Space-time domain decomposition for a nonlinear parabolic equation with discontinuous capillary pressure

Elyes Ahmed, Caroline Japhet, Michel Kern

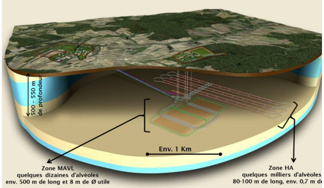
INRIA Paris — Maison de la Simulation – ENPC – University Paris 13
Work supported by Andra & ANR Dedales

SIAM Conference on Mathematical and Computational Issues in the Geosciences
September 11—14, 2017 Erlangen, Germany

Outline

Geological disposal of nuclear waste

Deep repository
(Long lived & High-level radioactive waste)

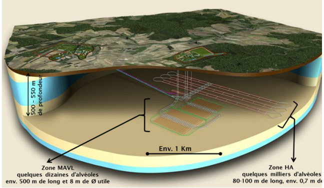


- Different materials $\to$ strong heterogeneity, different time scales.
- Large differences in spatial scales.
- Long-term computations.

Geological disposal of nuclear waste

Deep repository

(Long lived & High-level radioactive waste)

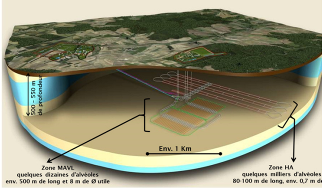


- Different materials $\rightarrow$ strong heterogeneity, different time scales.
- Large differences in spatial scales.
- Long-term computations.
- Use space-time (global in time) domain decomposition methods

Geological disposal of nuclear waste

Deep repository

(Long lived & High-level radioactive waste)



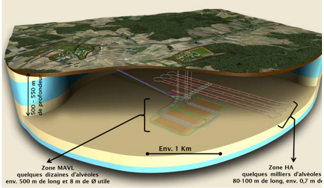
- Different materials $\rightarrow$ strong heterogeneity, different time scales.
- Large differences in spatial scales.
- Long-term computations.

✔ Use space-time (global in time) domain decomposition methods

☞ **Take into account different capillary pressure curves**

Geological disposal of nuclear waste

Deep repository
(Long lived & High-level radioactive waste)



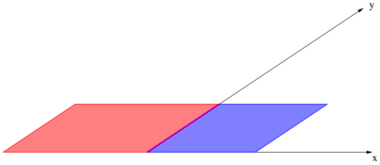
- Different materials → strong heterogeneity, different time scales.
- Large differences in spatial scales.
- Long-term computations.
- ✓ Use space-time (global in time) domain decomposition methods

☞ **Take into account different capillary pressure curves**

☞ **Extend optimized Schwarz method to nonlinear diffusion**

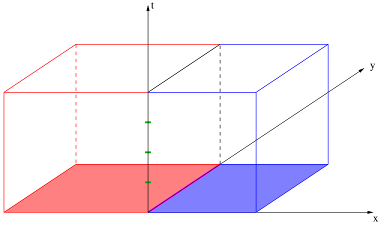
Space–time domain decomposition

Domain decomposition in space



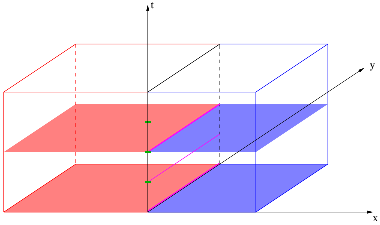
Space–time domain decomposition

Domain decomposition in space



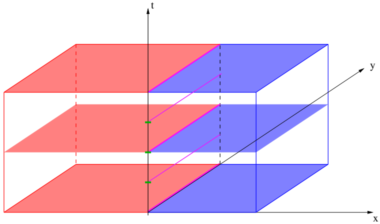
Space–time domain decomposition

Domain decomposition in space



Space–time domain decomposition

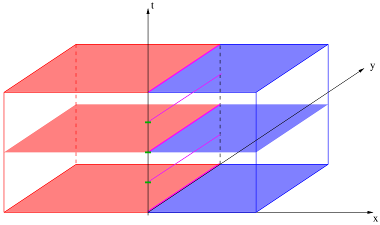
Domain decomposition in space



- Discretize in time and apply DD algorithm at each time step:
  - Solve stationary problems in the subdomains
  - Exchange information through the interface
- Use the same time step on the whole domain.

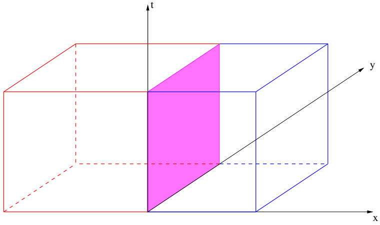
Space–time domain decomposition

Domain decomposition in space



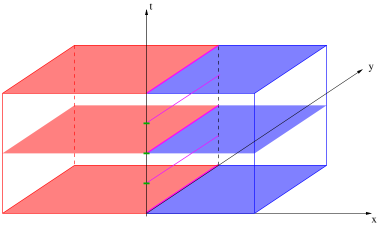
- Discretize in time and apply DD algorithm at each time step:
  - ▶ Solve stationary problems in the subdomains
  - ▶ Exchange information through the interface
- Use the same time step on the whole domain.

Space-time domain decomposition



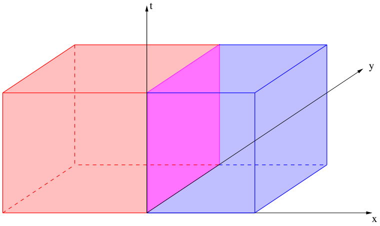
Space–time domain decomposition

Domain decomposition in space



- Discretize in time and apply DD algorithm at each time step:
  - ▶ Solve stationary problems in the subdomains
  - ▶ Exchange information through the interface
- Use the same time step on the whole domain.

Space-time domain decomposition



- Solve time-dependent problems in the subdomains
- Exchange information through the space-time interface
- Enable local discretizations both in space and in time

⟶ local time stepping

Model problem: nonlinear (degenerate) diffusion equation

Two–phase immiscible flow, global pressure + Kirchhoff transformation, neglect advection (Enchery et al. (06), Cances (08))

S: water saturation. $\pi(S)$ capillary pressure, increasing function on $[0,1]$ (extend continuously to $\mathbf{R}$). $\lambda(S)$ mobility, ω porosity

$$\phi(S) = \int_0^S K\lambda(u)\pi'(u)\,du.$$

Model problem: nonlinear (degenerate) diffusion equation

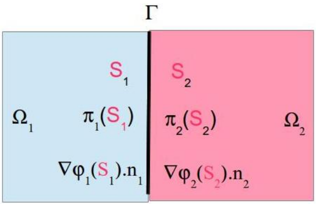
Two–phase immiscible flow, global pressure + Kirchhoff transformation, neglect advection (Enchery et al. (06), Cances (08))
S: water saturation. $\pi(S)$ capillary pressure, increasing function on $[0, 1]$ (extend continuously to **R**). $\lambda(S)$ mobility, ω porosity

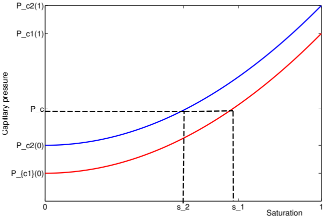
$$\phi(S) = \int_0^S K\lambda(u)\pi'(u)\,du.$$

Simplified equation

$$\omega\partial_t S - \Delta\phi(S) = 0 \quad \text{in } \Omega\times[0, T]$$

Two subdomains (rock types)





Multi-domain formulation

2 subdomains Ω_1, Ω_2, interface $\Gamma = \overline{\Omega_1} \cap \overline{\Omega_2}$. Solve (for $i = 1, 2$):

$$
\begin{aligned}
\omega \partial_t S_i - \Delta \phi_i(S_) &= 0 \quad \text{in } \Omega_i \times [0, T], \\
\frac{\partial \phi_i(S_i)}{\partial n_i} &= 0 \quad \text{on } (\partial \Omega_i \backslash \Gamma) \times (0, T) \\
S_i(., 0) = S_0(.)_{|\Omega_i} & \quad \text{in } \Omega_i
\end{aligned}
$$

Multi-domain formulation

2 subdomains Ω_1, Ω_2, interface $\Gamma = \overline{\Omega_1} \cap \overline{\Omega_2}$. Solve (for $i = 1, 2$):

$$\begin{aligned} \omega \partial_t S_i - \Delta \phi_i(S_)) &= 0 \quad \text{in } \Omega_i \times [0, T], \\ \frac{\partial \phi_i(S_i)}{\partial n_i} &= 0 \quad \text{on } (\partial \Omega_i \backslash \Gamma) \times (0, T) \\ S_i(.,0) &= S_0(.)_{|\Omega_i} \quad \text{in } \Omega_i \end{aligned}$$

together with *natural* transmission conditions on the space-time interface

Continuity of capillary pressure $\pi_1(S_1) = \pi_2(S_2) \quad \text{on } \Gamma \times (0, T)$

Continuity of the flux $\nabla \phi_1(S_1).n_1 + \nabla \phi_2(S_2).n_2 = 0 \quad \text{on } \Gamma \times (0, T)$

Multi-domain formulation

2 subdomains Ω_1, Ω_2, interface $\Gamma = \overline{\Omega_1} \cap \overline{\Omega_2}$. Solve (for $i = 1, 2$):

$$\begin{aligned} \omega \partial_t S_i - \Delta \phi_i(S_)) &= 0 \quad \text{in } \Omega_i \times [0, T], \\ \frac{\partial \phi_i(S_i)}{\partial n_i} &= 0 \quad \text{on } (\partial \Omega_i \backslash \Gamma) \times (0, T) \\ S_i(., 0) = S_0(.)_{|\Omega_i} & \quad \text{in } \Omega_i \end{aligned}$$

together with *natural* transmission conditions on the space-time interface

$$\begin{aligned} \text{Continuity of capillary pressure} \quad & \pi_1(S_1) = \pi_2(S_2) \quad \text{on } \Gamma \times (0, T) \\ \text{Continuity of the flux} \quad & \nabla \phi_1(S_1).n_1 + \nabla \phi_2(S_2).n_2 = 0 \quad \text{on } \Gamma \times (0, T) \end{aligned}$$

Replace by equivalent Robin transmission conditions

$$\begin{aligned} \nabla \phi_1(S_1).n_1 + \beta_1 \pi_1(S_1) &= -\nabla \phi_2(S_2).n_2 + \beta_1 \pi_2(S_2) \quad \text{on } \Gamma \times (0, T) \\ \nabla \phi_2(S_2).n_2 + \beta_2 \pi_2(S_2) &= -\nabla \phi_1(S_1).n_1 + \beta_2 \pi_1(S_1) \quad \text{on } \Gamma \times (0, T) \end{aligned}$$

$\beta_1, \beta_2 > 0$ given parameters

Global in time domain decomposition algorithm

Non-linear Optimized Schwarz waveform relaxation algorithm

Given S_i^0, iterate for $k = 0, \dots$
Solve for S_i^{k+1}, $i = 1, 2, j = 3 - i$

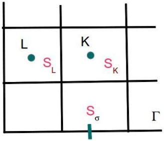
$$
\begin{aligned}
&\omega \partial_t S_i^{k+1} - \Delta \phi_i(S_i^{k+1}) = 0 && \text{in } \Omega_i \times [0, T] \\
&\frac{\partial \phi_i(S_i^{k+1})}{\partial n_i} = 0 && \text{on } (\partial \Omega_i \backslash \Gamma) \times (0, T) \\
&S_i^{k+1}(.,0) = S_0(.)_{|\Omega_i} && \text{in } \Omega_i \\
&\nabla \phi_i(S_i^{k+1}).n_i + \beta_i \pi_i(S_i^{k+1}) = -\nabla \phi_j(S_j^k).n_j + \beta_i \pi_j(S_j^k) && \text{on } \Gamma \times [0, T],
\end{aligned}
$$

β_1, β_2 can be chosen to optimize convergence rate (Bennequin-Gander-Halpern (09),Hoang-Jaffré,Japhet,Kern,Roberts (13))

Basic ingredient: subdomain solver with Robin bc. Existence: adapt proof from Enchery et al. (06), Cances (08), via convergence of finite volume scheme.

Cell centered finite volume scheme (Enchery et al, 06)

Triangulation $\mathscr{T}$, cells $K \in \mathscr{T}$, boundary faces $\sigma \subset \Gamma$.
Unknowns : cell values $(S_K)_{K\in\mathscr{T}}$, boundary face values $(S_\sigma)_{\sigma\in\mathscr{E}_\Gamma}$
Notations: $K|L$ = edge between K and L, $\tau_{K|L}$: transmissivity



Interior equation

$$m(K)\frac{S_K^{n+1}-S_K^n}{\delta t}+\sum_{L\in\mathscr{N}(K)}\tau_{K|L}\left(\phi(S_K^{n+1})-\phi(S_L^{n+1})\right)$$
$$+\sum_{\sigma\in\mathscr{E}_\Gamma\cap\mathscr{E}_K}\tau_{K,\sigma}\left(\phi(S_K^{n+1})-\phi(S_\sigma^{n+1})\right)=0, \quad K\in\mathscr{T}.$$

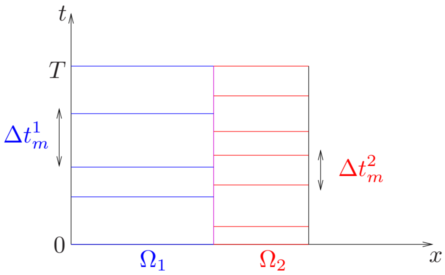
Robin BC for boundary faces

$$-\tau_{K,\sigma}\left(\phi(S_K^{n+1})-\phi(S_\sigma^{n+1})\right)+\beta m(\sigma)\pi(S_\sigma^{n+1})=g_\sigma, \quad \sigma\in\mathscr{E}_\Gamma$$

Implemented with Matlab Reservoir Simulation Toolbox (K. A. Lie et al. (14))
Solver with automatic differentiation : no explicit programming of Jacobian

Nonconforming discretization in time

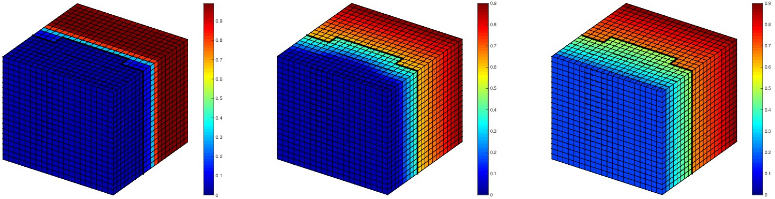
Use different time steps in the subdomains



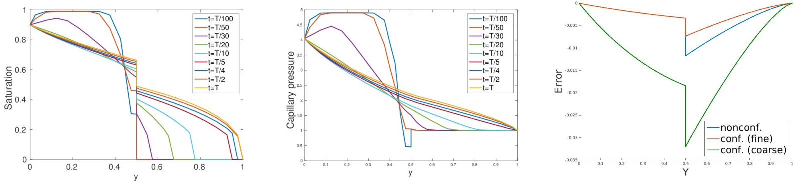
Information on one time grid at the interface is passed to the other time grid at the interface using L2-projections
$\rightarrow$ use an optimal projection algorithm, Gander-Japhet-Maday-Nataf (2005)

Validation example, 2 rock types

Homogenenous medium $\Omega = (0,1)^3$. Mobility $\lambda_0(S) = S$, $S \in [0,1]$,
Capillary pressure $\pi_1(S) = 5S^2$, $\quad \pi_2(S) = 5S^2 + 1 \quad S \in [0,1]$



Evolution of the saturation ($t = 0.019, t = 0.6, t = 3$)

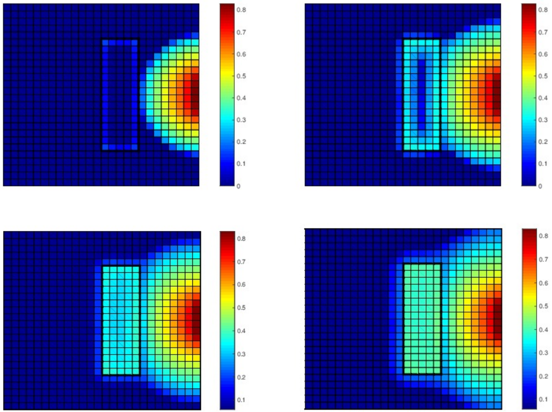


evolution of the saturation, capillary pressure, and error at final time along a line orthogonal to the interface.

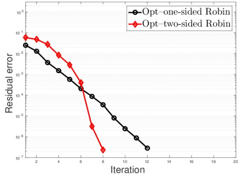
DNAPL infiltration: medium with a low capillarity lens

Mobilities $\lambda_{o,i}(S) = S^2$, and $\lambda_{g,i}(S) = 3(1-S)^2,\ i \in \{1,2\}$,

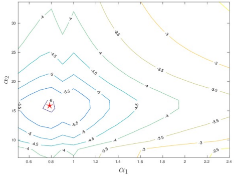
Capilary pressure $\pi_1(S) = \ln(1-S)$, and $\pi_2(S) = 0.5 - \ln(1-S)$.



Evolution of saturation ($t = 100, 200, 350, 480$)



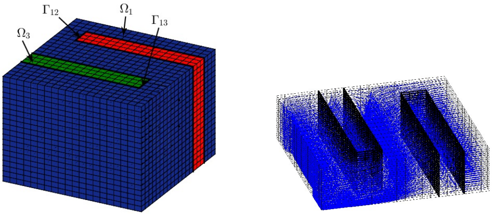
Convergence curve



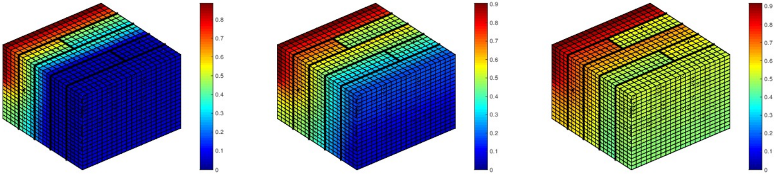
Influence of parameters β_1, β_2 on convergence

Example wth 3 rock types

Capillary pressure curves : $\pi_1(u) = 3u^2, \quad \pi_2(u) = 5u^2$, and $\quad \pi_3(u) = 3u^2 + 0.5$.
Use of time windows reduces number of DD iterations (after the first one)



Mesh and velocity streamlines



Evolution of the saturation ($t = 500, 2000, 4000$)