MPCube Development: modelisation for flow and transport in porous media in HPC context

Ph. Montarnal, Th. Abballe, F. Caro, E. Laucoin,
DEN Saclay DM2S/SFME/LSET

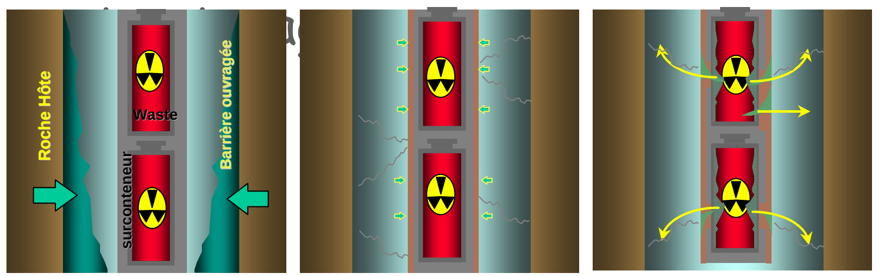
Outline

- Background
- Development Strategy
- Scalar diffusion : validation and cement paste application
- Two-phase flow porous media module
- Two directions to increase the precision/complexity keeping a “industrial” cpu time
  - ✓ *Multiscale FV/FE*
  - ✓ *Adaptive mesh refinement and a posteriori error*
- Link with MOMAS/CALCUL activities
- Conclusion and prospects

Background

- Simulation of flow and reactive transport at several scales

  - ✓ Material degradation (cement, glass, iron …)
  - ✓ Phenomenological simulation and performance study of nuclear



  - ✓ Industrial and Nuclear site pollution

Background

- First work done at CEA
  - ✓ Development in the context of the Castem code
  - ✓ Integration in Alliances plat-form (co-developped with ANDRA & EDF, using Salome)
  - ✓ Simulation with about 700 000 elements meshes
  - ➔ Use for the ANDRA report in 2005

- New step after 2006 : need to increase the accuracy of the computation in order to quantity the uncertainties
  - *Increase the complexity of the geometry*
  - *Take into account multi-phase flow : hydrogen migration due to corrosion, description of the saturation phase*
  - *Heterogeneous materials : Identification of fine scale behaviour in order to determine properties at the macro-scale*
  - *Material degradation : corrosion, glass, cement → Sharp coupling*

➔ Development of a new code in a HPC context

Development strategy

- Objectives
  - ✓ Development of numerical schemes appropriate for
    - *multi-physics problems : hydraulic, transport, chemistry transport, two-phase flow*
    - → *discretization schemes for scalar equation and systems of convection-diffusion*
    - → *easy to couple (cell centered, modular architectures)*
    - *Porous media characteristics : heterogeneous and anisotropic → implicit time schemes, unstructured meshes, adapted discretization schemes*
  - ✓ Use an software context adapted
    - *both to department clusters and massively parallel clusters of data processing centres*
    - *problem evolution (New model implementation has to be easy)*
    - *parallelism performance evolution*
- Choices
  - ✓ Use an existing parallel frame-work developed at CEA for TrioU and OVAP codes
    - *Memory management and parallel data entry; Distributed operations on vectors and matrices;*
    - *Ability to handle unstructured meshes; Link with partitioning tools;*
    - *Link with standard libraries (PETSC, HYPRE and SPARSKIT) for the resolution of sparse linear systems;*
    - *Possibilities of integration in the Salome platform (Med format)*

Development strategy

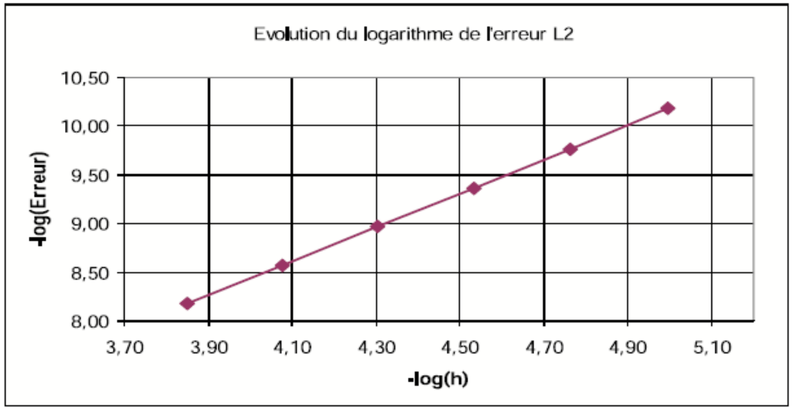
- ✓ Develop a generic framework for non-linear convection-diffusion equation able to be coupled with chemistry term

$$M(u_1,u_2,..)\partial_t \begin{bmatrix} u_1 \\ u_2 \\ \\ \end{bmatrix} - \mathrm{div}\left(\mathbf{A}(u_1,u_2,..) \begin{bmatrix} \nabla u_1 \\ \nabla u_2 \\ \ldots \end{bmatrix} \right) = \begin{pmatrix} \mathcal{F}_{u_1}(u_1,u_2,..) \\ \mathcal{F}_{u_2}(u_1,u_2,..) \\ \ldots \end{pmatrix}$$

- ✓ Use the numerical schemes which were validated previously in Castem : cell centered FV schemes suited to heterogoneous and anisotropic media (VF DIAM, MPFA, VF SYM, VF MON)
- ✓ Explore complementary ways for performance improvement :
  - *Adaptive Mesh Refinement and a priori error in the context of HPC*
  - *Other parallelism paradigm (domain decomposition, multiscale schemes)*
  - *Linear solver improvement*
  - *…*

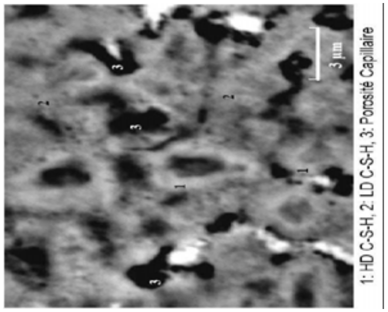
Scalar diffusion

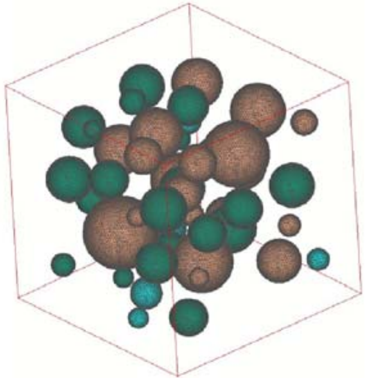
- Validation on analytical solution



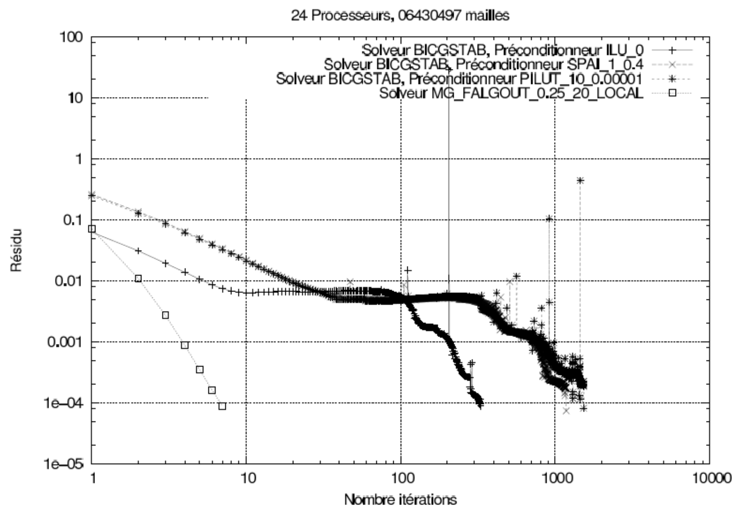
L^2 error evolution in logarithmic scale according to the mesh size (slope $\simeq 2$)

- Cement paste application case
  - ✓ Stationarry and diffusive problem on a cementitious material EVR
  - ✓ Diffusion coefficient D isotropic and heterogeneous, 4 mediums(D in $[10^{-19}, 10^{-11}]$ $m^2.s^{-1}$)
  - ✓ Geometry :
    - *Cube of 603 µm3*
    - *21 spheres for medium 1, 20 spheres for medium 2 and 6*
    - *spheres for medium 3*





Scalar diffusion

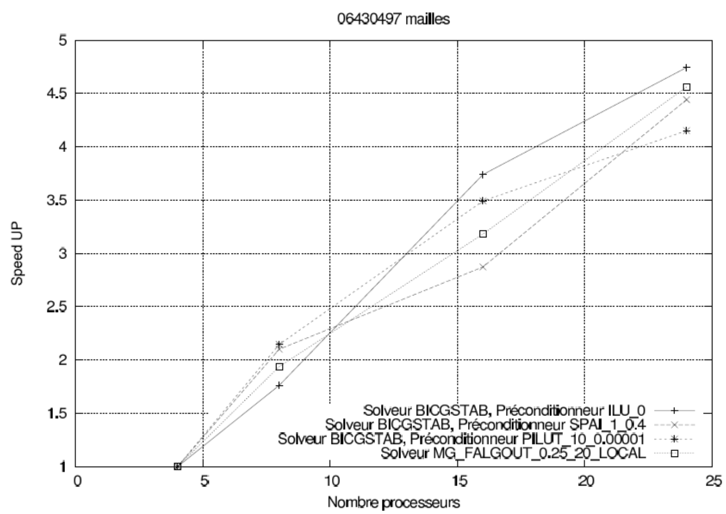


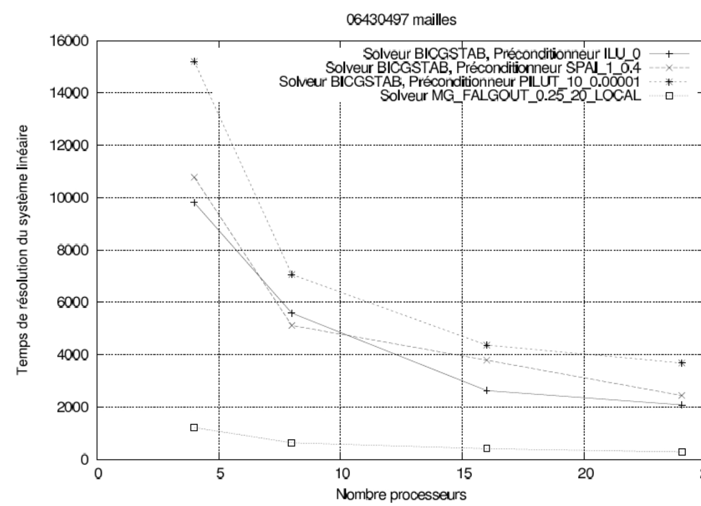
Number of iteration, CPU time evolution for the linear system resolution and performance depending on the choice of solver and preconditioner

Numerical scheme : FV Symetric

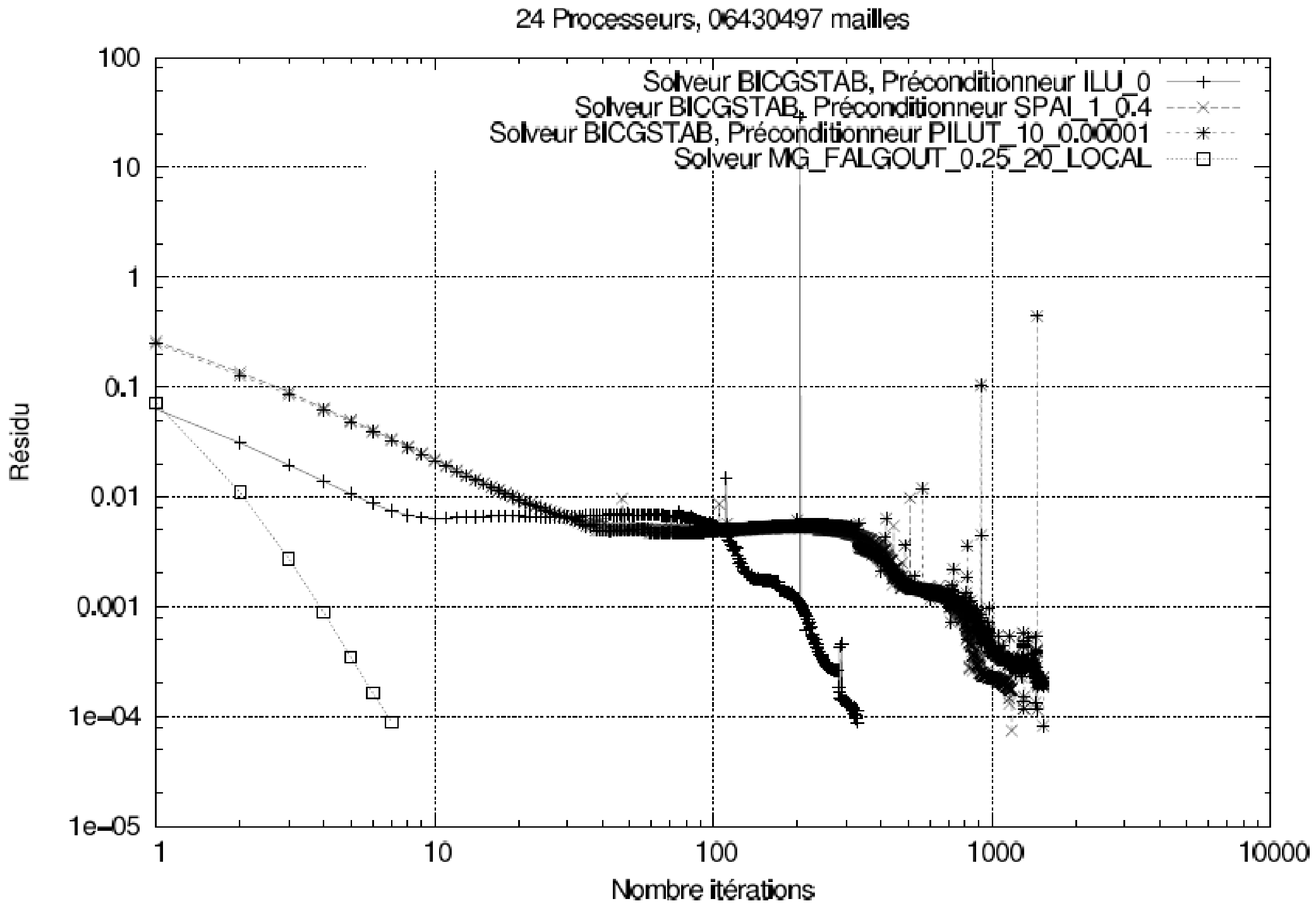
Number of cells ≃ 6 500 00

➔ Best results with AMG Hypre solver



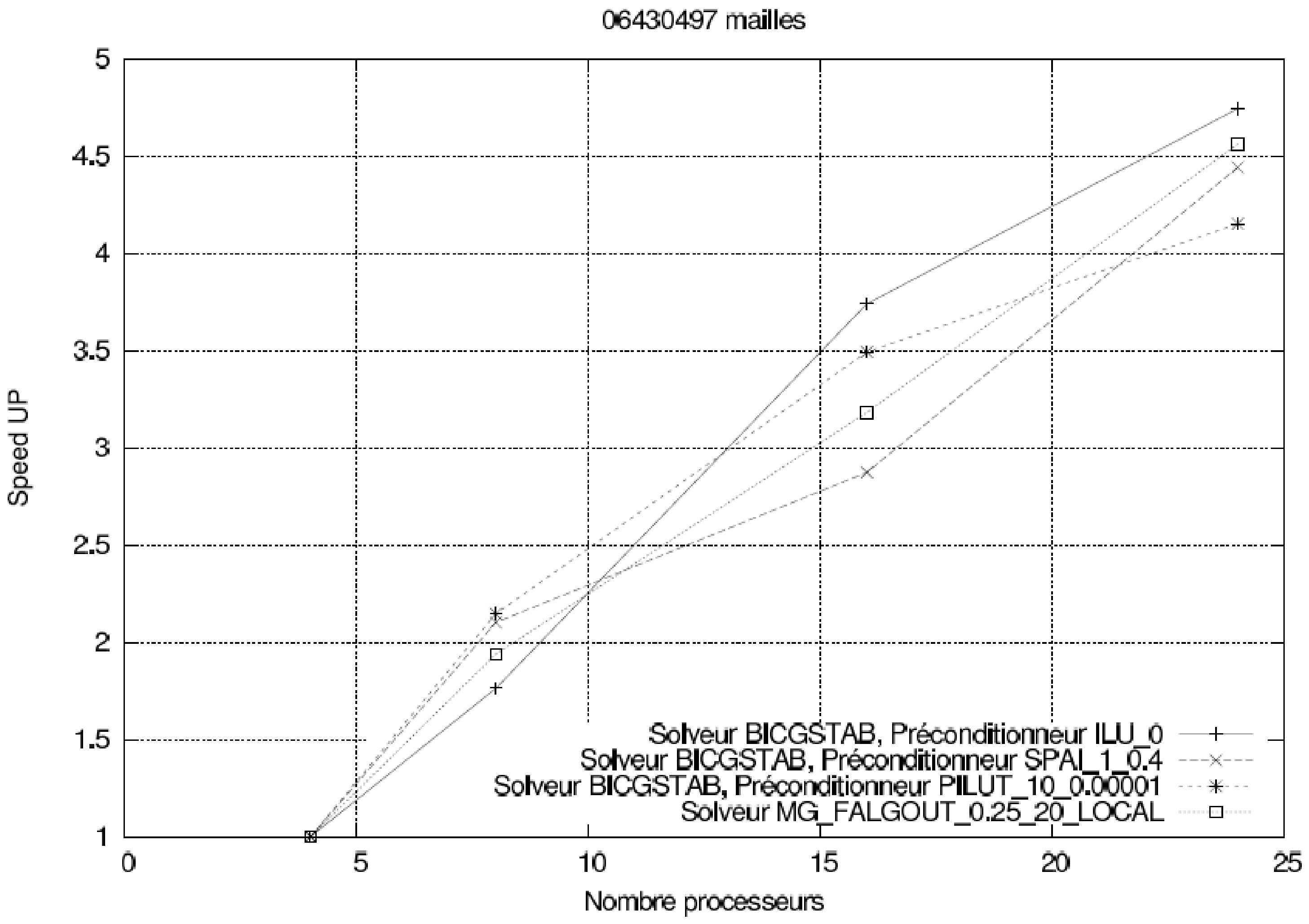


Threshold

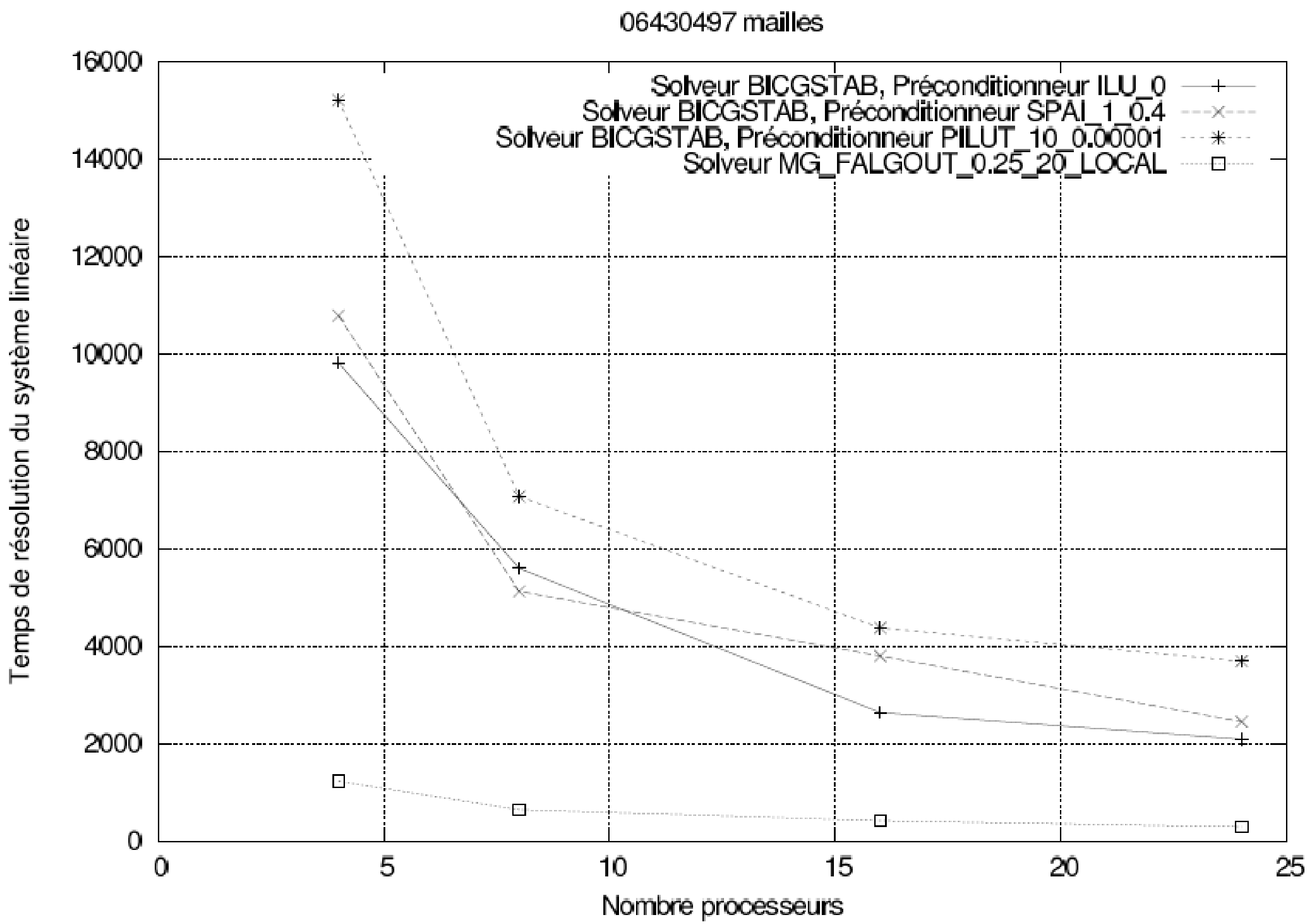


24 Processeurs, 06430497 mailles
Solveur BICGSTAB, Préconditionneur ILU_0
Solveur BICGSTAB, Préconditionneur SPAI_1_0.4
Solveur BICGSTAB, Préconditionneur PILUT_10_0.00001
Solveur MG_FALGOUT_0.25_20_LOCAL
Résidu
100
10
1
0.1
0.01
0.001
1e–04
1e–05
1
10
100
1000
10000
Nombre itérations

Performances



CPU time



06430497 mailles
Solveur BICGSTAB, Préconditionneur ILU_0
Solveur BICGSTAB, Préconditionneur SPAI_1_0.4
Solveur BICGSTAB, Préconditionneur PILUT_10_0.00001
Solveur MG_FALGOUT_0.25_20_LOCAL
Temps de résolution du système linéaire
Nombre processeurs
0
2000
4000
6000
8000
10000
12000
14000
16000
0
5
10
15
20
25

Two-phase flow porous media module

- Hypothesis
  1. 2 components (H2 and H2O) and 2 phases (liquid and gas)
  2. Mass conservation for each component
  3. Darcy's flow for each phase
  4. Fick's law for hydrogen diffusion
  5. Capillary pressure between liquid pressure and gas pressure: $P_c = P_g - P_l$
  6. Isothermal flow
  7. Compressible gas phase
  8. Perfect gas law for each component into gas phase
  9. Incompressible liquid phase
  10. Van Genuchten and Mualen laws for relative permeabilities and capillary pressure
  11. Equilibrium between vapor water and liquid water
  12. Equilibrium between dissolved hydrogen and gaz hydrogen using Henry's law
  13. Dalton's law for partial pressures of each component

Two-phase flow porous media module

- Physical equations

$$\begin{cases} \partial_t(\Phi\rho) + \mathrm{div}(\rho u) = \sum_\alpha Q_\alpha \\ \partial_t\left(\Phi\rho^{H_2}\right) + \mathrm{div}\left((\rho\mathbf{u})^{H_2} - \sum_\alpha \rho_\alpha D_\alpha^{H_2} \nabla X_\alpha^{H_2} \right) = \sum_\alpha Q_\alpha^{H_2} \end{cases}$$

with X^{H2}_α = mass fractions of hydrogen in the α phase (α=l,g) and Q^β_α source terms

- Closure laws :
  - ✓ Velocities

$$\mathbf{u}_\alpha = -k \frac{k_{r_\alpha}(S_l)}{\mu_\alpha} \nabla P_\alpha$$

  - ✓ Diffusion coefficients (non linear)

$$D_\alpha^{H_\alpha} = D_\alpha^{H_\alpha}(S_l, P_g)$$

Two-phase flow porous media module

- Mathematical model :

$$M(u,v)\partial_t \begin{bmatrix} u \\ v \end{bmatrix} - \mathrm{div}\left(\mathbf{A}(u,v) \begin{bmatrix} \nabla u \\ \nabla v \end{bmatrix} \right) = \begin{pmatrix} \mathcal{F}_u \\ \mathcal{F}_v \end{pmatrix}$$

- where u and v are the choosen unknows and M , A matrix non linearly dependent of unknows
- **Remarks** :
  - ✓ Different variants of the model can be addressed by this mathematical frame-work
  - ✓ Choice of unknowns is difficult and depends of the problem (for example we can choose liquid saturation-gas pressure, gas saturation-gas pressure, capillarity pressure-gas pressure, liquid pressure-gas pressure, and more . . .)

Two-phase flow porous media module

- Cell-center FV discretisation

$$\mathcal{M}(\boldsymbol{u},\boldsymbol{v})\partial_t \begin{bmatrix} \boldsymbol{u} \\ \boldsymbol{v} \end{bmatrix} - \mathrm{div}\left(\mathcal{A}(\boldsymbol{u},\boldsymbol{v}) \begin{bmatrix} \boldsymbol{u} \\ \boldsymbol{v} \end{bmatrix} \right) = \begin{pmatrix} \mathcal{F}_u \\ \mathcal{F}_v \end{pmatrix}$$

- Implicit Euler Scheme for time discretisation

$$\frac{\mathcal{M}(\boldsymbol{u}^{n+1},\boldsymbol{v}^{n+1})}{\Delta t} \begin{bmatrix} \boldsymbol{u}^{n+1} - \boldsymbol{u}^{n} \\ \boldsymbol{v}^{n+1} - \boldsymbol{v}^{n} \end{bmatrix} - \mathrm{div}\left(\mathcal{A}(\boldsymbol{u}^{n+1},\boldsymbol{v}^{n+1}) \begin{bmatrix} \boldsymbol{u}^{n+1} \\ \boldsymbol{v}^{n+1} \end{bmatrix} \right) = \begin{pmatrix} \mathcal{F}_u \\ \mathcal{F}_v \end{pmatrix}$$

- Fixed point method for the global non linear system resolution with unknows u^{n+1} and v^{n+1}

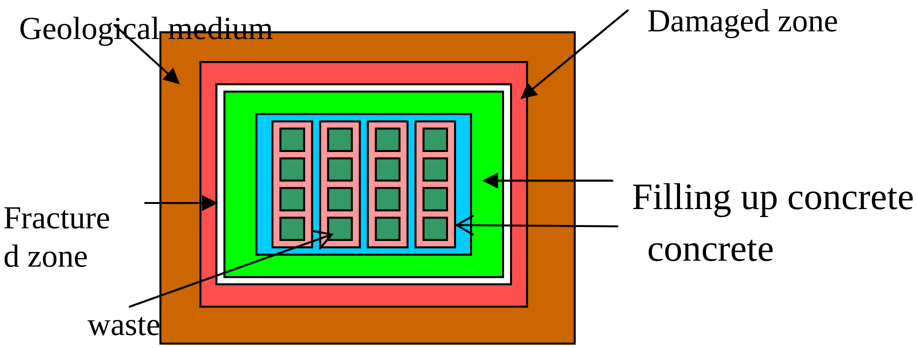
$$\mathcal{M}(\boldsymbol{u}_k^{n+1},\boldsymbol{v}_k^{n+1})\partial_t \begin{bmatrix} \boldsymbol{u}_{k+1}^{n+1} \\ \boldsymbol{v}_{k+1}^{n+1} \end{bmatrix} - \mathrm{div}\left(\mathcal{A}(\boldsymbol{u}_k^{n+1},\boldsymbol{v}_k^{n+1}) \begin{bmatrix} \boldsymbol{u}_{k+1}^{n+1} \\ \boldsymbol{v}_{k+1}^{n+1} \end{bmatrix} \right) = \begin{pmatrix} \mathcal{F}_u \\ \mathcal{F}_v \end{pmatrix}$$

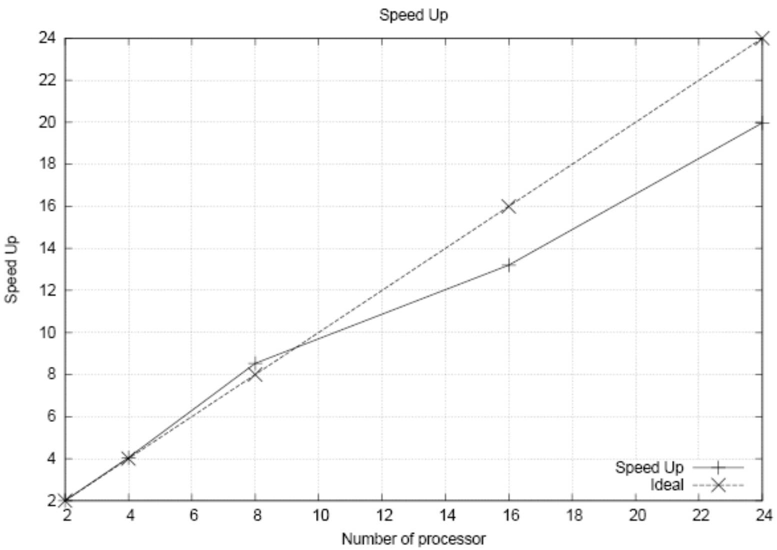
Two-phase flow porous media module

- Validation
  - ✓ Homogenous Unstationary Problem
  - ✓ Heterogeneous Stationary Problem
  - ➔ Keep a ***second (?)*** order convergence
  - ✓ Momas test case

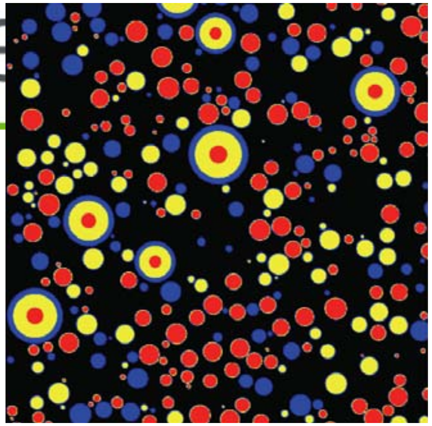
Two-phase flow porous media module

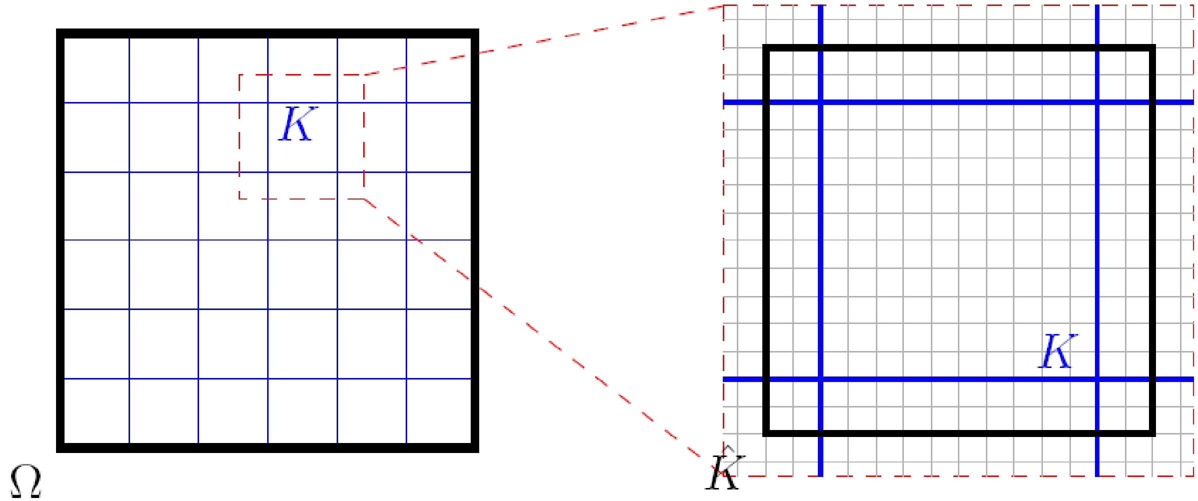
- Couplex gaz 1 b
  - Numerical parameters :
    - *Linear solver : BICGSTAB*
    - *Preconditioner : SPAI*
    - *Threshold 10^{-10} for the linear solver and 10^{-4} for the non linear solver*
    - *Mesh : 466 522 cells*
    - *Number of time step : 115 (400 years)*
  - CPU time over 2 processors ≃ 1 day 1/4 and over 24 processors, 3 hours.





Numerical Homogeneization : multi-scale FV/FE (TH. Abballe PhD Thesis at CEA with G. Allaire)





Coarse layer

- Finite Elements method
- Each base function is built from local resolutions on the fine layer

Fine layer

•Macro-element K and cell $\hat{K}$ (in black) by increasing K with a fraction ρ of its neighbours.
•Finite Volumes method
•Computations on coarse cells are independent
•Allows to capture low-scale details

➔Access to finer micro-structures without a global computation

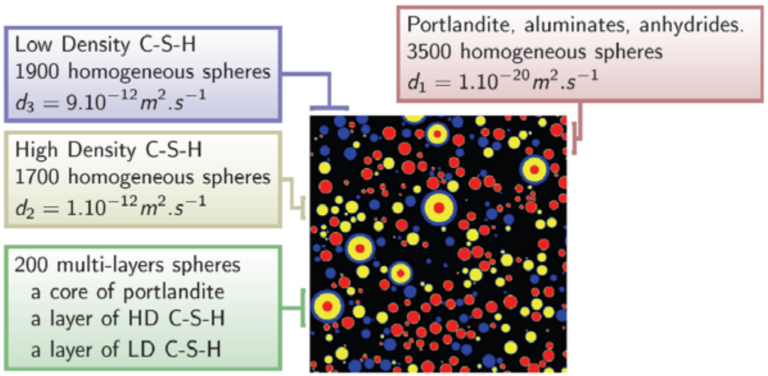
➔Two levels of parallelism

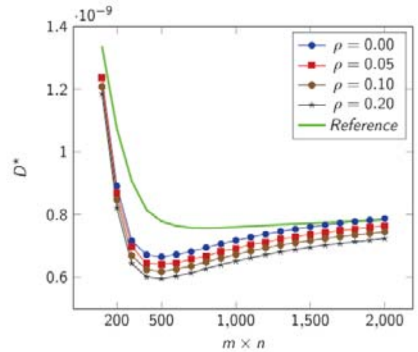
•Outer-cell : each local resolution solve independently

•Inner-cell : paralell solver for each cell computation

Numerical Homogeneization : multi-scale FV/FE coupling (TH. Abballe PhD Thesis at CEA with G. Allaire)

- Determination of an homogenized diffusion from the ingoing and outgoing fluxes.
- Work on 2D geometry (Python prototype)

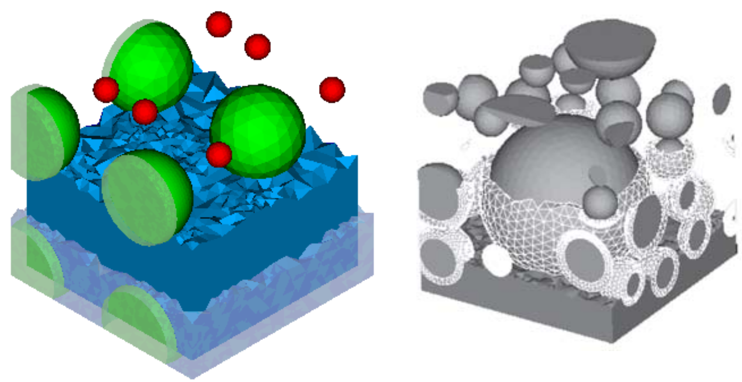




The number of macro-elements K was fixed to 100 as we improved the fine scale.

- The coefficient followed a similar evolution for the all values of ρ
- Gap from the direct resolution (in green) inferior to 10%

- Integration in MPCube to handle 3D geometry with transition layers
- Further work
  - ✓ 3D computation on VER in order to obtain effective coefficients
  - ✓ Use of Discontinuous Galerkin methods to solve the coarse problem, instead of a classical Finite Element method



3D geometry

Adaptive mesh refinement and a posteriori error

- Objectives
  - Improvement of global accuracy through local mesh adaptation
  - Error control using a posteriori error estimates
  - Integrated approach (no external meshing software)
- Constraints
  - Complex physical configurations ➔ HPC is essential !
  - AMR leads to load imbalance ➔ need for relevant load balancing algorithms
  - Numerical accuracy relies on mesh quality ➔ Geometric adaptation must retain mesh quality

- Choosen approach for adaptive mesh refinement
  - *Geometric adaptation through regular refinement*

  - *Pros :*
    - Mesh quality is conserved
    - Algebraic refinement ➔ implementation simplification
  - *Cons :*
    - Generates non-conformities

Adaptive mesh refinement and a posteriori error

- ✓ Development of new FV schemes supporting mesh non-conformities (PhD Thanh Hai Ong with C. Le Potier (CEA) and J. Droniou (U. Montpellier))
  - *Extension of existing FV schemes (FV-Diamond, FV-Sym, ...)*
  - *This is one the objectives of the ANR Project VF-Sitcom*
- ✓ Implementation of dynamic load balancing strategies
  - *Relying on graph partitionning heuristics*
  - *Load balance should be ensured for both numerical resolution and geometric adaptation*

➢ A posteriori error estimates

- ✓ Work done by A. Ern (ENPC), M. Vhoralik (P6) and P. Omnes (CEA) with PhD students Anh Ha Le and Nancy Chalhoub
- ✓ Objectives
  - *Unstationary problems (diffusion-convection equation)*
  - *Non-linearities in unsaturated flow*
  - *Coupled error for flow and transport*
- ✓ Integration in MPCube
  - *Error estimates for diffusion equation with VF-Diam done in 2008*
  - *Further integration will be done after (during?) the PhD thesis*

Link with MOMAS/CALCUL activities

- MPCube can be a way of MOMAS/CALCUL research integration in industrial tools
- It was (will be) the case for
  - ✓ Two-phase flow
  - ✓ Multiscale EF/VF method
  - ✓ A posteriori error estimator
  - ✓ Space-**time** domain decomposition
- CEA is also motivated to include highly scalable linear solver
- The development is currently done at CEA but bilateral cooperation on the development can be done

Conclusion and prospects

- MPCube is a code for non-linear convection-diffusion systems in a parallel context
  - ✓ First applications was done on diffusion on heterogeneous media and multi-phase flow
  - ✓ Specific functionalities was also developed for multi-scale FV/FE, Adaptive mesh refinement and a posteriori error
  - ✓ It's a way of academic research integration in industrial tools

- Short term prospects
  - ✓ Improve the multi-phase model and implementation
  - ✓ Test and analysis of computational performance on large clusters
- Middle/Long term prospects
  - ✓ New parallel preconditioning techniques will be tested : ILU factorization with tangential filtering from PETAL ANR project, improvement of AMG solver
  - ✓ Space-**time** domain decomposition in order to discretize the different zones with space and time steps adapted to their physical properties (PhD of P.M. Berthe at CEA with L. Halpern, P13)
  - ✓ Extend to other applications (polymer evolution,)