Stokes and porous media flow solvers

Barbara Wohlmuth*, Rainer Helmig*, Uli Rüde*
B. Gmeiner*, T. Horger*, M. Huber*, A. Lauser*, C. Waluga*


wohlmuth@ma.tum.de

http://www-m2.ma.tum.de

January 30, 2015; National University of Singapore
High Performance and Parallel Computing for Materials Defects and Multiphase
Overview

- **Large scale simulation**
  - FE for Stokes
    - Local mass conservation
    - Gradient versus strain
  - New challenges
    - Hardware faults and recovery
    - Uncertainty and scheduling

- **Porous media applications**
  - Model variational inequality
    - Primal-dual formulation
    - Semi-smooth Newton
  - Multi-phase multi-component system
    - Dynamic capillary pressure
    - Phase change
The mantle convection model

The **physical model** consists of conservation of momentum, mass and energy

\[- \text{div} \sigma = \rho g\]
\[\text{div}(\rho u) = 0\]
\[\partial_t (\rho e) + \text{div}(\rho e u) = -\text{div} \, q + \rho H + \sigma : \dot{\varepsilon}\]

**Key quantities:** velocity $u$, temperature $T$, pressure $p$, and the **mantle viscosity** $\mu$. The **density** $\rho$ is given by the **mineralogy** via an equation of state:

$$\rho = \rho(p, T)$$

The **rheology** of the mantle is an active field:

$$\sigma = 2\mu(\dot{\varepsilon} - \frac{1}{3} \text{tr} \dot{\varepsilon} \cdot I) - pI,$$

with $\mu = \mu(r, T, \dot{\varepsilon})$

**Notation:**
- $\sigma$: stress tensor
- $\rho$: density
- $g$: gravitational acceleration
- $u$: velocity
- $e$: internal energy
- $q$: heat flux per unit area
- $H$: volumetric radiogenic heat production rate
- $\dot{\varepsilon}$: rate of strain tensor
From a million to a billion

make it lean

push it hard

make it fast
Linear finite elements for mantle-convection

Advantages of linear finite elements

- equal-order Stokes discretizations show superconvergence for velocity and pressure \( \mathcal{O}(h^{3/2}) \)-convergence rate; [Eichel/Tobiska/Xie 2011]
- systems with \( \mathcal{O}(10^{12}) \) unknowns can be solved within 100 seconds on current peta-scale machines

Disadvantages of linear finite elements

- they are **not locally** mass-conservative (spurious sources/sinks)
- they are **not stable** for the Stokes problem (require additional stabilization)
- poorly suitable for convection-dominated transport (high numerical diffusion)
Why do we head for excellent strong scalability?

Why is it not easy on peta-scale machines?
**Strong scalability**

**SuperMUC:** Run-times of strong scaling experiments from 1920 to 30 720 cores.

4.4 $\cdot 10^{10}$ DOF,
Residual reduction: $10^3$ (A) and $10^7$ (B)
Pressure correction: 7 (A) and 22 (B)

**JUQUEEN:** Run-times of strong scaling experiments from 3 840 to 122 880 cores.
When is even perfect weak scalability not enough?

How close do we get on peta-scale machines?
Weak scalability on SuperMUC

- **Pressure correction scheme** [Verfürth 84], [Grinevich+Olshanskii09]

<table>
<thead>
<tr>
<th>Nodes</th>
<th>Threads</th>
<th>DOF</th>
<th>Resolution</th>
<th>Time (A)</th>
<th>(B)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4</td>
<td>$4.8 \cdot 10^7$</td>
<td>32 km</td>
<td>16 s</td>
<td>51 s</td>
</tr>
<tr>
<td>2</td>
<td>30</td>
<td>$6.4 \cdot 10^8$</td>
<td>16 km</td>
<td>20 s</td>
<td>63 s</td>
</tr>
<tr>
<td>15</td>
<td>240</td>
<td>$5.2 \cdot 10^9$</td>
<td>8 km</td>
<td>24 s</td>
<td>73 s</td>
</tr>
<tr>
<td>120</td>
<td>1920</td>
<td>$4.4 \cdot 10^{10}$</td>
<td>4 km</td>
<td>27 s</td>
<td>80 s</td>
</tr>
<tr>
<td>960</td>
<td>15360</td>
<td>$3.4 \cdot 10^{11}$</td>
<td>2 km</td>
<td>34 s</td>
<td>107 s</td>
</tr>
<tr>
<td>7680</td>
<td>122880</td>
<td>$2.8 \cdot 10^{12}$</td>
<td>1 km</td>
<td>41 s</td>
<td>131 s</td>
</tr>
</tbody>
</table>

- **Operator count** for different solvers: expected gain in run-time is a factor of two
Single node performance

Operator evaluations:
- **Fully stencil based** for constant coefficients (30 FLOPS per grid point and operator evaluation)
- **Matrix-free** for variable coefficients (efficient assembling *on-the-fly*)

Execution-Cache-Memory model (ECM) [Hager et al, 10-13]:

<table>
<thead>
<tr>
<th>Intel Sandy Bridge core</th>
<th>assumed limiting resource</th>
<th>FLOPS throughput</th>
<th>memory bandwidth</th>
<th>ECM model</th>
<th>measured</th>
</tr>
</thead>
<tbody>
<tr>
<td>const. coeff. (CC)</td>
<td></td>
<td>720 MLups/s</td>
<td>667 MLups/s</td>
<td>159-189 MLups/s</td>
<td>176 MLups/s</td>
</tr>
<tr>
<td>var. coeff. (VC)</td>
<td></td>
<td>79.4 MLups/s</td>
<td>500 MLups/s</td>
<td>40.0-42.5 MLups/s</td>
<td>39.5 MLups/s</td>
</tr>
</tbody>
</table>
Eight core Sandy Bridge processor

![Graph showing performance in MLups/s vs # cores for constant and variable coefficients ECM models.]
Communication performance

The overall performance does not only depend on the node performance but strongly on the communication performance which is limited by:

- **Bandwidth (BW)**
  estimate the transferred data on each level: effects mainly finer grids

- **Latency:**
  - **Direct communication (DCL)**
    investigate the message throughput: effects mainly coarser grids
  - **Collective communication (CCL)**
    MPI overhead and hop latency: effects mainly solvers with global communication

<table>
<thead>
<tr>
<th></th>
<th>Model prediction</th>
<th>Scalasca</th>
</tr>
</thead>
<tbody>
<tr>
<td>BW</td>
<td>5.45 s</td>
<td>6.97 s</td>
</tr>
<tr>
<td>DCL</td>
<td>0.75 s</td>
<td>1.68 s</td>
</tr>
<tr>
<td>CCL</td>
<td>$6.36 \cdot 10^{-3}$ s</td>
<td>0.99 s</td>
</tr>
</tbody>
</table>
Run-time break up on JUQUEEN

<table>
<thead>
<tr>
<th>Method</th>
<th>Computation</th>
<th>Communication</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multigrid, CL, scalar product</td>
<td>0.1 s</td>
<td>1.0 s</td>
</tr>
<tr>
<td>Multigrid, CL</td>
<td>4.5 s</td>
<td>1.6 s</td>
</tr>
<tr>
<td>Multigrid, w.o. finest level</td>
<td>8.8 s</td>
<td>4.6 s</td>
</tr>
<tr>
<td>Multigrid</td>
<td>28.2 s</td>
<td>8.5 s</td>
</tr>
<tr>
<td>Pressure correction</td>
<td>38.2 s</td>
<td>12.2 s</td>
</tr>
<tr>
<td>Overall</td>
<td>42.6 s</td>
<td>15.8 s</td>
</tr>
</tbody>
</table>

Observation: Flops are not yet for free
Naive coarse grid solver works well
Textbook multigrid efficiency

[Brandt 98]: “Textbook multigrid efficiency means solving a discrete PDE problem with a computational effort that is only a small (less than 10) multiple of the operation count associated with the discretized equations itself.”

Parallel textbook MG efficiency factor $E(N, U)$: $T(N, U)$ time to solution

$$E(N, U) := T(N, U) \frac{U}{N} \mu_s$$

$N$ number of unknowns, $U$ number of processor cores, $\mu_s$ single core performance

Parallel TME factors for different problem sizes and settings

<table>
<thead>
<tr>
<th>Grid points</th>
<th>2 · $10^6$</th>
<th>3 · $10^7$</th>
<th>9 · $10^9$</th>
<th>2 · $10^{11}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Processor cores $U$</td>
<td>1</td>
<td>16</td>
<td>4 096</td>
<td>16 384</td>
</tr>
<tr>
<td>Ratio $N/U$</td>
<td>2 · $10^6$</td>
<td>2 · $10^6$</td>
<td>2 · $10^6$</td>
<td>$10^7$</td>
</tr>
<tr>
<td>(constant coefficient) - FMG(2,2)</td>
<td>15</td>
<td>22</td>
<td>26</td>
<td>22</td>
</tr>
<tr>
<td>(variable coefficient) - FMG(2,2)</td>
<td>11</td>
<td>13</td>
<td>15</td>
<td>13</td>
</tr>
<tr>
<td>(Stokes) - FMG(2,1)</td>
<td>64</td>
<td>100</td>
<td>118</td>
<td>-</td>
</tr>
</tbody>
</table>
Simplified earth-mantle simulation

temperature plumes over time (6.5 \cdot 10^9 unknowns, 10000 time steps, run time 7 days)

Low cost mid size cluster (288 compute cores arranged in 9 nodes) of LSS at FAU
Which picture does not fit?
Mass conservation via dual mesh

**Observation:** Most low order discretization do not yield $\text{div } \mathbf{u}_h = 0$ but

$$\int_{\Omega} \text{div } \mathbf{u}_h \phi_p = c_h(p_h, \phi_p) + g_h(\phi_p)$$

**Idea:** Use barycentric dual $\mathcal{B}_h$ mesh associated with $\mathcal{T}_h$

Define the skeleton $\Gamma_h := \bigcup \partial B \setminus \partial \Omega$, $B \in \mathcal{B}_h$, with a fixed normal vector $\mathbf{n}$
Locally conservative mass fluxes on dual boxes

Corrected mass-fluxes: \( j_{ij}(u_h, p_h) := u_h \cdot n|_{f_{ij}} - \kappa_{ij}(u_h, p_h) \), \( f_{ij} := \partial B_i \cap \partial B_j \),

where \( \kappa_{ij} \) is a locally defined mass-defect correction on each primal element \( T \)

\[
\kappa_{ij}^T(u_h, p_h) := \frac{1}{(d + 1)|f_{ij}^T|}(\mathcal{R}_{i}^T(u_h, p_h) - \mathcal{R}_{j}^T(u_h, p_h)),
\]

where the residuals \( \mathcal{R}_{i}^T \) are defined as

\[
\mathcal{R}_{i}^T(u_h, p_h) := \int_{B_i^T} \text{div } u_h \, dx - \int_{T} \text{div } u_h \phi_i \, dx - c_T(p_h, \phi_i) - g_T(\phi_i),
\]
Main results

**Theorem:** Under the local consistency assumptions \( c_T(\cdot, 1) = g_T(1) = 0, T \in \mathcal{T}_h, \) we obtain a **order preserving local mass** conservation on each dual cell

\[
\int_{\partial B} j(u_h, p_h) \, ds = 0, \quad B \in \mathcal{B}_h.
\]

**Examples:**

- for a **stable** pairing (e.g., \( P_1\)-iso-\( P_2 - P_1 \), Hood-Taylor, etc.), the correction \( \kappa_{ij}^T \) **does not depend on** \( p_h \) and can be directly computed by quadrature.

\[
\kappa_{ij}^T(u_h) := \frac{1}{(d + 1)|f_{ij}^T|} \left\{ \int_{B_i^T} \text{div} \, u_h \, dx - \int_{B_j^T} \text{div} \, u_h \, dx - \int_T \text{div} \, u_h (\phi_i - \phi_j) \, dx \right\},
\]

- for a **stabilized** \( P_1 - P_1 \) element the correction **does not depend on** \( u_h \), e.g., the popular pressure-stabilization [Brezzi/Pitkäranta 1984] reads

\[
c_T(p_h, q_h) = \int_T \delta_T \nabla p_h \cdot \nabla q_h \, dx, \quad \text{and we find} \quad \kappa_{ij}^T(p_h) = \delta_T \nabla p_h |_T \cdot n_{ij}^T,
\]
Blankenbach example: Typ 1

Non-linear viscosity model: $\mu(T) := \exp(-\ln(1000)\frac{T}{2})$

Initial condition: $T_0 = 2(1 - x_2^2) + 0.01 \cos(4\pi x_1) \sin(2\pi x_2), \ \Omega := (0, 1)^2$
Blankenbach example: Typ II

Non-linear viscosity model: \( \mu(x_2, T) := \exp(-\ln(16384)\frac{T}{2} + \ln(64)(1 - x_2)) \)

Local flux correction yields higher accuracy on coarse meshes
Cylindrical channel with spheres

Inflow:
\[ u = g, \quad c = 1 \]

Free outflow:
\[ u = 0 \]

Initial concentration: \( c(t = 0) = 0 \).
Cylindrical channel with spheres

Compare the concentration solutions at $t=3$ for different mesh sizes:

**uncorrected FEM/FV coupling**

**corrected FEM/FV coupling**

Top to bottom: 6,689, 49,477, and 392,753 cells (range limited to $[0, 1]$ for left case).
Synthetic benchmark for temperature plumes

Left: Mesh (free slip boundary conditions for velocity)
Middle: uncorrected FEM/FV coupling
Right: corrected FEM/FV coupling

Observation: Mass conservation is extremely important in multi-physics
The effect of iterative solvers: pure transport

Remark: All our arguments are based on the assumption that the system is solved exactly which is never the case.

Iterative solver: MINRES + block-preconditioner using ML-AMG

<table>
<thead>
<tr>
<th>Stokes residual</th>
<th>without correction $|c - c_h|_0$</th>
<th>with correction $|c - c_h|_0$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.383655e+01</td>
<td>1.292926e-01</td>
<td>1.178171e-01</td>
</tr>
<tr>
<td>6.188999e-01</td>
<td>2.573021e-02</td>
<td>8.332195e-04</td>
</tr>
<tr>
<td>1.472225e-01</td>
<td>2.564630e-02</td>
<td>1.953160e-04</td>
</tr>
<tr>
<td>1.312045e-05</td>
<td>2.570321e-02</td>
<td>1.479205e-07</td>
</tr>
<tr>
<td>3.045244e-09</td>
<td>2.570320e-02</td>
<td>8.175043e-11</td>
</tr>
<tr>
<td>1.340526e-11</td>
<td>2.570320e-02</td>
<td>5.153043e-13</td>
</tr>
</tbody>
</table>

Case no correction: The consistency error for the transport depends on the discretization error of the Stokes

Case mass correction: The consistency error for the transport depends on the iteration error of the Stokes
No fault recovery strategy within a MG

From almost on the top back to the checkpoint level
Why do we need fault tolerant algorithms?

- With more than $10^6$ compute cores hardware failure can happen easily
- Hardware-based strategies are too costly (redundancy)
- Software-based strategies are too slow (checkpointing)
- Algorithmic-based strategies are in its infancy [Cui et al 13, Xu 13, Shantharam et al 14]
Optimal fault recovery strategy within a MG

From almost on the top to the top without delay
Numerical results

Algorithmic strategy:
- Freeze the data on the adjoint lower primitives (Dirichlet IC)
- Replace the faulty processor by several ones (over balancing)
- Control the catch-up progress (hierarchical residual representation)
Basics: Standard Multilevel Monte Carlo

- **Model problem:** \( \nabla \cdot (k(x, \omega) \nabla p(x, \omega)) = f(x, \omega), \ \omega \in \Omega \)

- **Sampling** from \( k(x, \omega) \) by e.g.:
  - Truncated Karhunen-Loeve (KL) expansion
  - Circulant embedding (FFT)
  - PDE-based variants (e.g. [Lindgren+Rue+Lindström 11])

- **Standard Monte Carlo** estimator: \( (Q \) is the quantity of interest) \[
\hat{Q}^{MC}_{N;h} := \frac{1}{N} \sum_{i=1}^{N} Q^{(i)}_h, \quad \text{MSE} = \frac{\text{Var}[Q_h]}{N} + (\mathbb{E}[Q_h - Q])^2
\]

- **Standard Multilevel Monte Carlo** estimator: \( h := h_L, \ h_l := 1/2 h_{l-1}, \ Q_{h_{-1}} := 0 \) \[
\text{MSE} = \sum_{l=0}^{L} \frac{\text{Var}[Q_{h_l} - Q_{h_{l-1}}]}{N_l} + (\mathbb{E}[Q_h - Q])^2, \quad [\text{Cliffe et al. 11}]\]
Coarse grid variations (randomly layered domain)

- Largest jumps resolved by the coarsest mesh
- Jumps by up to 5 orders of magnitude from layer to layer
- Asymptotic MG convergence rate of 0.2 for a $V(3,3)$-cycle
- Fully randomly generated samples for UQ
Basic Multilevel Monte Carlo Algorithm

**Sequential adaptive algorithm:** [Cliffe et al. 11]

Step 1 Start with the coarsest level $L = 0$, set $Q_{-1} :=$

Step 2 Estimate $\nabla [Q_L - Q_{L-1}]$ by a small number of samples

Step 3 Compute the estimated optimal $N_l$, $l = 0, ..., L$

Step 4 Evaluate extra samples at each level

Step 5 Check if the bias is small enough: if not go back to Step 2 and increase $L$

**Open question:** How to parallelize it?

- **We know** the cost for one sample on each level
- **We know** the parallel efficiency of the solver
- **We do not know** the number of required samples on each level
- **We do not know** the number of maximal levels

We need smart **scheduling** strategies.
How to balance and schedule?

solver

versus samples
Scheduling approaches for MLMC

**Fully sequential sampling:**
- Simple job *script* processing possible
- Requires extreme *strong* scaling of the solver (coarse level)

**Homogeneous bulk synchronous:**
- Parallelization over solver and samples
- Might lead to *over-sampling* especially for adaptive MLMC
Heterogeneous bulk synchronous

Objectives: minimize the required time and optimize the synchronization points

Flexibility: strong scaling of the forward problem for certain runs

Algorithm: genetic multiobjective optimization (e.g. SPEA-II or NSGA-II)

Aim: Design of a C layer/wrapper, which is minimally intrusive.

Possibly most intrusive changes for the solver:
- Allow to use a splitted MPI communicator for the forward simulation for all MPI calls
- No memory leaks at the end of a simulation run

Outcome:
- Parallelization over the samples for the coarser levels
- Parallelization within the solver for the finer levels
Parallel efficiency: strong scaling

Number of processes $P_{max}$

Time [seconds]
Simulated annealing

Additional constraints allow **more flexibility** for mutation

**More likely** that the number of parallel samples will be adjusted properly
Homogeneous versus heterogeneous scheduling

Heterogeneous scheduling is more robust and often more efficient.
Variational inequalities in applications

Option pricing
Variational inequalities in applications

Option pricing

Contact problems
Variational inequalities in applications

Option pricing

Contact problems

Plasticity
Variational inequalities in applications

Option pricing

Contact problems

Plasticity

Phase transition
Obstacle model problem

- Constrained minimization problem:
  Find $u \in K := \{v \in H^1_0(0,1); v \geq \psi\}$ such that
  
  $$J(u) = \min_{v \in K} J(v)$$

  with $J(v) := \frac{1}{2} \int_0^1 (v')^2 ds - \int_0^1 fv ds$

- How to discretize and solve in the more general case?

\[ f = \text{const} \]
\[ \psi = -1 \]
\[ a = \sqrt{-2/f}, f \leq -8 \]
From constrained to unconstrained

**Unconstrained minimization:** Find $u_\epsilon \in V$ such that

$$ J_\epsilon(u_\epsilon) = \inf_{v \in V} J_\epsilon(v), \quad J_\epsilon(v) := \frac{1}{2}a(v, v) - f(v) + \chi_\epsilon(v) $$

with the penalty parameter $\epsilon$ and $\chi_\epsilon(v) = 0$ iff $v \in K$, proper l.s.c., non-negative

**Unconstrained penalty** versus **constrained hybrid formulation**

**Hybrid formulation:** Find $(u, \lambda) \in (V, M)$ ($M$ convex cone) such that

$$ a(u, v) + b(v, \lambda) = f(v), \quad v \in V, $$

$$ b(u, \mu - \lambda) \leq g(\mu - \lambda), \quad \mu \in M $$
Variational inequalities

- **Obstacle model problem:** Find $u$ such that
  \[- \text{div} (\nabla u) + \lambda = f, \quad \text{and} \quad u \leq g \quad \text{in} \ \Omega, \quad u = 0 \quad \text{on} \ \partial \Omega\]

  \[\text{PDE + algebraic constraints on domain}\]

- **Signorini model problem:** Find $u$ such that
  \[- \text{div} (\nabla u) = f, \quad \text{and} \quad \lambda = -\frac{\partial u}{\partial n}, \quad u \leq g \quad \text{on} \ \Gamma \subset \partial \Omega, \quad u = 0 \quad \text{on} \ \partial \Omega \setminus \Gamma\]

  \[\text{PDE + algebraic constraints on interface/boundary}\]

**Weak discrete formulation has a non-linear saddle-point structure**

\[
A(\underbrace{u_h}_{\text{primal}}) + B(\underbrace{\lambda_h}_{\text{dual}}) = f, \\
C(u_h, \lambda_h) = 0
\]

NCP-function

e.g., Fischer-Burmeister, radial return mapping [Facchinei/Pang 03, Hintermüller et al 03]
Variational inequalities in porous media

- **Signorini type**: PDE + algebraic constraints on interfaces
  
  **Application**: Brooks–Corey type capillary pressure for heterogeneous media
  
  \[
  p_n - p_w = p_c = p_c^{BC}(S_w) - \tau \frac{dS_w}{dt}
  \]
  
  [Brooks, Corey 71]
  [Hassanizadeh, Gray 90]

- **Obstacle type**: PDE + algebraic constraints on domain
  
  **Application**: phase transition in multi-phase multi-component systems
  
  \[
  \sum_{\kappa=1}^{N} x^\kappa_\alpha \leq 1, \quad \sum_{\kappa=1}^{N} x^\kappa_\alpha = 1 \text{ if phase } \alpha \text{ is present.}
  \]

How to discretize and how to solve?
From bi-directional coupled...

... via fractional flow ...

to one-directional coupled equalities
Extension: Heterogeneous media

- **Van-Genuchten [80]:** Regularization at $S_w = 1$ of Brooks–Corey [64]

  \[ p_e = 0: \text{v. Genuchten} \]

  High permeability - low permeability

  \[ p_e > 0: \text{Brooks–Corey} \]

  \[ p_e \geq p_c \geq 0, \ S_n^l \geq 0, \ [p_c]S_n^l = 0 \]

  Inequality constraint

  Regularized model numerically easier but introduces artificial diffusion

- **Heterogeneous media** possibly results in jumps in \([p_e]\) [DeNeef 00]
Two domains - Influence of $\tau$ and $[p_e]$ cuts at $t = 0.2, \ldots, 1.0$

Brooks–Corey model: $k_{rw} = s^{3+\frac{2}{\theta}}$, $k_{rn} = (1 - s)^2(1 - s^{1+\frac{2}{\theta}})$, $p_{c}^{stat} = p_es^{-\frac{1}{\theta}}$, $\theta = 2$

$\Rightarrow$ Results in sharper wave fronts
Influence of $\tau$ at different time-points $t_i, i = 1, 2, 3$

$t_1 = 0$: $\Rightarrow$ more diffusive

$t_2 = 60$: $\Rightarrow$ different wave fronts

$t_3 = 200$: $\Rightarrow$ no difference in the long-range
Semi-smooth Newton for two-phase flow problems

Step 3: Inequality constraint for Brooks–Corey at interface:

\[
[p_c] \geq 0, \quad S_n^l \geq 0, \quad [p_c]S_n^l = 0
\]

NCP-function:

\[
C([p_c], S_n^l) := [p_c] - \max(0, [p_c] - cS_n^l), \quad c > 0
\]

Active set strategy:

Active node \( p \in \mathcal{A} \): \( [p_c]_p = 0 \), Dirichlet condition in pressure
Inactive node \( p \in \mathcal{I} \): \( s_p = 1 \), Dirichlet condition in saturation, \( (s = 1 - S_n^l) \)

Iteration to find correct active set: Start with \( \mathcal{A}_0 = \mathcal{A}^{\text{old}}, \quad \mathcal{I}_0 = \mathcal{I}^{\text{old}} \)

If \( p \in \mathcal{A} \) and \( s_p > 1 \) \( \rightarrow \) set \( p \) inactive in next iteration step
If \( p \in \mathcal{I} \) and \( [p_c]_p < 0 \) \( \rightarrow \) set \( p \) active in next iteration step

Similar approach to include condition \( [p_e] \geq [p_c] \) in model and discretization
Domain with lenses - active sets

Setting:

- $K = 1$
- $K = 0.25$
- $K = 0.04$

Graphs showing the evolution of active nodes over time for different $K$ values.
From multi-phase multi-component...

... constitutive relations ...

to coupled equalities
Multi-phase multi-component system (MpNc)

- Considering a system with \( N \) different components and with \( M \) different phases, we have \( MN + 2M + N + 1 \) unknowns.

| \( T \) | temperature |
| \( p_\alpha \) | pressure in phase \( \alpha \in \{1, \ldots, M\} \) |
| \( S_\alpha \) | saturation of phase \( \alpha \in \{1, \ldots, M\} \) |
| \( f^\kappa \) | fugacity of comp. \( \kappa \in \{1, \ldots, N\} \) |
| \( x_\alpha^\kappa \) | mole fraction of comp. \( \kappa \in \{1, \ldots, N\} \) in phase \( \alpha \in \{1, \ldots, M\} \) |

- Non-linear constitutive relations

| \( \mu_\alpha(T) \) | viscosity of phase \( \alpha \) |
| \( \rho_{\text{mol},\alpha}(p_\alpha, x_\alpha^\kappa, T) \) | molar density of phase \( \alpha \) |
| \( \rho_{\text{mass},\alpha}(p_\alpha, x_\alpha^\kappa, T) \) | mass density of phase \( \alpha \) |
| \( u_\alpha(p_\alpha, x_\alpha^\kappa, T) \) | specific internal energy of phase \( \alpha \) |
| \( h_\alpha(p_\alpha, x_\alpha^\kappa, T) \) | specific enthalpy of phase \( \alpha \) |
| \( k_{r\alpha}(S_\alpha) \) | relative permeability of phase \( \alpha \) |
| \( \lambda_{pm}(p_\alpha, x_\alpha^\kappa, S_\alpha, T) \) | heat conduction coefficient |

- \( N + 1 \) balance equations of the form \( \frac{\partial \xi}{\partial t} + \nabla \cdot \Psi - q = 0 \)
Balance equations

• \( N \) mass conservation of the components \( \kappa \in \{1, \ldots, N\} \)

\[
\xi = \phi \sum_{\alpha=1}^{M} \rho_{\text{mol,}\alpha} x_{\alpha}^{\kappa} S_{\alpha}, \quad \rho_{\text{mass,}\alpha} = \rho_{\text{mol,}\alpha} \sum_{\kappa=1}^{N} x_{\alpha}^{\kappa} M^{\kappa}.
\]

\[
\Psi = - \sum_{\alpha=1}^{M} \left( \frac{k_{\alpha}}{\mu_{\alpha}} \rho_{\text{mol,}\alpha} x_{\alpha}^{\kappa} K (\nabla p_{\alpha} - \rho_{\text{mass,\alpha}} g) + D_{\text{pm},\alpha} \rho_{\text{mol,}\alpha} \nabla x_{\alpha}^{\kappa} \right),
\]

• One energy conservation

\[
\xi = \phi \sum_{\alpha=1}^{M} \rho_{\text{mass,}\alpha} u_{\alpha} S_{\alpha} + (1 - \phi) \rho_{s} c_{s} T,
\]

\[
\Psi = - \sum_{\alpha=1}^{M} \left( \frac{k_{\alpha}}{\mu_{\alpha}} \rho_{\text{mass,}\alpha} h_{\alpha} K (\nabla p_{\alpha} - \rho_{\text{mass,\alpha}} g) \right)
\]

\[
- \sum_{\kappa=1}^{N} \sum_{\alpha=1}^{M} \left( D_{\text{pm},\alpha}^{\kappa} \rho_{\text{mol,}\alpha} h_{\alpha}^{\kappa} M^{\kappa} \nabla x_{\alpha}^{\kappa} \right) - \lambda_{\text{pm}} \nabla T
\]
Consitutive relations

- **One** saturation scaling:

\[
\sum_{\alpha=1}^{M} S_{\alpha} = 1
\]

- **\(M-1\)** capillary pressure relations: \(\alpha \in \{2, \ldots, M\}\)

\[
p_{\alpha-1} - p_{\alpha} = p_{c,(\alpha-1)\alpha}(S_{\alpha})
\]

- **\(MN\)** fugacity relations: \(\kappa \in \{1, \ldots, N\}\), \(\alpha \in \{1, \ldots, M\}\)

\[
f_{\kappa} = \Phi_{\alpha}^{\kappa}(p_{\alpha}, x_{\alpha}^{1}, \ldots, x_{\alpha}^{N}, T)x_{\alpha}^{\kappa}p_{\alpha}, \quad x_{\alpha}^{\kappa} = x_{\alpha}^{\kappa}(p_{1}, f^{1}, \ldots, f^{N}, T)
\]

\[\implies\] \(N + 1\) PDES and \(1 + (M - 1) + MN\) constitutive relations

\[\implies\] \(MN + 2M + N + 1 - (N + 1 + M + NM) = M\) missing conditions
From multi-phase multi-component...

... constitutive relations ...

to coupled equalities
Phase transition

- **Sum** of mole fractions per phase is **bounded**:
  \[ \sum_{\kappa=1}^{N} x_{\kappa}^{\alpha} \leq 1, \quad \sum_{\kappa=1}^{N} x_{\kappa}^{\alpha} = 1 \text{ if phase } \alpha \text{ is present.} \]

- **Saturation** \( S_{\alpha} \) for face \( \alpha \) is **non-negative**:
  \[ 1 - \sum_{\kappa=1}^{N} x_{\kappa}^{\alpha} \geq 0, \quad S_{\alpha} \geq 0, \quad S_{\alpha} \left( 1 - \sum_{\kappa=1}^{N} x_{\kappa}^{\alpha} \right) = 0. \]

- **Elimination** of the \( MN \) mole fractions, \( S_1 \) and \( p_\alpha, \alpha \in \{2, \ldots, M\} \) results in a complex system with \( N + 1 + M \) unknowns
  \[ \left( p_1, f^1, \ldots, f^N, S_2, \ldots, S_M, T \right) \]

\( N + 1 \) **PDEs** plus \( M \) **Karush–Kuhn–Tucker triplets**
Semi-smooth Newton for two-phase flow problems

Inequality constraint for phase transition in domain:

\[ S_\alpha \geq 0, \quad 1 - \sum_{\kappa=1}^{N} x_{\kappa \alpha} \geq 0, \quad (1 - \sum_{\kappa=1}^{N} x_{\kappa \alpha}) S_\alpha = 0, \quad \alpha = 1, \ldots, M, \]

where \( S_\alpha \) phase saturation, \( x_{\kappa \alpha} \) mole fraction of component \( \kappa \)

**NCP-functions:** \( C_\alpha := S_\alpha - \max(0, S_\alpha - c_\alpha (1 - \sum_{\kappa=1}^{N} x_{\kappa \alpha})), \quad c_\alpha > 0 \)

**But:** \( S_\alpha, x_{\kappa \alpha} \) are not the primary variables

**Primary variables:** \( p_1, T, S_2, \ldots S_M, x_1^1, \ldots, x_1^N \)

Rewrite: \( S_1 = 1 - \sum_{\alpha=2}^{M} S_\alpha, \quad x_{\kappa \alpha} = \gamma_{\kappa \alpha}^\alpha(T, p_1) x_1^\kappa, \quad \alpha = 2, \ldots M, \quad \kappa = 1, \ldots N, \)

where the function \( \gamma_{\kappa \alpha}^\alpha \) depends on Dalton’s, Henry’s, Raoult’s law

\[ \Longrightarrow \] New NCP functions in terms of primary variables

\[ \Longrightarrow \] Semismooth Newton can be applied to the coupled non-linear system
Semismooth Newton for a two phase system

- Setting and Newton iteration

- Saturation and active sets
Time step-size control

- Heuristic strategy to balance number of Newton and time steps

- Heuristic strategy to maximize time step while Newton converges
MPFA discretisation

**Idea:** Use a multi-point flux approximation

**Observation:** L-methods have small stencils [Aavatsmark et al 02–08]

**New** selection criteria:
Increases the stencil in streamline direction
Still locally conservative
More accurate for heterogeneous permeabilities

**Hanging** nodes case

**Adaptive** mesh in time for DNAPL infiltration
3 phases 7 component system

- **3D oil extraction problem**
  5th benchmark of the Society of Petroleum Engineers [Killough, Kossack 87]

- production well is located in one lower corner

- water alternating gas is placed in the opposite corner

- \( N = 7 \) components: \( \text{H}_2\text{O}, \text{C}_1, \text{C}_3, \text{C}_6, \text{C}_{10}, \text{C}_{15}, \text{C}_{20} \)

- \( M = 3 \) phases: gas, oil, water
Ketzin: on-shore CO\textsubscript{2} injection project

Numerical comparison between an **immiscible** and **miscible** approach
Input data taken from [Martens et al 12], Simulation by Andreas Lauser (IWS Stuttgart)

• Gas phase saturation for the **miscible** model

• Gas phase saturation for the **immiscible** model
Conclusion

- Abstract framework for **saddle point problems** for
  - constructing fast Stokes solvers
  - analyzing variational inequalities

- Abstract framework of **dual and hierarchical hybrid meshes** for
  - reduced communication
  - strong mass conservation

- **Challenges** in high performance computation
  - Fault tolerant solvers
  - Scheduling in uncertainty quantification