
-\nabla (K(x) \nabla u) = f.
\]
Upscaling amounts to finding a new field $K^*(\bar{x})$ on a coarser grid such that
\[
-\nabla (K^*(\bar{x}) \nabla u^*) = \bar{f},
\]
\[
u^* \sim \bar{u}, \quad q^* \sim \bar{q}.
\]
How do we represent fine-scale heterogeneities on a coarse scale?

- Combinations of arithmetic, geometric, harmonic averaging
- Power averaging \( \left( \frac{1}{|V|} \int_V a(x)^p \, dx \right)^{1/p} \)
- Equivalent permeabilities \( a_{xx}^* = -Q_x L_x / \Delta P_x \)

There are many difficulties associated with upscaling

- Bottleneck in the workflow
- Loss of details
- Lack of robustness
- Need for resampling for complex grid models
- Not obvious how to extend the ideas to 3-phase flows

Need for fine-scale computations?

In the future: need for multiphysics on multiple scales?
Fluid Simulations Directly on Geomodels

**Research vision:**
Direct simulation of complex grid models of highly heterogeneous and fractured porous media - a technology that bypasses the need for upscaling.

**Applications:**
Huge models, multiple realizations, prescreening, validation, optimization, data integration, ..

To this end, we seek a methodology that
- incorporates small-scale effects into coarse-scale system;
- gives a detailed image of the flow pattern on the fine scale, without having to solve the full fine-scale system;
- is robust, conservative, accurate, and efficient.

---

**Multiscale Pressure Solvers**
Efficient flow solution on complex grids – without upscaling

**Basic idea:**
- Upscaling and downscaling in one step
- Pressure varies smoothly and can be resolved on coarse grid
- Velocity with subgrid resolution
From Upscaling to Multiscale Methods

**Standard upscaling:**

- Coarse grid blocks:
- Flow problems:
  - $P=1$ $P=0$
  - $P=1$

**Multiscale method:**

- Coarse grid blocks:
- Flow problems:
  - $p=1$ $p=1$
  - $q=1$

---

From Upscaling to Multiscale Methods

**Standard upscaling:**

- Coarse grid blocks:
- Flow problems:
  - $p=1$ $p=0$ $p=1$

**Multiscale method:**

- Coarse grid blocks:
- Flow problems:
  - $p=1$ $p=1$ $q=1$
The Multiscale Mixed Finite-Element Method

**Standard finite-element method (FEM):**
Piecewise polynomial approximation to pressure, \( \int l \nabla K \nabla p \, dx = \int l q \, dx \)

**Mixed finite-element methods (MFEM):**
Piecewise polynomial approximations to pressure and velocity
\[
\int_{\Omega} k^{-1}v \cdot u \, dx - \int_{\Omega} p \nabla \cdot u \, dx = \int_{\Omega} k^{-1} \rho g \nabla D \cdot u \, dx \quad \forall u \in U,
\]
\[
\int_{\Omega} l \nabla \cdot v \, dx = \int_{\Omega} ql \, dx \quad \forall l \in V.
\]

**Multiscale mixed finite-element method (MsMFEM):**
Velocity approximated in a (low-dimensional) space \( V^{ms} \) designed to embody the impact of fine-scale structures.

---

Multiscale Mixed Finite Elements

**Grids and basis functions**

Assume we are given a *fine* grid with permeability and porosity attached to each fine-grid block:

![Image of a fine grid]

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

- For each coarse block \( T_i \), there is a basis function \( \phi_i \in U \).
- For each coarse edge \( \Gamma_{ij} \), there is a basis function \( \psi_{ij} \in V^{ms} \).
Saddle-point problem:
\[
\begin{pmatrix}
B & C \\
C^T & 0
\end{pmatrix}
\begin{pmatrix}
v \\
p
\end{pmatrix}
=
\begin{pmatrix}
f \\
g
\end{pmatrix},
\]

\[b_{ij} = \int_{\Omega} \psi_i k^{-1} \psi_j \, dx,\]

\[c_{ij} = \int_{\Omega} \phi_j \nabla \cdot \psi_i \, dx.\]

Basis \( \phi_j \) for pressure: equal one in cell \( j \), zero otherwise

Basis \( \psi_i \) for velocity:

1. order Raviart–Thomas:

Multiscale:

Velocity basis function \( \psi_{ij} \): unit flow through \( \Gamma_{ij} \) defined as

\[
\nabla \cdot \psi_{ij} = \begin{cases}
w_i(x), & \text{for } x \in T_i, \\
-w_j(x), & \text{for } x \in T_j,
\end{cases}
\]

and no flow \( \psi_{ij} \cdot n = 0 \) on \( \partial(T_i \cup T_j) \).

Global velocity:

\[v = \sum_{ij} v_{ij} \psi_{ij}, \text{ where } v_{ij} \text{ are (coarse-scale) coefficients.}\]
Multiscale Simulation versus Upscaling
10th SPE Comparative Solution Project

- Geomodel: $60 \times 220 \times 85 \approx 1.1$ million grid cells, $\max K_x / \min K_x \approx 10^7$, $\max K_z / \min K_z \approx 10^{11}$
- Simulation: 2000 days of production (2-phase flow)

Commercial (finite-difference) solvers: incapable of running the whole model

Upscaling results reported by industry

- single-phase upscaling
- two-phase upscaling
Multiscale Simulation versus Upscaling
10th SPE Comparative Solution Project

Runtime: 2 min 22 sec on 2.4 GHz desktop PC

Robustness
SPE10, Layer 85 (60 × 220 Grid)
Comparison of Multiscale and Upscaling Methods

1. Local-global upscaling (Durlofsky et al)
   - global boundary conditions, iterative improvement (bootstrap)
   - reconstruction of fine-grid velocities

2. Multiscale mixed finite elements (Chen & Hou, ...)
   - multiscale basis functions for velocity
   - coarse-scale pressure

3. Multiscale finite-volume method (Jenny, Tchelepi, Lee, ...)
   - multiscale basis functions for pressure
   - reconstruction of velocity on fine grid

4. Numerical subgrid upscaling (Arbogast, ...)
   - direct decomposition of the solution, \( V = V_c \oplus V_f \)
   - RT0 on fine scale, BDM1 on coarse

Comparison of Multiscale and Upscaling Methods
SPE 10, individual layers

Saturation errors at 0.3 PVI on 15 × 55 coarse grid

![Graph showing saturation errors for different methods.](image)
Comparison of Multiscale and Upscaling Methods

Average saturation errors on Upper Næss formation (Layers 36–85)

**Cartesian coarse grids:**
Multiscale methods give enhanced accuracy when subgrid information is exploited.

**Fluid transport:**
- Coarse grid
- Fine grid
Comparison of Multiscale and Upscaling Methods
MsMFEM versus upscaling on complex coarse grids

Complex coarse grid-block geometries:
MsMFEM is more accurate than upscaling, also for coarse-grid simulation.

![Graphs showing Coarse-grid velocity and saturation errors for different grid sizes and methods.]

Up-gridded $30 \times 30 \times 333$ corner-point grid with layered log-normal permeability

Computational Complexity
Order-of-magnitude argument

Assume:
- Grid model with $N = N_s \times N_c$ cells:
  - $N_c$ number of coarse cells
  - $N_s$ number of fine cells in each coarse cell
- Linear solver of complexity $O(m^\alpha)$ for $m \times m$ system
- Negligible work for determining local b.c., numerical quadrature, and assembly (can be important, especially for NSUM)

Direct solution
$N^{\alpha}$ operations for a two-point finite volume method

MsMFEM
- Computing basis functions: $D \cdot N_c \cdot (2N_s)^\alpha$ operations
- Solving coarse-scale system: $(D \cdot N_c)^\alpha$ operations
Computational Complexity

Example: $128 \times 128 \times 128$ fine grid

Comparison with algebraic multigrid (AMG), $\alpha = 1.2$

Comparison with less efficient solver, $\alpha = 1.5$
**Multiphase Flow**

Time-dependent problems: \( \nabla (K(x)\lambda(S)\nabla p) = q(S) \)

Direct solution may be more efficient, so why bother with multiscale?

- Full simulation: \( O(10^2) \) time steps.
- Basis functions need not always be recomputed

Also:
- Possible to solve very large problems
- Easy parallelization

---

**Two-Phase Flow**

Example: quarter five-spot, Layer 85 from SPE 10, coarse grid: 10 x 22

Water cuts obtained by never updating basis functions:

- Favorable (\( M = 0.1 \))
- Unfavorable (\( M = 10.0 \))
Improved accuracy by adaptive updating of basis functions:

Assimilation of production data to calibrate model

- 1 million cells, 32 injectors, and 69 producers
- 2475 days ≈ 7 years of water-cut data
- 6 iterations in data integration method
- 7 forward simulations, 15 pressure updates each

Computation time (on desktop PC):
- Original method: ~ 40 min (pressure solver: 30 min)
- Multiscale method: ~ 17 min (pressure solver: 7 min)
Another challenge:
- Industry-standard grids are often nonconforming and contain skewed and degenerate cells
- There is a trend towards unstructured grids
- Standard discretization methods produce wrong results on skewed and rough cells

Corner point:  
Tetrahedral:  
PEBI:

Corner-Point Grids
Industry standard for modelling complex reservoir geology

Specified in terms of:
- areal 2D mesh of vertical or inclined pillars
- each volumetric cell is restricted by four pillars
- each cell is defined by eight corner points, two on each pillar
Discretisation on Corner-Point Grids
Exotic cell geometries from a simulation point-of-view

Skew and deformed grid blocks:

Non-matching cells:

- Can use standard MFEM provided that one has mappings and reference elements
- Can subdivide corner-point cells into tetrahedra
- We use mimetic finite differences (recent work by Brezzi, Lipnikov, Shashkov, Simoncini)

Let \( u, v \) be piecewise linear vector functions and \( u, v \) be the corresponding vectors of discrete velocities over faces in the grid, i.e.,

\[
v_k = \frac{1}{|e_k|} \int_{e_k} v(s) \cdot n \, ds
\]

Then the block \( B \) in the mixed system satisfies

\[
\int_\Omega v^T K^{-1} u = v^T Bu = \sum_{E \in \Omega} v^T B_E u_E
\]

The matrices \( B_E \) define discrete inner products

Mimetic idea:

Replace \( B_E \) with a matrix \( M_E \) that mimics some properties of the continuous inner product (SPD, globally bounded, Gauss-Green for linear pressure)
Mimetic Finite Difference Methods
General method applicable to general polyhedral cells

Standard method + skew grids = grid-orientation effects

\( K \): homogeneous and isotropic, symmetric well pattern
\( \rightarrow \) symmetric flow

Streamlines with standard method
Streamlines with mimetic method

Water-cut curves for two-point FVM
Water-cut curves for mimetic FDM

Multiscale Mixed Finite Elements
An automated alternative to upscaling?

Coarse grid = union of cells from fine grid
MsMFEMs allow fully automated coarse gridding strategies: grid blocks need to be connected, but can have arbitrary shapes.

Uniform up-gridding: grid blocks are shoe-boxes in index space.
Model is courtesy of Alf B. Rustad, Statoil
Multiscale Mixed Finite Elements
Examples of exotic grids

Ideal for coupling with well models

Fine grid to annulus, one coarse block for each well segment \(\implies\) no well model needed.
### Summary

**Advantages of multiscale mixed/mimetic pressure solvers**

<table>
<thead>
<tr>
<th>Ability to handle industry-standard grids</th>
</tr>
</thead>
<tbody>
<tr>
<td>- highly skewed and degenerate cells</td>
</tr>
<tr>
<td>- non-matching cells and unstructured connectivities</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Compatible with current solvers</th>
</tr>
</thead>
<tbody>
<tr>
<td>- can be built on top of commercial/inhouse solvers</td>
</tr>
<tr>
<td>- can utilize existing linear solvers</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>More efficient than standard solvers</th>
</tr>
</thead>
<tbody>
<tr>
<td>- faster and requires less memory than fine-grid solvers</td>
</tr>
<tr>
<td>- automated generation of coarse simulation grids</td>
</tr>
<tr>
<td>- easy to parallelize</td>
</tr>
</tbody>
</table>