Hybrid dimensional Darcy flow in fractured porous medium and parallel implementation in code ComPASS

Feng Xing[1,2,3]

joint work with

Konstantin Brenner[1,2], **Simon Lopez**[3], **Roland Masson**[1,2]

(1) *Laboratoire J.A. Dieudonné, Université de Nice*
(2) *Team COFFEE, INRIA Sophia Antipolis*
(3) *Institut BRGM*

ComPASS : Computing Parallel Architecture to Speed up Simulations

Goal: develop a parallel prototype to test promising numerical methods on realistic cases

Brief history

0. CEMRACS 2012: heat equation as a toy problem.
   E. Dalissier, C. Guichard, P. Havé, R. Masson, C. Yang.
1. Two phase flow in porous media: work on the linear solvers (FVCA 7).
   R. Eymard, C. Guichard, R. Masson.
2. Tracer model on a fractured porous media.
   R. Masson, F. Xing

Goal: develop a parallel prototype to test promising numerical methods on realistic cases

Main specifications

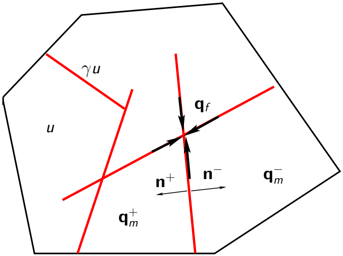
- Parallel programming with MPI using Fortran 2003
- General meshes (polyhedral cells, possibly non planar faces)

- Adapted to Finite Volume schemes with d.o.f. at nodes, cells, faces and 'usual' compact stencil
- Connected with scientific computing libraries: METIS, PETSc, Trilinos, VTK

- Tracer problem on discrete fracture network
- Vertex-centered discretisation (VAG)
- Implementation in code ComPASS
- Numerical results

Hybrid dimensional models for DFN [Alboin-Jaffré-Roberts-Serres 2002]
$d_f \ll \mathrm{diam}(\Omega)$ continuous pressure u on $\overline{\Omega}$ dim(Γ)=dim(Ω)-1

1. Pression equation



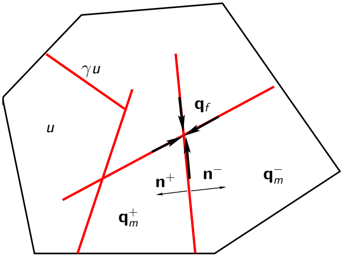
$$
\left\{
\begin{aligned}
\mathrm{div}(\mathbf{q}_m) &= 0 && \text{on } \Omega \setminus \overline{\Gamma} \\
\mathrm{div}_\tau(\mathbf{q}_f) + [\![\mathbf{q}_m \cdot \mathbf{n}]\!] &= 0 && \text{on } \Gamma \\
\mathbf{q}_m &= -\Lambda_m \nabla u && \text{on } \Omega \setminus \overline{\Gamma} \\
\mathbf{q}_f &= -d_f \Lambda_f \nabla_\tau \gamma u && \text{on } \Gamma
\end{aligned}
\right.
$$

with the jump $[\![\mathbf{q}_m \cdot \mathbf{n}]\!] = \mathbf{q}_m^+ \cdot \mathbf{n}^+ + \mathbf{q}_m^- \cdot \mathbf{n}^-$

Tracer problem on a Discrete Fracture Network (DFN)

Hybrid dimensional models for DFN [Alboin-Jaffré-Roberts-Serres 2002]
$d_f \ll \mathrm{diam}(\Omega)$ continuous pressure u on $\overline{\Omega}$ dim(Γ)=dim(Ω)-1

1. Pression equation



$$
\left\{
\begin{aligned}
\mathrm{div}(\mathbf{q}_m) &= 0 && \text{on } \Omega \setminus \overline{\Gamma} \\
\mathrm{div}_\tau(\mathbf{q}_f) + [\![\mathbf{q}_m \cdot \mathbf{n}]\!] &= 0 && \text{on } \Gamma \\
\mathbf{q}_m &= -\Lambda_m \nabla u && \text{on } \Omega \setminus \overline{\Gamma} \\
\mathbf{q}_f &= -d_f \Lambda_f \nabla_\tau \gamma u && \text{on } \Gamma
\end{aligned}
\right.
$$

with the jump $[\![\mathbf{q}_m \cdot \mathbf{n}]\!] = \mathbf{q}_m^+ \cdot \mathbf{n}^+ + \mathbf{q}_m^- \cdot \mathbf{n}^-$

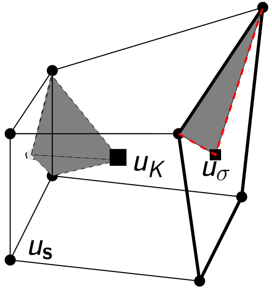
2. Tracer equation

$$
\left\{
\begin{aligned}
\partial_t c_m + \mathrm{div}(c_m \mathbf{q}_m) &= 0 && \text{on } \Omega \setminus \overline{\Gamma} \\
\partial_t (d_f c_f) + \mathrm{div}_\tau(c_f \mathbf{q}_f) + [\![c_m \mathbf{q}_m \cdot \mathbf{n}]\!] &= 0 && \text{on } \Gamma \\
c_m^+ &= c_f && \text{on } \{x \in \Gamma \,|\, \mathbf{q}_m^+ \cdot \mathbf{n}^+ > 0\} \\
c_m^- &= c_f && \text{on } \{x \in \Gamma \,|\, \mathbf{q}_m^- \cdot \mathbf{n}^- > 0\}
\end{aligned}
\right.
$$

⋆ **Discrete Unknowns** ⋆

$$u_{\mathcal{D}} = (u_K, u_\sigma, u_{\mathbf{s}}, K \in \mathcal{M}, \sigma \in \mathcal{F}_\Gamma, \mathbf{s} \in \mathcal{V})$$

$\mathcal{M}$: cells, $\mathcal{F}_\Gamma$: fracture faces, $\mathcal{V}$: vertex



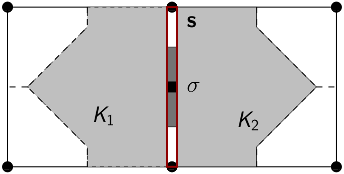
⋆ **Discrete Unknowns** ⋆

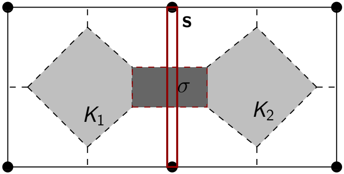
$$u_{\mathcal{D}} = (u_K, u_\sigma, u_{\mathbf{s}}, K \in \mathcal{M}, \sigma \in \mathcal{F}_\Gamma, \mathbf{s} \in \mathcal{V})$$

$\mathcal{M}$: cells, $\mathcal{F}_\Gamma$: fracture faces, $\mathcal{V}$: vertex

⋆ **Volume redistribution** ⋆

$$\omega_K,\ K \in \mathcal{M}, \quad \omega_\sigma,\ \sigma \in \mathcal{F}_\Gamma, \quad \omega_{\mathbf{s}},\ \mathbf{s} \in \mathcal{V}$$



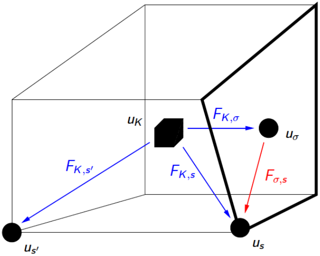


mixing of rocktype
⇒ non accurate results

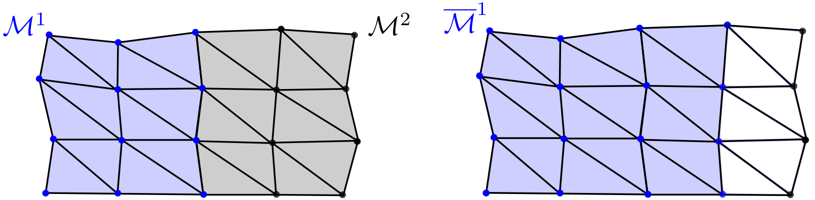
⋆ Fluxes ⋆

In the matrix : The fluxes $F_{K,\mathbf{s}}(u_{\mathcal{D}})$, $F_{K,\sigma}(u_{\mathcal{D}})$ are computed from all the nodes and the fracture faces connected to K.

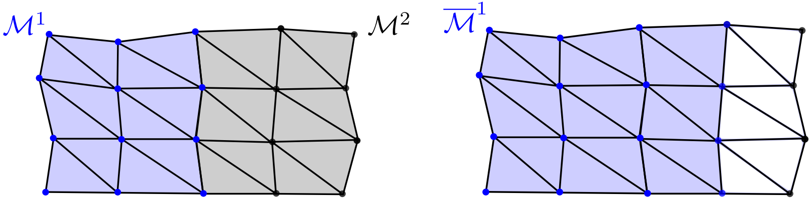
In the fracture : The fluxes $F_{\sigma,\mathbf{s}}(u_{\mathcal{D}})$ are computed from all the nodes connected to σ.



- Mesh decomposition by METIS: $\mathcal{M}^1$ and $\mathcal{M}^2$
- One layer ghost cells: $\overline{\mathcal{M}}^1$ and $\overline{\mathcal{M}}^2$

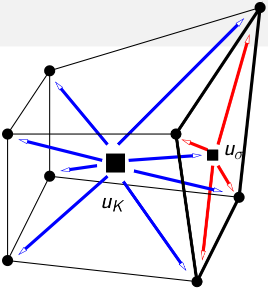


- Mesh decomposition by METIS: $\mathcal{M}^1$ and $\mathcal{M}^2$
- One layer ghost cells: $\overline{\mathcal{M}}^1$ and $\overline{\mathcal{M}}^2$



- Vertex: $\mathcal{V}^1$ and $\mathcal{V}^2$
- Vertex with ghost: $\overline{\mathcal{V}}^1$ and $\overline{\mathcal{V}}^2$
- Fracture faces: $\mathcal{F}^1$ and $\mathcal{F}^2$
- Fracture faces with ghost: $\overline{\mathcal{F}}^1$ and $\overline{\mathcal{F}}^2$

VAG discretization of the pressure equation



On each processor p, the unknowns are:

$$\overline{u}_{\mathcal{D}}^{p} = \left(u_K, u_\sigma, u_{\mathbf{s}}, K \in \overline{\mathcal{M}}^p, \sigma \in \overline{\mathcal{F}}_\Gamma^p, \mathbf{s} \in \overline{\mathcal{V}}^p\right)$$

On each processor p, the equations are:

$$\left\{\begin{array}{l}
\displaystyle\sum_{\nu \in \mathcal{V}_K \cup (\mathcal{F}_K \cap \mathcal{F}_\Gamma)} F_{K,\nu}(u_{\mathcal{D}}) = 0,\ K \in \overline{\mathcal{M}}^p \\[2ex]
\displaystyle\sum_{\mathbf{s} \in \mathcal{V}_\sigma} F_{\sigma,\mathbf{s}}(u_{\mathcal{D}}) + \sum_{K \in \mathcal{M}_\sigma} -F_{K,\sigma}(u_{\mathcal{D}}) = 0,\ \sigma \in \mathcal{F}_\Gamma^p \\[2ex]
\displaystyle\sum_{K \in \mathcal{M}_s} -F_{K,\mathbf{s}}(u_{\mathcal{D}}) + \sum_{\sigma \in \mathcal{F}_{\Gamma,\mathbf{s}}} -F_{\sigma,\mathbf{s}}(u_{\mathcal{D}}) = 0,\ \mathbf{s} \in \mathcal{V}^p \setminus dof_{Dir} \\[2ex]
u_{\mathbf{s}} = u_{Dir},\ \mathbf{s} \in dof_{Dir}
\end{array}\right.$$

On each processor p, at each time step $n \to n+1$, the unknowns are:

$$\overline{c}_{\mathcal{D}}^{p,\{n+1\}} = \left(c_K^{n+1}, c_\sigma^{n+1}, c_{\mathbf{s}}^{n+1}, K \in \overline{\mathcal{M}}^p, \sigma \in \overline{\mathcal{F}}_\Gamma^p, \mathbf{s} \in \overline{\mathcal{V}}^p\right)$$

On each processor p, the equations are:

$$\begin{cases} |\omega_K| \dfrac{c_K^{n+1} - c_K^n}{\Delta t} + \displaystyle\sum_{\nu \in \mathcal{V}_K \cup (\mathcal{F}_K \cap \mathcal{F}_\Gamma)} H_{K,\nu}(c_{\mathcal{D}}^n) = 0, \ K \in \mathcal{M}^p \\ |\omega_\sigma| \dfrac{c_\sigma^{n+1} - c_\sigma^n}{\Delta t} + \displaystyle\sum_{\mathbf{s} \in \mathcal{V}_{\mathcal{F}}} H_{\sigma,\mathbf{s}}(c_{\mathcal{D}}^n) - \sum_{K \in \mathcal{M}_\sigma} H_{K,\sigma}(c_{\mathcal{D}}^n) = 0, \ \sigma \in \mathcal{F}_\Gamma^p \\ |\omega_{\mathbf{s}}| \dfrac{c_{\mathbf{s}}^{n+1} - c_{\mathbf{s}}^n}{\Delta t} - \displaystyle\sum_{K \in \mathcal{M}_{\mathbf{s}}} H_{K,\mathbf{s}}(c_{\mathcal{D}}^n) - \sum_{\sigma \in \mathcal{F}_{\Gamma,\mathbf{s}}} H_{\sigma,\mathbf{s}}(c_{\mathcal{D}}^n) = 0, \ \mathbf{s} \in \mathcal{V}^p \setminus dof_{Dir} \\ c_{\mathbf{s}} = c_{Dir}, \ \mathbf{s} \in dof_{Dir} \end{cases}$$

with the following Explicit Upwind Two Point Fluxes:

$$H_{K,\nu}(c_{\mathcal{D}}^n) = c_K^n F_{K,\nu}(u_{\mathcal{D}})^+ + c_\nu^n F_{K,\nu}(u_{\mathcal{D}})^-$$

$$H_{\sigma,\mathbf{s}}(c_{\mathcal{D}}^n) = c_\sigma^n F_{\sigma,\mathbf{s}}(u_{\mathcal{D}})^+ + c_{\mathbf{s}}^n F_{\sigma,\mathbf{s}}(u_{\mathcal{D}})^-$$

VAG discretization of the concentration equation, MUSCL scheme

On each processor p, two steps for $n \to n+1$:

A second order MUSCL type reconstruction

$$\overline{c}_{\mathcal{D}}^{p,*} = \{c_{K,\nu}^*, K \in \overline{\mathcal{M}}^p, \nu \in dof(K)\} \cup \{c_{\sigma,\nu}^*, \sigma \in \overline{\mathcal{F}}_\Gamma^p, \nu \in dof(\sigma)\}$$

Acceptable slopes:

$$\overline{l}_{\mathcal{D}}^{p,*} = (l_K^*, l_\sigma^*, l_{\mathbf{s}}^*, K \in \overline{\mathcal{M}}^p, \sigma \in \overline{\mathcal{F}}_\Gamma^p, \mathbf{s} \in \overline{\mathcal{V}}^p)$$

Compute

$$\begin{aligned} H_{K,\nu}(c_{\mathcal{D}}^{p,*}, \overline{l}_{\mathcal{D}}^{p,*}) &= \big(x_K^n + \mathcal{P}_{l_K^*}(c_{K,\nu}^* - c_K^n)\big) F_{K,\nu}(u_{\mathcal{D}})^+ \\ &+ \big(x_\nu^n + \mathcal{P}_{l_K^*}(c_{K,\nu}^* - c_K^n)\big) F_{K,\nu}(u_{\mathcal{D}})^- \end{aligned}$$

and

$$\begin{aligned} H_{\sigma,\nu}(c_{\mathcal{D}}^{p,*}, \overline{l}_{\mathcal{D}}^{p,*}) &= \big(x_\sigma^n + \mathcal{P}_{l_\sigma^*}(c_{\sigma,\nu}^* - c_\sigma^n)\big) F_{\sigma,\nu}(u_{\mathcal{D}})^+ \\ &+ \big(x_\nu^n + \mathcal{P}_{l_\sigma^*}(c_{\sigma,\nu}^* - c_\sigma^n)\big) F_{\sigma,\nu}(u_{\mathcal{D}})^- \end{aligned}$$

Outline of ComPASS implementation

0. **Initialization**
    - Global to Local Mesh
    - VAG scheme transmissivities
1. **Pressure equation**
    - Assembling of non-square linear systems

$$\begin{pmatrix} A^p & B^p \\ C^p & D^p \end{pmatrix} \begin{pmatrix} U_{\overline{\mathcal{V}^p} \cup \overline{\mathcal{F}_\Gamma^p}} \\ U_{\overline{K^p}} \end{pmatrix} = RHS^p \qquad \begin{array}{ll} A^p \in \mathbb{R}^{\mathcal{V}^p \cup \mathcal{F}_\Gamma^p} \otimes \mathbb{R}^{\overline{\mathcal{V}^p} \cup \overline{\mathcal{F}_\Gamma^p}} & C^p \in \mathbb{R}^{\overline{\mathcal{M}^p}} \otimes \mathbb{R}^{\overline{\mathcal{V}^p} \cup \overline{\mathcal{F}_\Gamma^p}} \\ B^p \in \mathbb{R}^{\mathcal{V}^p \cup \mathcal{F}_\Gamma^p} \otimes \mathbb{R}^{\overline{\mathcal{M}^p}} & D^p \in \mathbb{R}^{\overline{\mathcal{M}^p}} \otimes \mathbb{R}^{\overline{\mathcal{M}^p}} \end{array}$$

    - Schur complement system $(A^p - B^p(D^p)^{-1}C^p)U_{\overline{\mathcal{V}^p} \cup \overline{\mathcal{F}_\Gamma^p}} = \widetilde{RHS}^p$
    - Resolution by PETSc (Trilinos) $\Rightarrow \; U_{\mathcal{V}^p \cup \mathcal{F}_\Gamma^p}$
    - Synchronization $\Rightarrow \; U_{\overline{\mathcal{V}^p} \cup \overline{\mathcal{F}_\Gamma^p}}$
    - Schur complement $\Rightarrow \; U_{\overline{K^p}}$

Outline of ComPASS implementation

0. **Initialization**
    - Global to Local Mesh
    - VAG scheme transmissivities
1. **Pressure equation**
    - Assembling of non-square linear systems

$$\begin{pmatrix} A^p & B^p \\ C^p & D^p \end{pmatrix} \begin{pmatrix} U_{\overline{\mathcal{V}^p} \cup \overline{\mathcal{F}^p_\Gamma}} \\ U_{\overline{K^p}} \end{pmatrix} = RHS^p \qquad \begin{matrix} A^p \in \mathbb{R}^{\mathcal{V}^p \cup \mathcal{F}^p_\Gamma} \otimes \mathbb{R}^{\overline{\mathcal{V}^p} \cup \overline{\mathcal{F}^p_\Gamma}} & C^p \in \mathbb{R}^{\overline{\mathcal{M}^p}} \otimes \mathbb{R}^{\overline{\mathcal{V}^p} \cup \overline{\mathcal{F}^p_\Gamma}} \\ B^p \in \mathbb{R}^{\mathcal{V}^p \cup \mathcal{F}^p_\Gamma} \otimes \mathbb{R}^{\overline{\mathcal{M}^p}} & D^p \in \mathbb{R}^{\overline{\mathcal{M}^p}} \otimes \mathbb{R}^{\overline{\mathcal{M}^p}} \end{matrix}$$

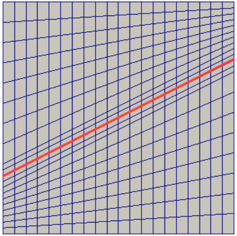
    - Schur complement system $(A^p - B^p(D^p)^{-1}C^p)U_{\overline{\mathcal{V}^p} \cup \overline{\mathcal{F}^p_\Gamma}} = \widetilde{RHS}^p$
    - Resolution by PETSc (Trilinos) $\Rightarrow \ U_{\mathcal{V}^p \cup \mathcal{F}^p_\Gamma}$
    - Synchronization $\Rightarrow \ U_{\overline{\mathcal{V}^p} \cup \overline{\mathcal{F}^p_\Gamma}}$
    - Schur complement $\Rightarrow \ U_{\overline{K^p}}$
3. **CFL condition** $\Rightarrow \ \Delta t$
4. **Time loop for concentration equation**
    - Explicit scheme: $c^{n+1}_{\mathcal{D}} \longleftarrow \overline{c}^n_{\mathcal{D}}$
    - Synchronization: $\overline{c}^{n+1}_{\mathcal{D}} \longleftarrow c^{n+1}_{\mathcal{D}}$

2D analytical example - Test case presentation

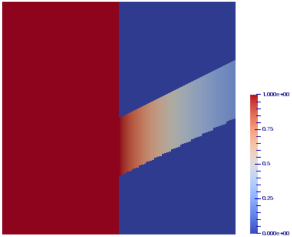
Geometry : 1 planar fracture, $\Omega = (0,1)^2$

Test case configuration

- Isotropic media, $\Lambda_f = 20\ \Lambda_m$, $d_f = 0.01$
- $1d$ linear pressure
- Initial concentration $c = 0$
- Injection at the left side $c = 1$



Example mesh



exact solution for c at $t = 0.5$

Analytical solution : discrete errors on the solution

2D analytical example - Discrete concentration - mesh $1600 \times 1 \times 1600$

2D analytical example - Discrete errors on the concentration c

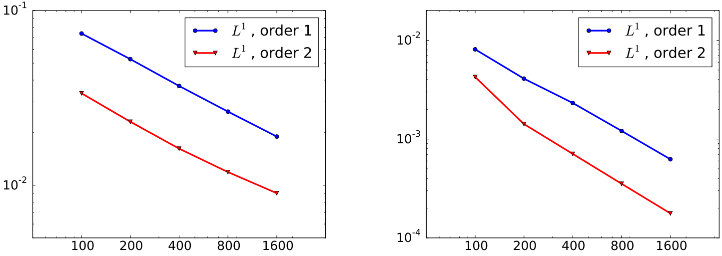
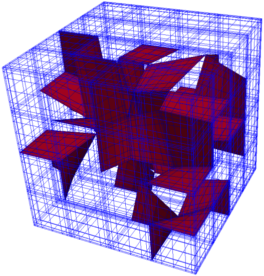


Figure: Errors in the matrix domain (left) and in the fracture (right).

Geometry : 3D network of fractures of $\Omega = (0,1)^3$ with hexahedral meshes

Test case configuration

- Isotropic media, $\Lambda_f = 20\ \Lambda_m$, $d_f = 0.01$
- non linear pressure
- Initial concentration $c = 0$
- Injection at the bottom side $c = 1$



3D fracture network - Discrete concentration - 128^3 cells

N_p	2	4	8	16	32	64	128	256	512
GMRES + Boomer AMG	15	15	15	15	15	16	15	15	15
GMRES + Aggregation AMG	65	70	98	95	59	86	65	91	54
GMRES + ILU(0)	751	707	655	644	648	634	633	624	613
GMRES + ILU(1)	> 1000								
GMRES + ILU(2)	> 1000								
BiCGSTAB + Boomer AMG	9	9	9	9	9	10	9	9	10
BiCGSTAB + ILU(0)	508	476	484	503	473	513	491	487	484
BiCGSTAB + ILU(1)	> 1000								
BiCGSTAB + ILU(2)	> 1000								

Table: Number of iterations vs. number of MPI processes with hexahedral mesh.

Mesh: 2.1×10^6 cells, 2.1×10^6 vertexes and 5.2×10^4 fractures faces

3D fracture network with hexahedral mesh - strong scaling

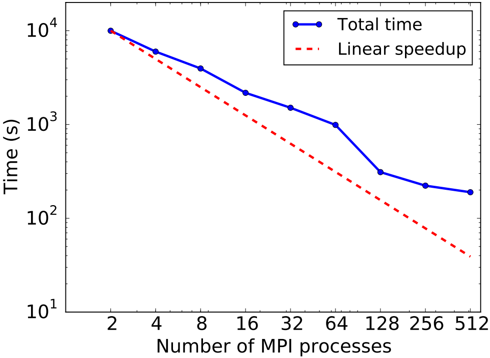
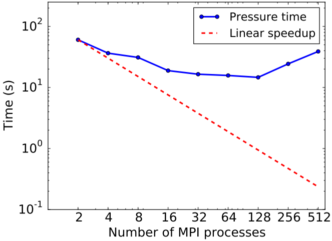


Figure: Total computation times vs. number of MPI processes with hexahedral mesh.

Mesh: $128^3 \Rightarrow$: 2.1×10^6 cells, 2.1×10^6 vertexes and 5.2×10^4 fractures faces

cluster Cicada: `http://calculs.unice.fr/` - 72 Cpu nodes: 16 cores (2 Intel Sandy Bridge E5-2670), 64 GB, GCC 4.9.1, OpenMPI 1.8.2, 1 core/MPI

3D fracture network with hexahedral mesh



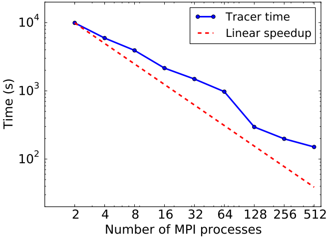


Figure: Computation times for pressure (left) and computation times for tracer (right) vs. number of MPI processes with hexahedral mesh.

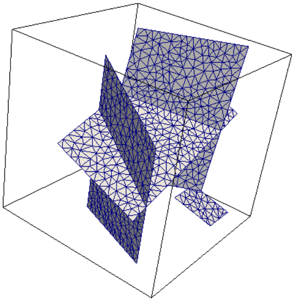
Mesh: 2.1×10^6 cells, 2.1×10^6 vertexes and 5.2×10^4 fractures faces

3D fracture network with tetrahedral mesh

Geometry : 3D network of fractures of $\Omega = (0,1)^3$ with tetrahedral meshes

Test case configuration

- Isotropic media, $\Lambda_f = 20\ \Lambda_m$, $d_f = 0.01$
- non linear pressure
- Initial concentration $c = 0$
- Injection at the bottom side $c = 1$



N_p	2	4	8	16	32	64	128	256	512
GMRES + Boomer AMG	11	12	12	12	12	12	12	12	12
GMRES + ILU(0)	-	725	717	682	667	656	644	629	612
GMRES + ILU(1)	> 1000								
GMRES + ILU(2)	154	153	152	151	149	147	144	142	140
BiCGSTAB + Boomer AMG	8	7	8	8	8	8	8	8	8
BiCGSTAB + ILU(0)	565	513	527	544	535	483	489	483	473
BiCGSTAB + ILU(1)	374	367	432	404	317	382	348	307	271
BiCGSTAB + ILU(2)	104	105	101	103	98	106	97	93	103

Table: Number of iterations vs. number of MPI processes with tetrahedral mesh.

Mesh: 6.2×10^6 cells, 9.7×10^5 vertexes and 7.1×10^4 fracture faces

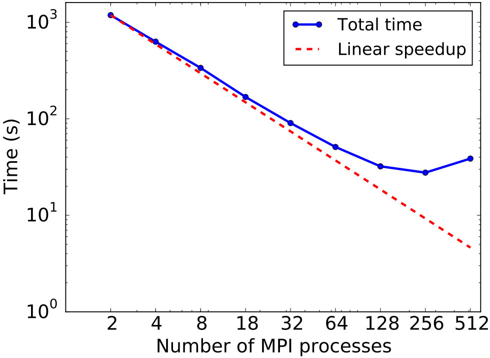
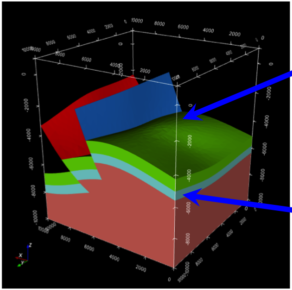


Figure: Total computation times vs. number of MPI processes with tetrahedral mesh.

Mesh: 6.2×10^6 cells, 9.7×10^5 vertexes and 7.1×10^4 fracture faces

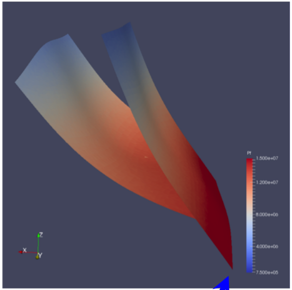
cluster Cicada: `http://calculs.unice.fr/` - 72 Cpu nodes: 16 cores (2 Intel Sandy Bridge E5-2670), 64 GB, GCC 4.9.1, OpenMPI 1.8.2, 1 core/MPI

A real case from S. Lopez (BRGM)



caprock layer

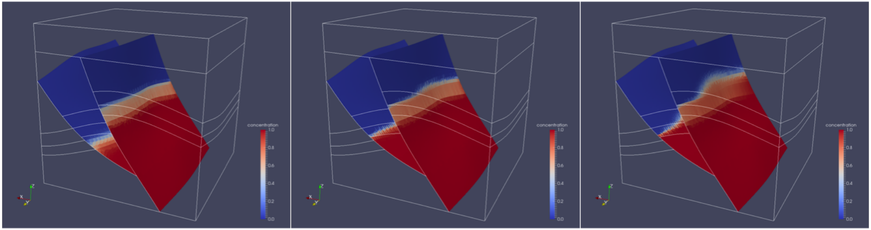
aquifer layer

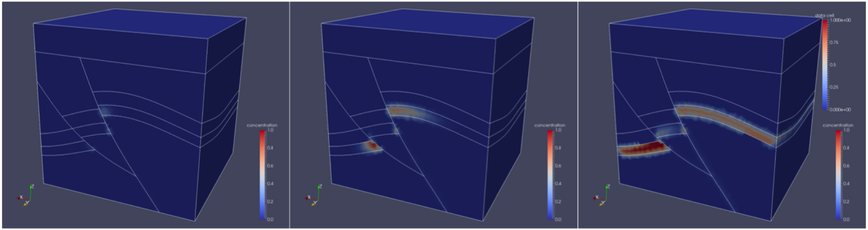


tracer injection
along the fault

Geology simulation

A real case from S. Lopez (BRGM)





Ongoing works

- Multiphase compositional Darcy flux in fracture porous media
    - N_P phases, N_C components
    - Model is defined by a matrix of size $N_C \times N_P$
- Applications (S. Lopez at BRGM)
    - Real case studies, geothermal reservoir simulation in Guadeloupe
- Code
    - Optimization (OpenMP?)
    - User-friendly interface

Thanks

Thanks for your attention!

This work is also supported by “institut Carnot” (ANR, INRIA, BRGM).

This work was granted access to the HPC and visualization resources of the ”Centre de Calcul Interactif” hosted by University Nice Sophia Antipolis.

VAG discretization for a tracer problem on a Discrete Fracture Network (DFN)

⋆ **Discrete Unknowns** ⋆

$$u_{\mathcal{D}} = (u_K, u_\sigma, u_{\mathbf{s}}, K \in \mathcal{M}, \sigma \in \mathcal{F}_\Gamma, \mathbf{s} \in \mathcal{V})$$

⋆ **Discrete Operators** ⋆

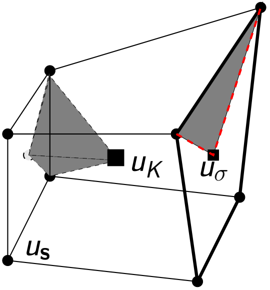
In the matrix : $\forall K \in \mathcal{M}$

$$\Pi_{\mathcal{D}_m} u_{\mathcal{D}}(\boldsymbol{x}) = \begin{cases} u_K & \text{for all } \boldsymbol{x} \in \omega_K \\ u_{\mathbf{s}} & \text{for all } \boldsymbol{x} \in \omega_{\mathbf{s}} \end{cases}$$

$$\nabla_{\mathcal{D}_m} u_{\mathcal{D}} = \nabla_{\mathcal{T}_{K,\sigma,e}} u_{\mathcal{D}} \ , \ \sigma \in \mathcal{F}_K \ , \ e \in \mathcal{E}_\sigma$$

$\mathcal{T}_{K,\sigma,e}$: tetrahedron joining cell center $\boldsymbol{x}_K$ to triangle $T_{\sigma,e}$

$\forall \sigma \in \mathcal{F} \setminus \mathcal{F}_\Gamma$ interpolation of the face unknown $u_\sigma = \dfrac{\sum_{\mathbf{s} \in \mathcal{V}_\sigma} u_{\mathbf{s}}}{\mathrm{card}(\mathcal{V}_\sigma)}$



In the fracture : $\forall \sigma \in \mathcal{F}_\Gamma$

$$\Pi_{\mathcal{D}_f} u_{\mathcal{D}}(\boldsymbol{x}) = \begin{cases} u_\sigma & \text{for all } \boldsymbol{x} \in \omega_\sigma \\ u_{\mathbf{s}} & \text{for all } \boldsymbol{x} \in \omega_{\mathbf{s}} \end{cases}$$

$$\nabla_{\mathcal{D}_f} u_{\mathcal{D}} = \nabla_{T_{\sigma,e}} u_{\mathcal{D}} \ , \ e \in \mathcal{E}_\sigma$$

$T_{\sigma,e}$: triangle joining edge e to face center $\boldsymbol{x}_\sigma$