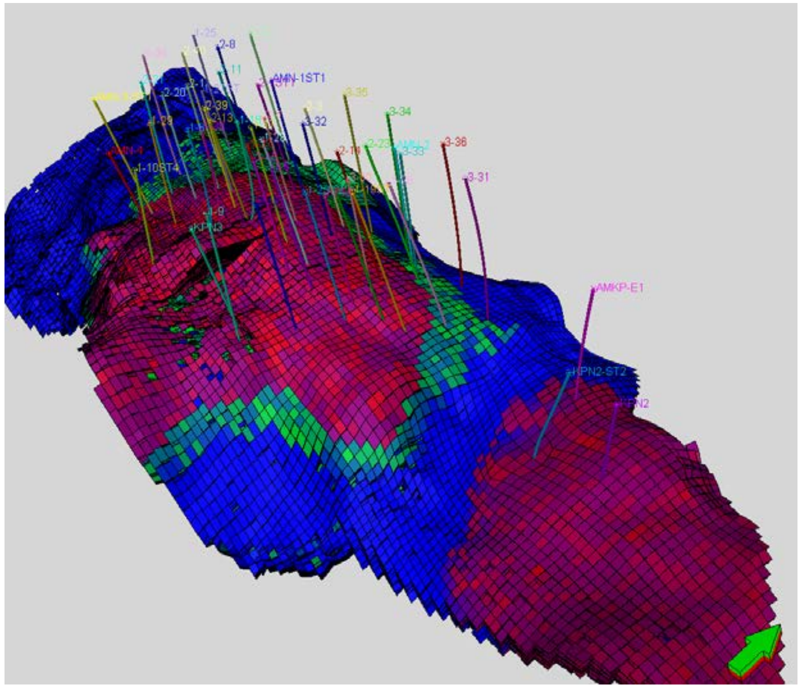
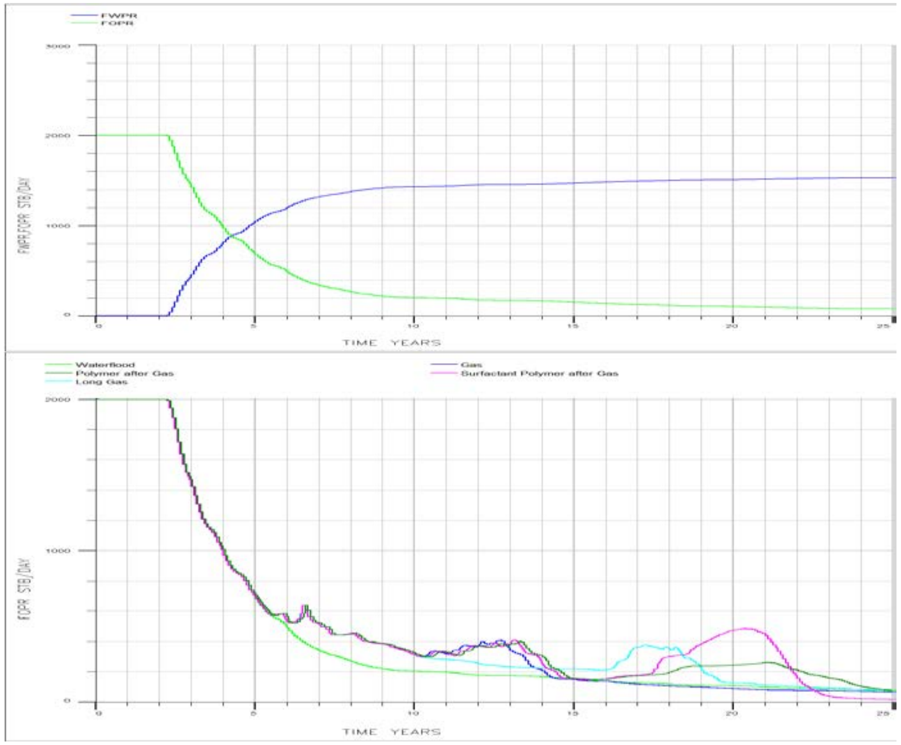
IMPROVING THE SCALABILITY OF RESERVOIR SIMULATION ON MULTICORE ARCHITECTURE

Pascal Hénon

ORAP

RESERVOIR SIMULATION : PURPOSES

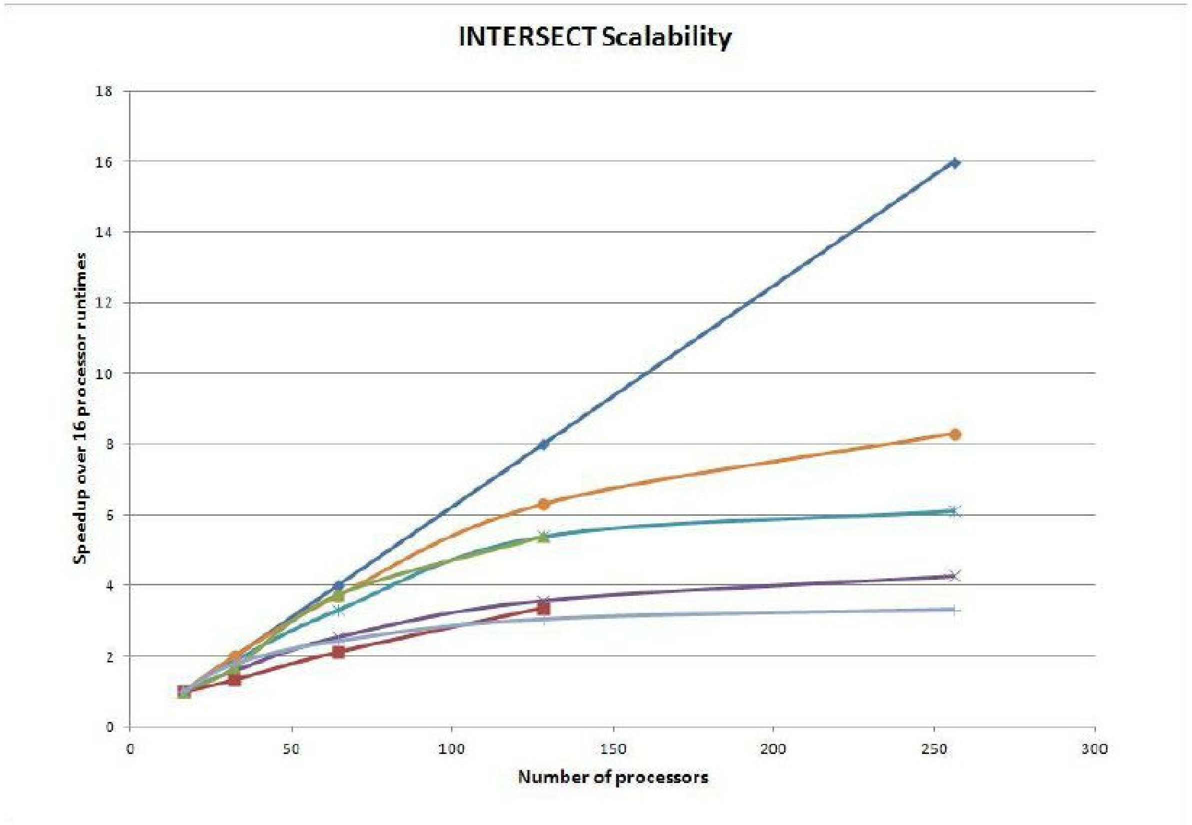




- Estimation of Recovery Factor, Production plateau, EOR
- We have a limited knowledge of underground properties (fault, kr, ..)
- A big part of the work consists in « history matching » : needs many runs

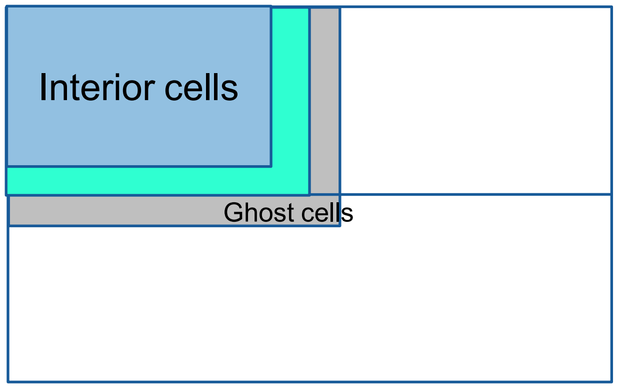
INTERSECT : PARALLEL PERFORMANCE

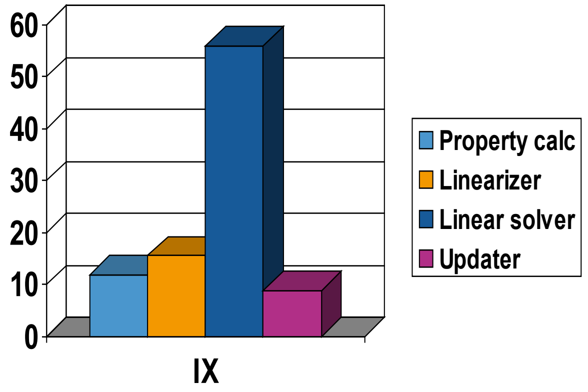
- Scalability of a few models : (upper one =13M cells BO: around 50k cells/processors)



Source : Schlumberger : SPE ACTE, Oct 31th 2011

SCALABILITY BOTTLENECK IN RESERVOIR SIMULATION



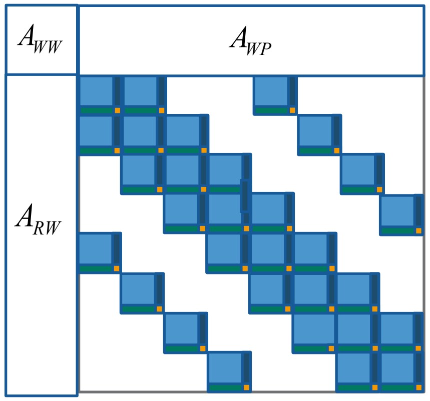


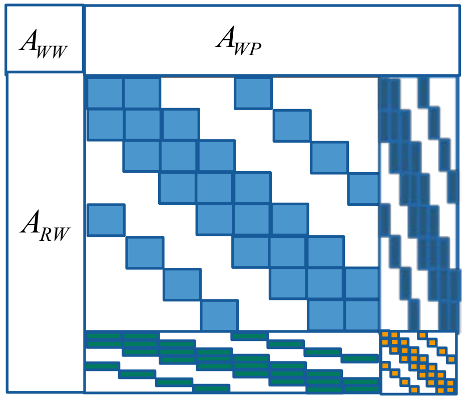
- Distributed memory framework (MPI)
- Load balancing : work per reservoir cell varies during the simulation.
- Linear solver method CPR-AMG is scalable with number of unknowns (weak scalability) but poorly scalable with number of processors (strong scalability)

LINEAR SOLVER : CONSTRAINT PRESSURE RESIDUAL

- The Constraint Pressure Residual (CPR) solver : this solver specific to reservoir is a two stages method (John Wallis and co. SPE 1985) :

$$\begin{pmatrix} A_{WW} & A_{WR} \\ A_{RW} & A_{RR} \end{pmatrix} \qquad P^t.A_{RR}.P = \begin{pmatrix} A_{SS} & A_{SP} \\ A_{PS} & A_{PP} \end{pmatrix} \qquad \begin{pmatrix} A_{WW} & A_{WS} & A_{WP} \\ A_{SW} & A_{SS} & A_{SP} \\ A_{PW} & A_{PS} & A_{PP} \end{pmatrix}$$

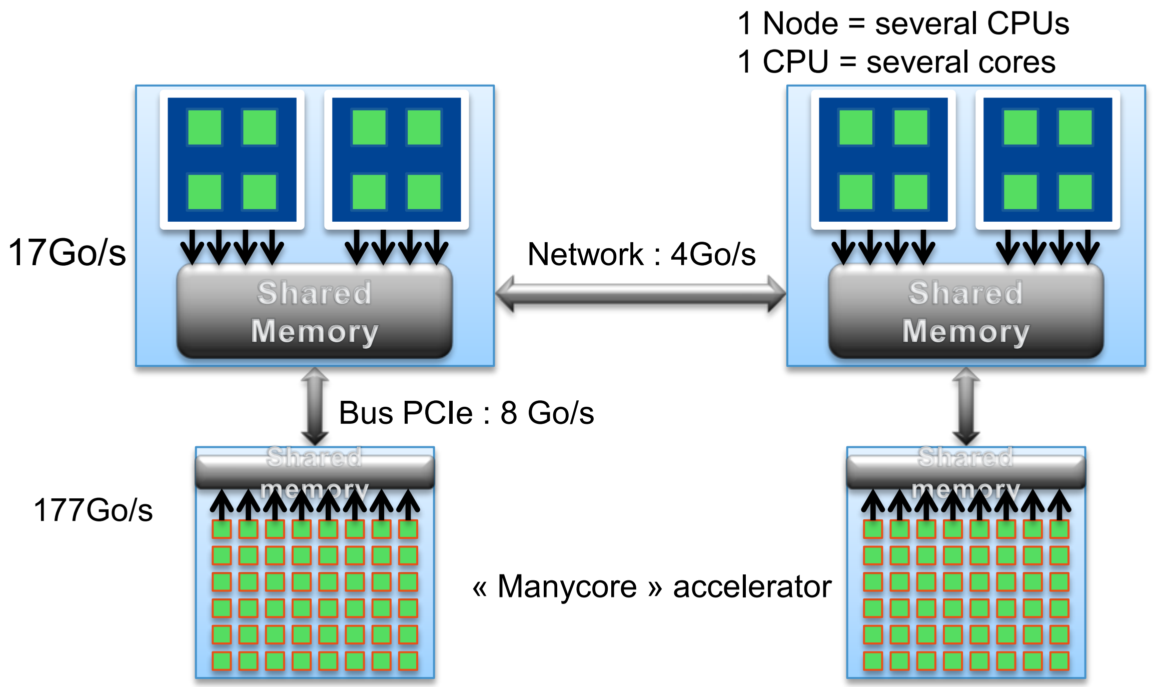




$$\boxed{M_{CPR}^{-1} = M_2^{-1}(I - A.M_1^{-1}) + M_1^{-1}}$$

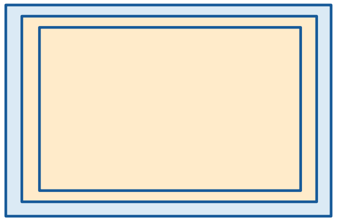
- M_1 : **1er stage is global : find approximate pressure** (near-elliptic problem : AMG)
- M_2 : **2nd stage is applied on the A_{RR} system** (eg BILU(0)) : block preconditioner

ADAPT LINEAR SOLVER AND PROGRAMMATION TO SUPERCOMPUTER ARCHITECTURE

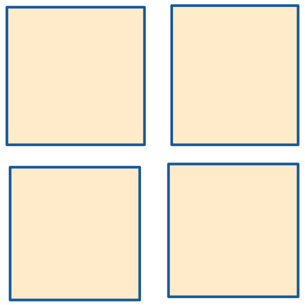


- Intel « mainstream » processor evolution : Nehalem 4 cores, Westmere 6 cores, SandyBridge 8 cores, Haswell 14 cores …
- Manycore processor Xeon Phi 60 cores (x4 threads)

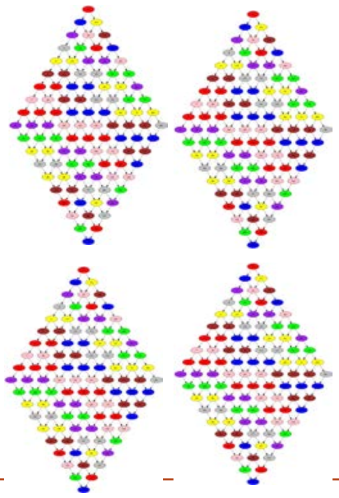
CPR IMPLEMENTATION: THREE LEVELS OF PARALLELISM



- **MPI level : number of domains >= number of cluster node. Static partition : load balancing is now at the cluster node level (not the cores)**



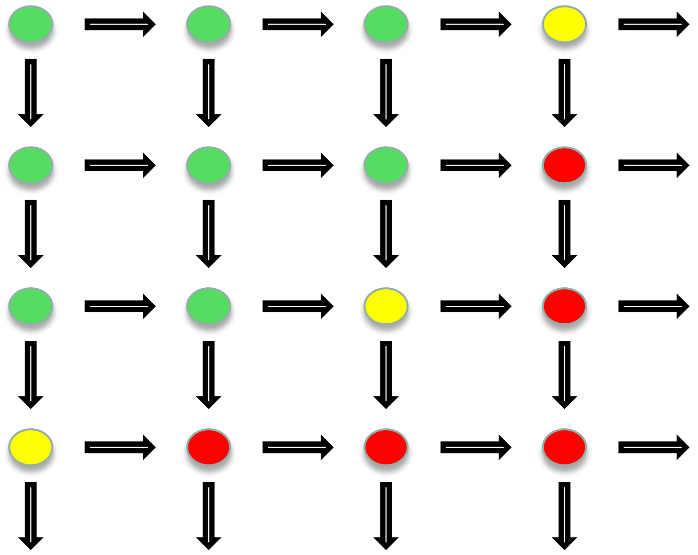
- **1st thread level : number of domains 1 to Ncores**



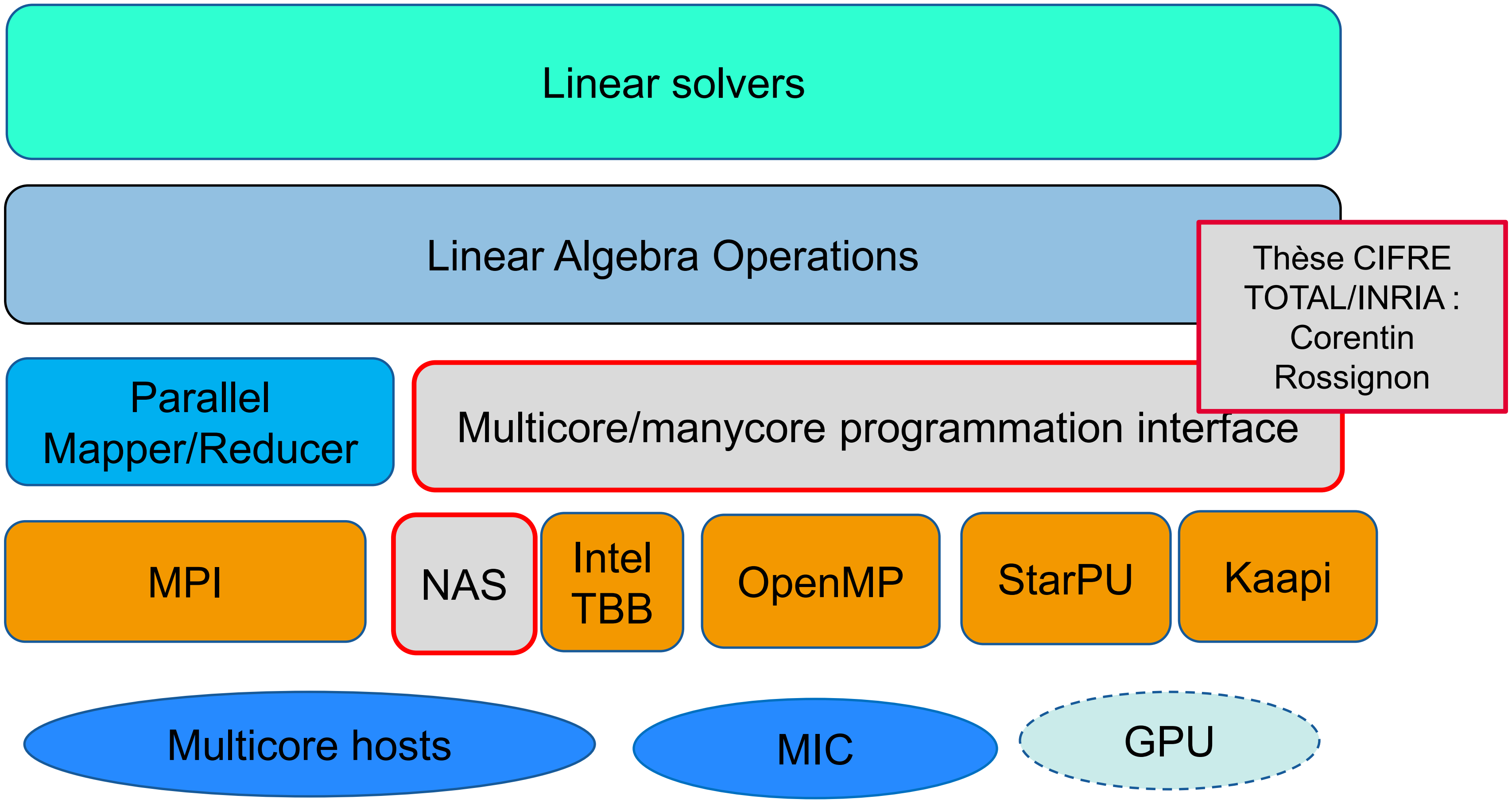
- **2nd thread level : number of tasks > 10000 : allow to load balance work between cores, use lower number of domains (better convergence)**

ILU : PARALLELIZATION USING A TASK PROGRAMMATION MODEL

- Program described as tasks is automatically parallelized : the number of tasks should be (much) greater than the number of cores
- Several strategies are possible for the task scheduler
- Example ILU(0) with natural ordering : wave front propagation

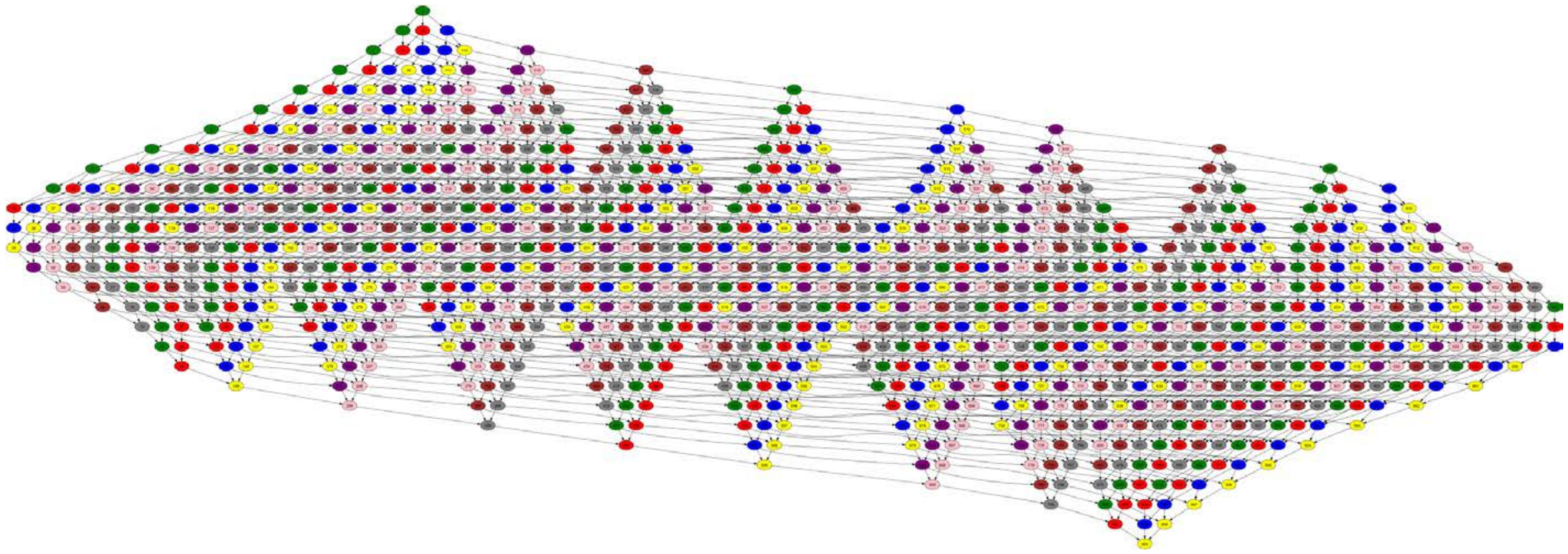


LINEAR ALGEBRA PACKAGE

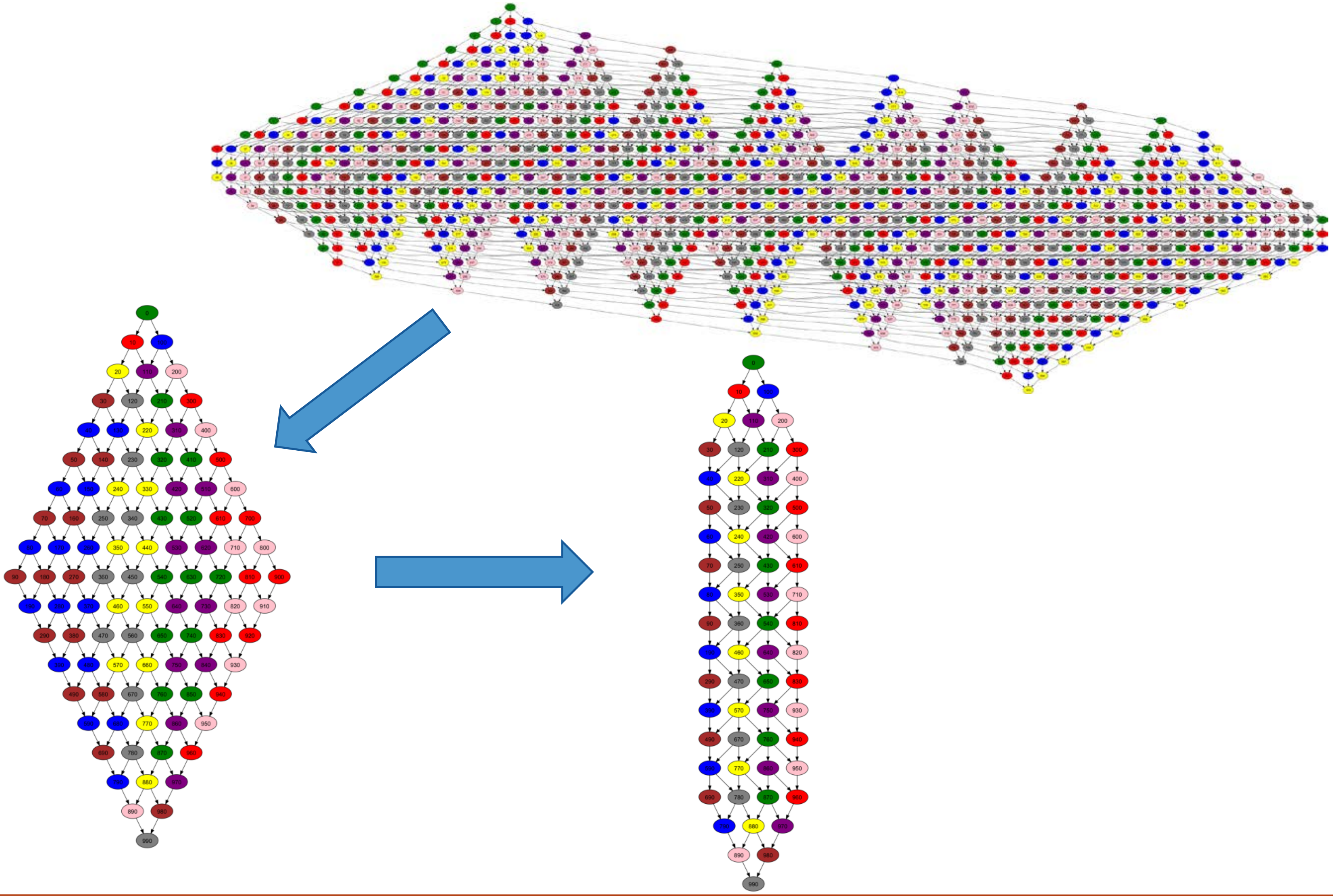

Linear solvers
Linear Algebra Operations
Thèse CIFRE TOTAL/INRIA : Corentin Rossignon
Parallel Mapper/Reducer
Multicore/manycore programmation interface
MPI
NAS
Intel TBB
OpenMP
StarPU
Kaapi
Multicore hosts
MIC
GPU

PARALLELISM GRAIN : CASE OF CUBE 10X10X10

Only 1000 tasks but



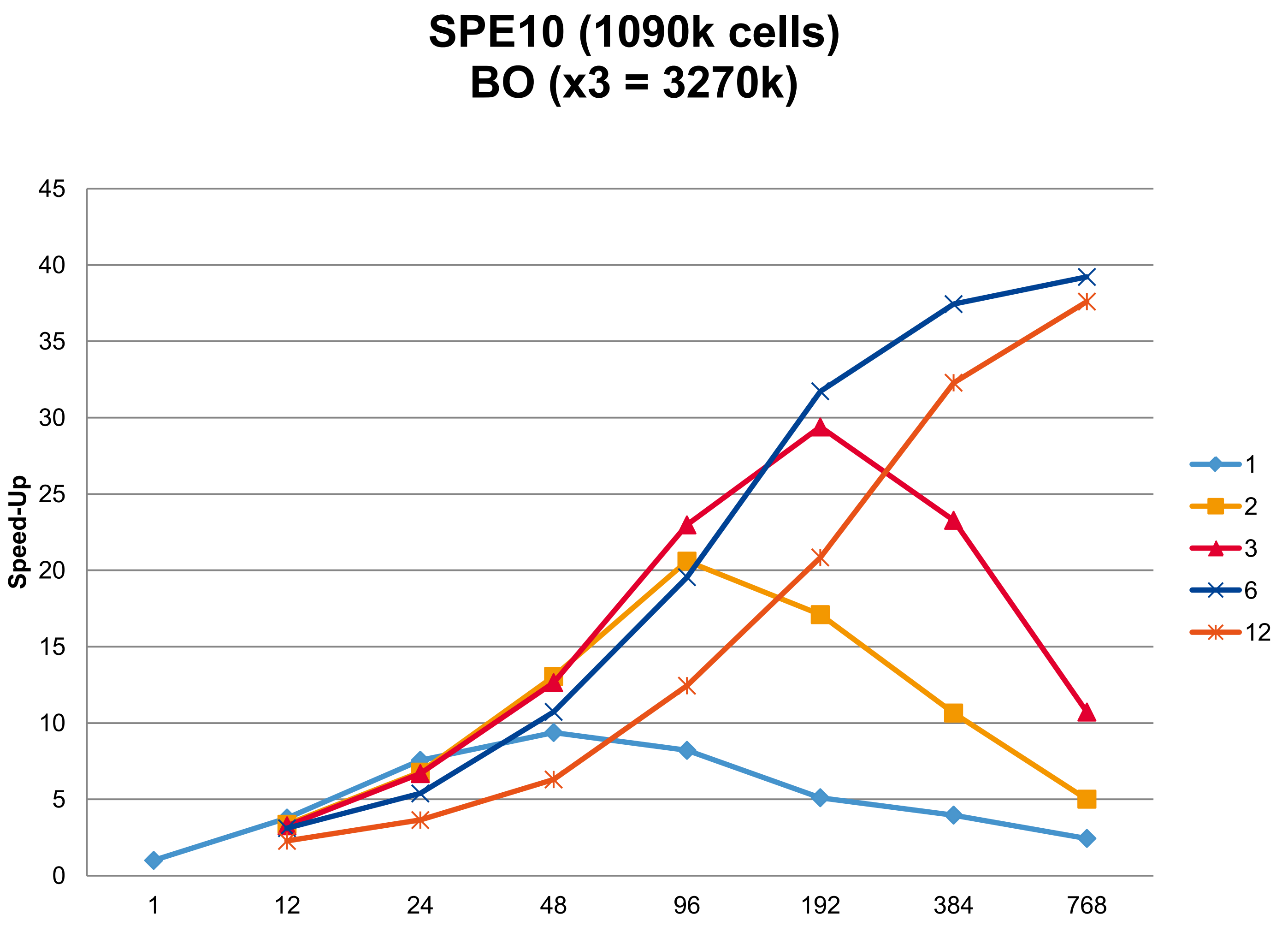
PARALLELISM GRAIN : SEVERAL PHASE



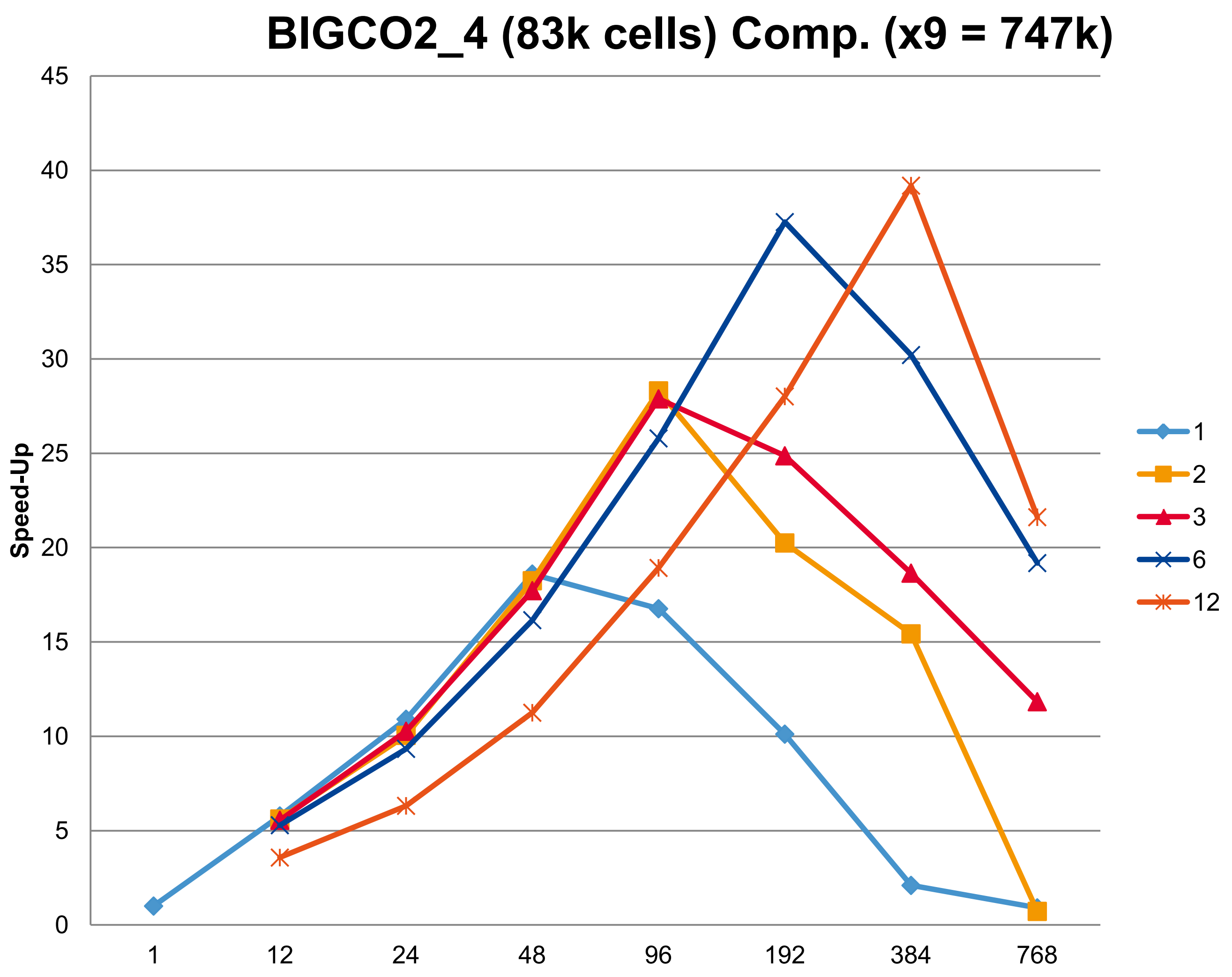
EXPERIMENTAL CONDITION

- Test on one rack SGI ICE 8200 (64 nodes of 12 cores : 768 cores)
- Socket X5660 Intel Xeon 6 cores @ 2.8 MHz (Westmere)
- 48 GB of memory
- Hyperthreading/SMT ON   BUT we use only 12 threads
- Turbo mode ON
- 2 Infiniband ConnectX DDR 4X (20GB/s)
- First stage of preconditioning : Hypre BoomerAMG (LLNC)
- Stopping Tolerance 0.001   (standard setting in our simulations)

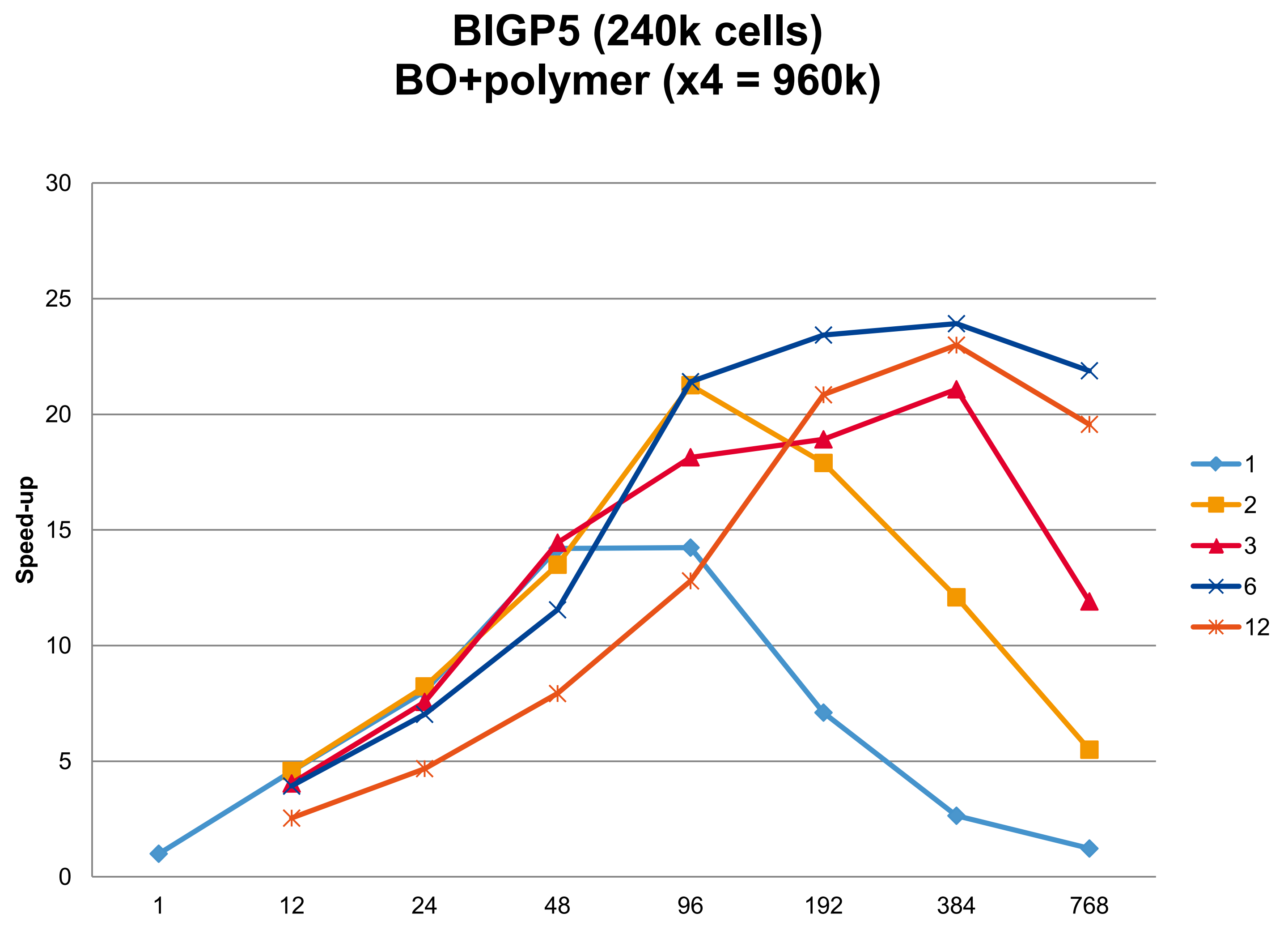
SPEED-UP VS NUMBER OF CORES



SPEED-UP VS NUMBER OF CORES



SPEED-UP VS NUMBER OF CORES



CONCLUSION

- MPI/Thread implementation allows a better « strong scalability »
- Fine grain parallelism will be more and more important for upcoming processors.
- MPI/thread implementation is important in term of programming efficiency but also in terms of numerical robustness (less domains)
- CPR solver involves global communications (dot products, gather on smaller grids etc..) : to reduce synchronizations due to the linear solver we are also investigating some methods at the non-linear level.