Fourth Brazil-France Workshop

On High Performance Computing and Scientific Data Management Driven by Highly Demanding Applications

Recent Advances in High Performance Computing for Multiphysics Problems

Alvaro Coutinho
alvaro@nacad.ufrj.br

High Performance Computing Center
COPPE/Federal University of Rio de Janeiro
www.nacad.ufrj.br

COPPE 50 UFRJ

15-18 September, Gramado, Brazil 2014

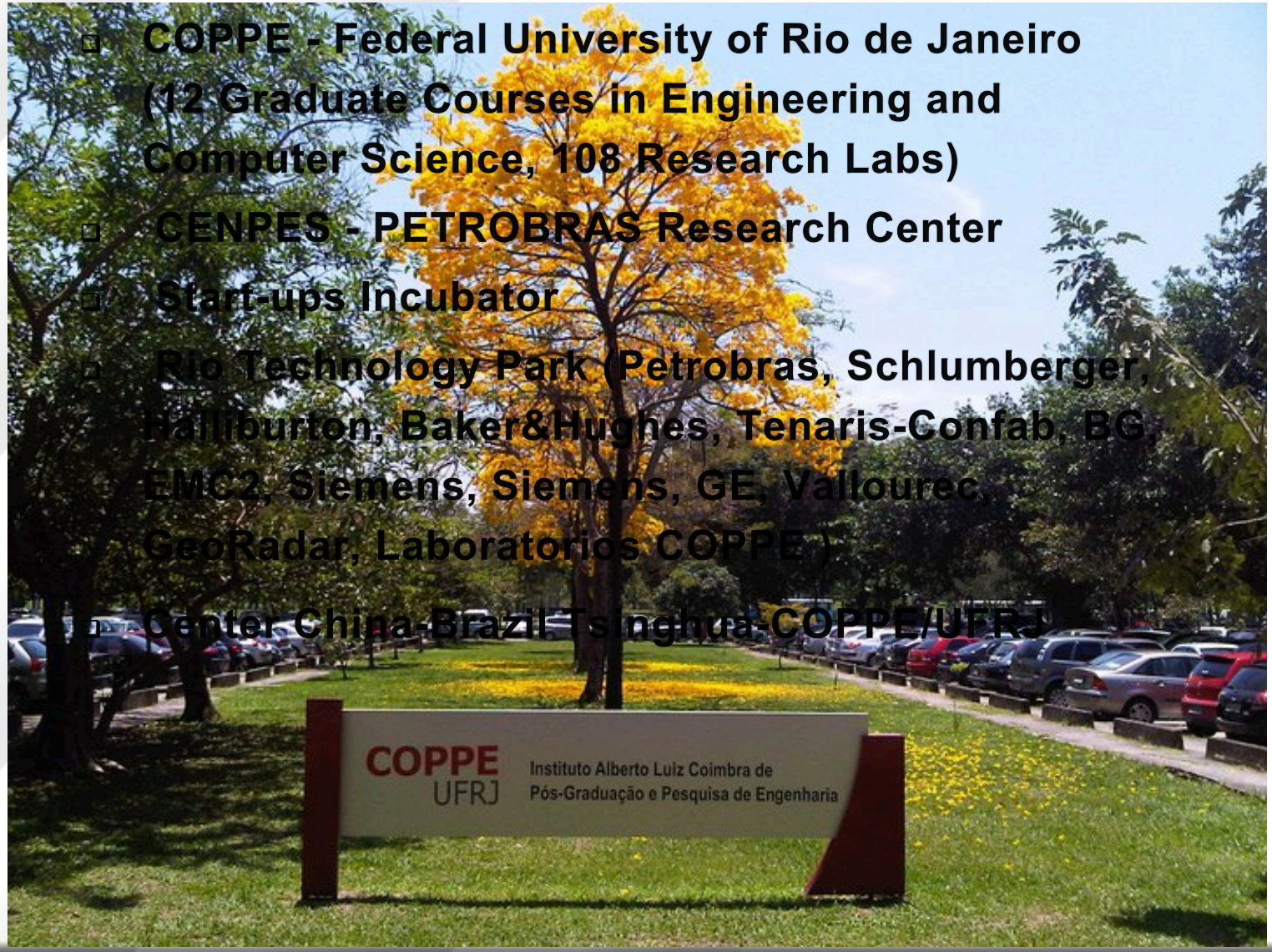
Contents

- **First things first: Who we are and what we do!**
- **A Few Words on Benchmarking**
- **Multiphysics and Multiscale**
  - Example Computations
  - Algorithms and Simulation Software
  - Adaptive Mesh Refinement and Coarsening
- **Pushing the limits**
  - Parallel Mesh Generation
  - Exploring the Stochastic Space
- **Final Remarks and Discussion**

First things first

WHO WE ARE AND WHAT WE DO

COPPE/UFRJ Innovation Ecosystem

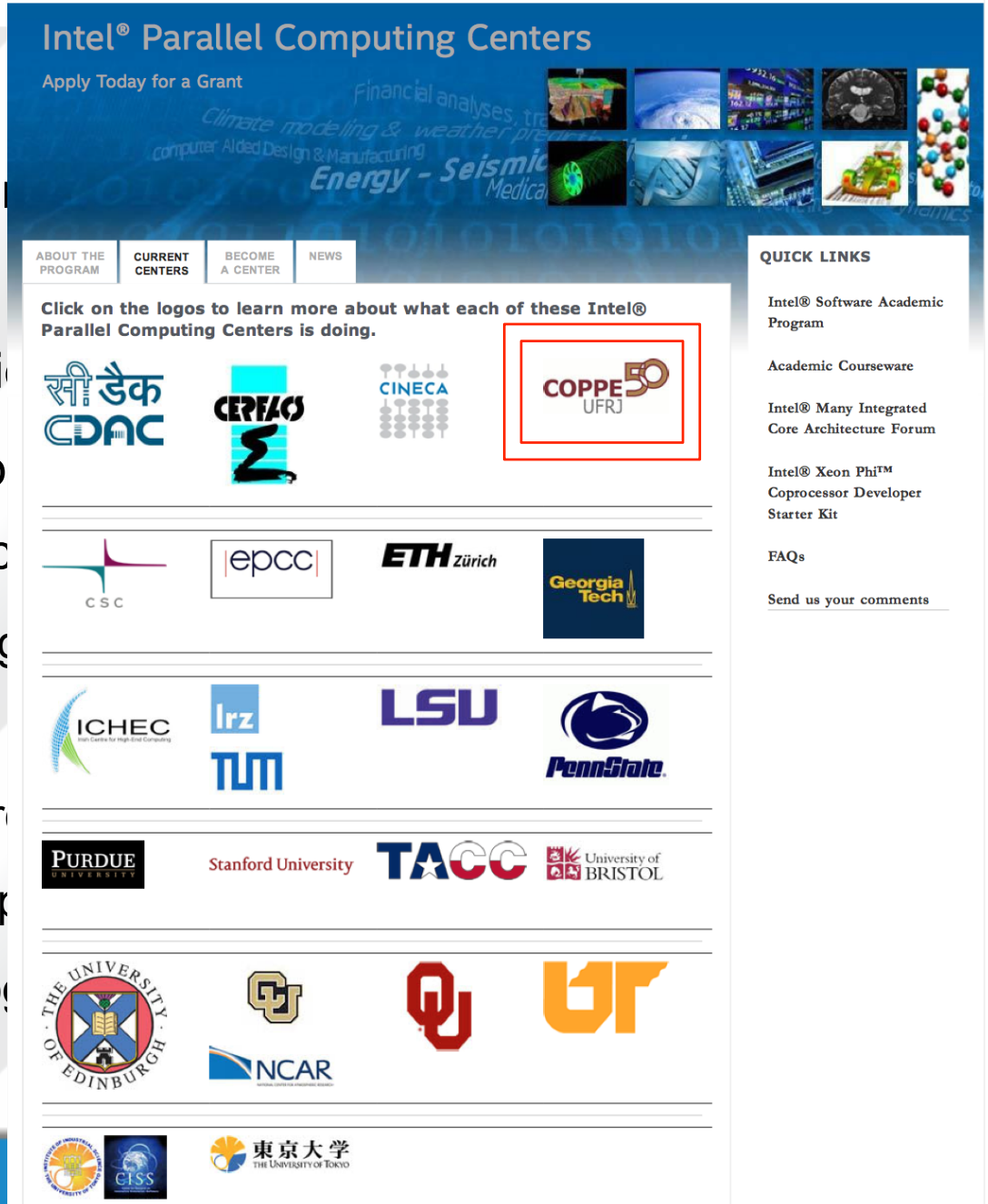


- COPPE - Federal University of Rio de Janeiro (12 Graduate Courses in Engineering and Computer Science, 108 Research Labs)
- CENPES - PETROBRAS Research Center
- Start-ups Incubator
- Rio Technology Park (Petrobras, Schlumberger, Halliburton, Baker&Hughes, Tenaris-Confab, BG, EMC2, Siemens, Siemens, GE, Vallourec, GeoRadar, Laboratorios COPPE)
- Center China-Brazil Tsinghua-COPPE/UFRJ

- High Pe mputer Science
- To provi omputing
- Develop of relevanc
  - Energ
  - Civil,
  - Envir
  - Comp
  - Biolo

Intel® Developer Zone:

Intel® Parallel Computing Centers



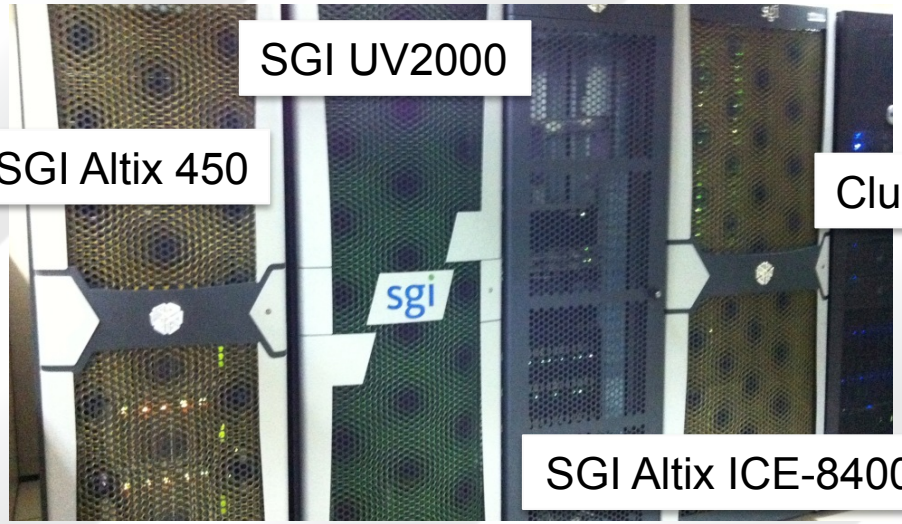
SINAPAD

HPC Center Systems

SGI UV2000

SGI Altix 450

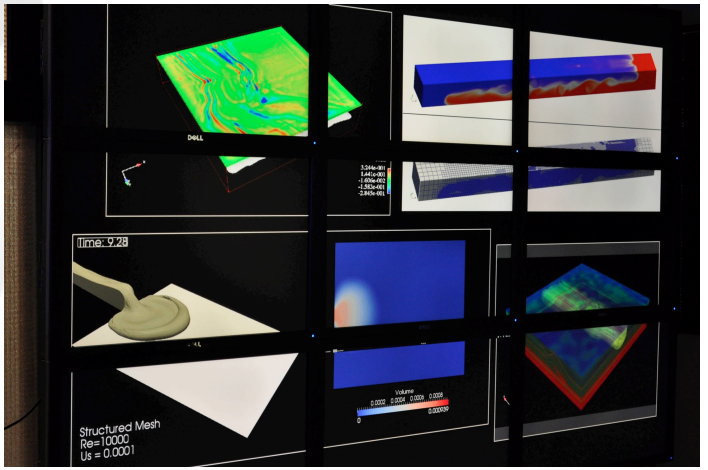
Cluster Dell



SGI Altix ICE-8400

NetApp Storage 100TB

Dell's Tiled Wall Display

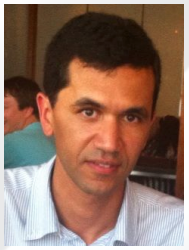


Galileu, Oracle Server #1 LA 2010
Memory 21TB, Storage 200TB

NACAD's Schedule on HOSCAR

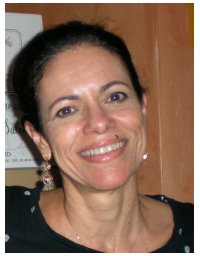
- **App Track**

  - Multiphysics (me)
  - Unstructured CFD Apps on Xeon Phi (Elias)



  - Regular Stencil Apps (Seismic) on Xeon Phi (Costa)

- **Scientific Workflows Track**



  - Iterations on UQ (Mattoso)

  - Gateways (Horta)
  - Adaptive Techniques (Silva)

A FEW WORDS ON BENCHMARKING

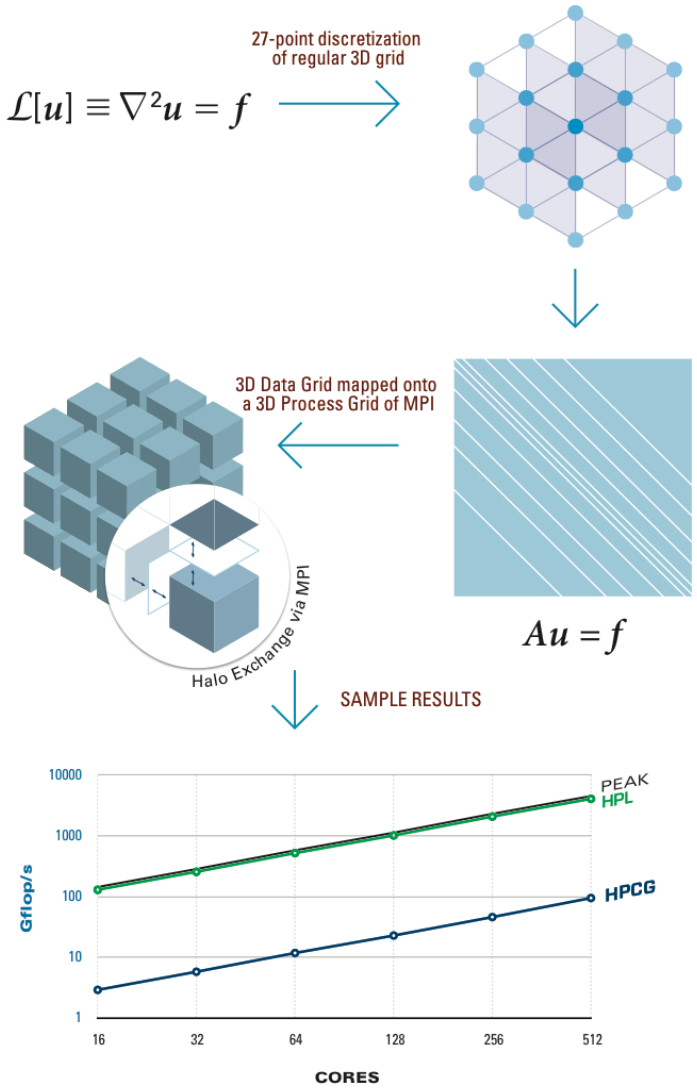
Brazil in TOP500 List[1], Jun 2014

Rank	Site	System	Cores	Rmax (TFlop/s)	Rpeak (TFlop/s)	Power (kW)
95	SENAI/CIMATEC Brazil	**CIMATEC01** - SGI ICE X, Intel Xeon E5-2690v2 10C 3GHz, Infiniband FDR SGI	17,200	405.4	412.8	
190	Petróleo Brasileiro S.A Brazil	**Grifo04** - Itautec Cluster, Xeon X5670 6C 2.930GHz, Infiniband QDR, NVIDIA 2050 Itautec	17,408	251.5	563.4	365.5
231	INPE (National Institute for Space Research) Brazil	**Tup** - Cray XE6, Opteron 6172 12C 2.10GHz, Cray Gemini interconnect Cray Inc.	31,104	214.2	261.3	
380	Petróleo Brasileiro S.A Brazil	**Grifo06** - Itautec Cluster, Xeon E5-2643 4C 3.300GHz, Infiniband FDR, NVIDIA 2075 Itautec	10,368	160.3	357.5	

Dongarra, Luszczek, Petitet, The LINPACK Benchmark: past, present and future, Concurrency Computat.: Pract. Exper. 2003; 15:803–820

[1]http://www.top500.org
Lists the top 500 supercomputers; Updated in 06/XX and 11/XX

The HPCG Benchmark[1]



We Need a New Yardstick

As a candidate for a new HPC metric, the HPC Preconditioned Conjugate Gradient Benchmark (HPCG) implements the preconditioned conjugate gradient method with a local symmetric Gauss-Seidel preconditioner.

In doing so, HPCG is designed to measure performance that is representative of many important calculations, like Type 1 patterns, with low computation-to-data-access ratios. To simulate patterns commonly found in real applications, HPCG exhibits the same irregular accesses to memory and fine-grain recursive computations that dominate so many scientific workloads.

In contrast to the new HPCG metric, the older HPL standard is a program that factors and solves a large dense system of linear equations using Gaussian Elimination with partial pivoting. The dominant calculations in this algorithm are dense matrix-matrix multiplication and related kernels, Type 2 patterns. With proper organization of the computation, data access is predominantly unit-stride and is mostly hidden by concurrently performing

HPCG on Galileu

Distributed Processes: 1400

Global Problem Dimensions:
nx: 1040 ny: 1040 nz: 1456

Number of Equations: 1,574,809,600

Number of Nonzero Terms: 42,445,920,184

GFLOP/s rating of: 234.165 ~50X

Total time: 234.165 s

60X

HPL
HPCG

Site	Computer	Cores	HPL Rmax (Pflops)	HPL Rank	HPCG (Pflops)
NSCC / Guangzhou	Tianhe-2 NUDT, Xeon 12C 2.2GHz + Intel Xeon Phi 57C + Custom	3,120,000	33.9	1	.580
RIKEN Advanced Inst for Comp Sci	K computer Fujitsu SPARC64 VIIIfx 8C + Custom	705,024	10.5	4	.427
DOE/OS Oak Ridge Nat Lab	Titan, Cray XK7 AMD 16C + Nvidia Kepler GPU 14C + Custom	560,640	17.6	2	.322
DOE/OS Argonne Nat Lab	Mira BlueGene/Q, Power BQC 16C 1.60GHz + Custom	786,432	8.59	5	.101#
Swiss CSCS	Piz Daint, Cray XC30, Xeon 8C + Nvidia Kepler 14C + Custom	115,984	6.27	6	.099
Leibniz Rechenzentrum	SuperMUC, Intel 8C + IB	147,456	2.90	12	.0833
CEA/TGCC-GENCI	Curie tine nodes Bullx B510 Intel Xeon 8C 2.7 GHz + IB	79,504	1.36	26	.0491
Exploration and Production Eni S.p.A.	HPC2, Intel Xeon 10C 2.8 GHz + Nvidia Kepler 14C + IB	62,640	3.00	11	.0489
DOE/OS L Berkeley Nat Lab	Edison Cray XC30, Intel Xeon 12C 2.4GHz + Custom	132,840	1.65	18	.0439 #
Texas Advanced Computing Center	Stampede, Dell Intel (8c) + Intel Xeon Phi (61c) + IB	78,848	.881*	7	.0161
Meteo France	Beaufix Bullx B710 Intel Xeon 12C 2.7 GHz + IB	24,192	.469 (.467*)	79	.0110
Meteo France	Prolix Bullx B710 Intel Xeon 2.7 GHz 12C + IB	23,760	.464 (.415*)	80	.00998
U of Toulouse	CALMIP Bullx DLC Intel Xeon 10C 2.8 GHz + IB	12,240	.255	184	.00725
Cambridge U	Wilkes, Intel Xeon 6C 2.6 GHz + Nvidia Kepler 14C + IB	3584	.240	201	.00385
TiTech	TUSBAME-KFC Intel Xeon 6C 1.2 GHz + IB	2720	.150	436	.00370

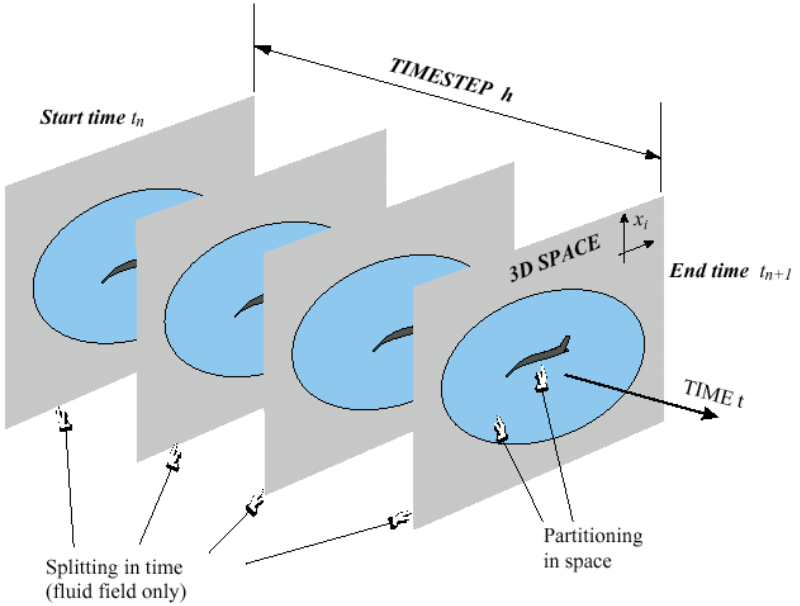
* scaled to reflect the same number of cores
# unoptimized implementation

[1]from: //software.sandia.gov/hpcg/

MULTIPHYSICS: WHAT IS IT?

Multiphysics Basics

- According Keyes et al (IJHPC Apps, 2013), a multiphysics system consists of more than one component governed by its own principle(s) for evolution or equilibrium, typically conservation or constitutive laws
- Multiphysics systems are analyzed by decomposition or breakdown. Coupling occurs in the bulk (e.g., through source terms or constitutive relations that are active in the overlapping domains of the individual components) or it occurs over an idealized interface that is lower dimensional.



From Felippa, Park and Farhat (CMAME, 2002)

Multiphysics Basics (cont'd)

- **Multiphysics problems are coupled systems characterized as 2-field, 3-field, etc.**
- **Coupling can be weak or strong, systems may have different time and spatial scales**
- **Fields are discretized in space and time. A field partition is a field-by-field decomposition of the space discretization. A splitting is a decomposition of the time discretization of a field within its time interval.**
- **Partitioning may be algebraic or differential.**
  - Algebraic partitioning; coupled system is spatially discretized and then decomposed.
  - Differential partitioning: decomposition is done first and each field is then discretized.

Examples of Coupled Multiphysics Problems

- Fluid-Structure Interaction (2-field)
- Thermal-Structure Interaction (2-field)
- Control-Structure Interaction (2-field)
- Solid-Fluid-Thermal Interaction (3-field)
- Fluid-Structure-Combustion (3-field)
- Chemo-Thermal-Structure (3-field)
- Fluid-Porous Media-Thermal (3-field)
- Fluid-Fluid Interaction (n-field)
- Fluid-Particle Interaction (1 field, n-particles)
- etc

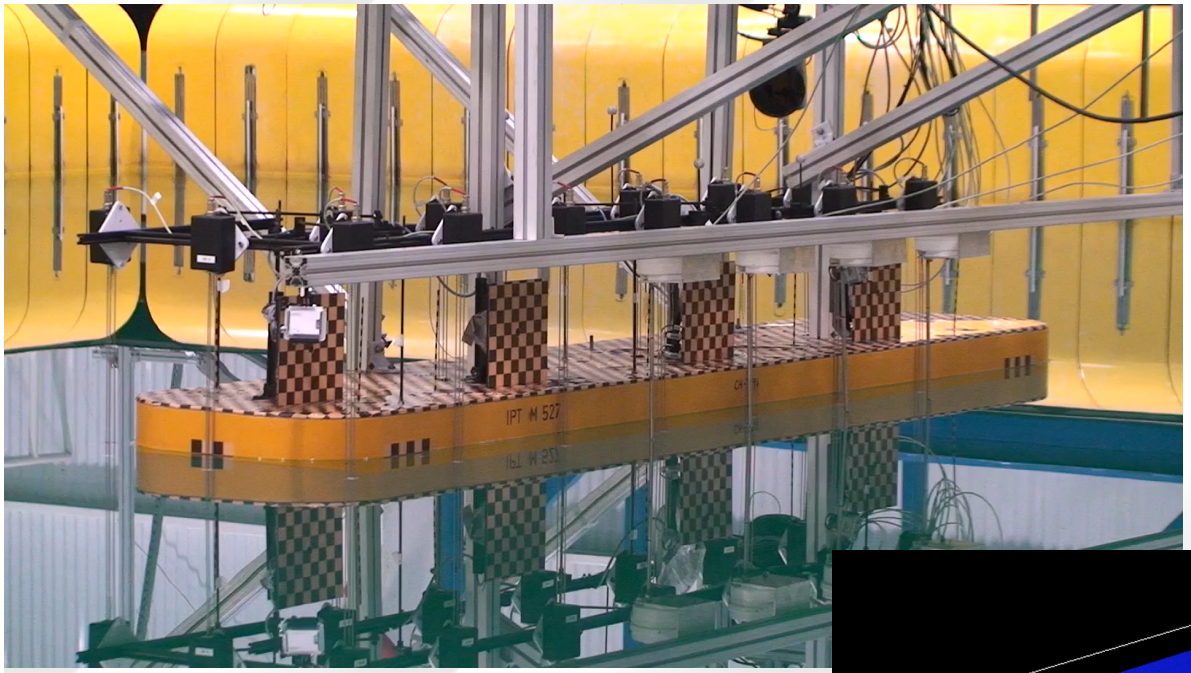
Wide ranges of interacting length and time scales are present, such as in turbulence, micromechanics, failure and damage, etc.

Strategies to Multiphysics Simulation

- **Monolithic or Simultaneous Treatment**
  - The whole problem is treated as a monolithic entity, advancing all fields simultaneously in time.
- **Partitioned Treatment**
  - Field models are computationally treated as isolated entities that are separately stepped in time.
  - Interaction effects are communicated between the individual components.
- **Monolithic and Partitioned Approaches are General**
  - We prefer the partitioned approach, because it allows the coupling of different programs, maybe written by different teams, etc.
  - However, we may see degradation of time-stepping stability in linear problems. Accuracy can be improved by iterating the state between fields. However, these iterations can be more costly than simply to reduce stepsize.

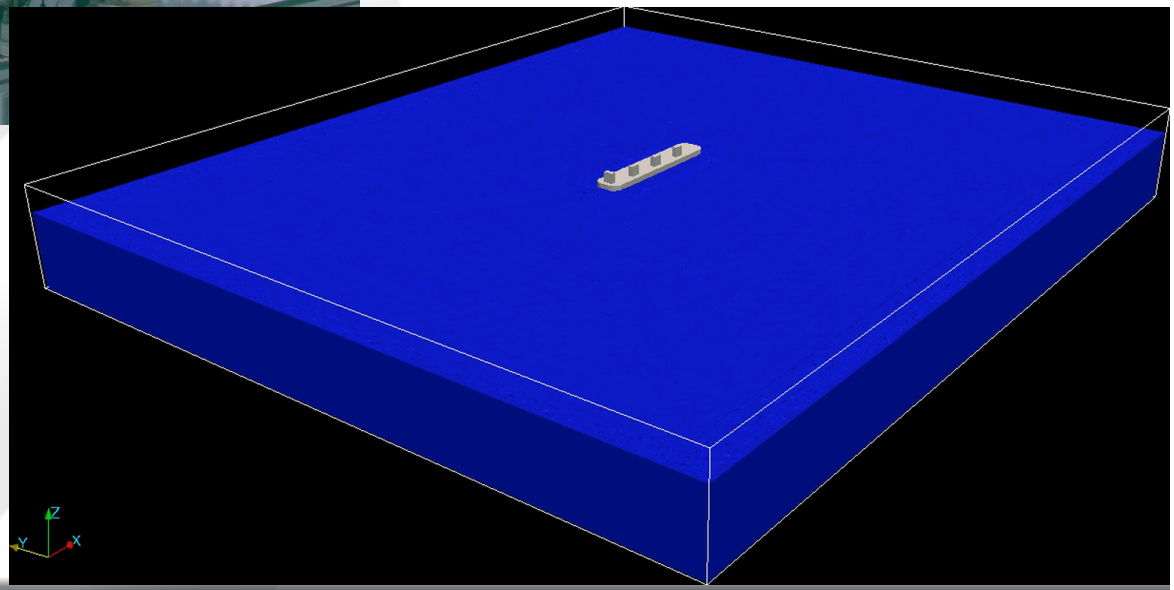
EXAMPLE COMPUTATIONS

Ship-Wave Interaction



Model test
TPN USP

Simulation

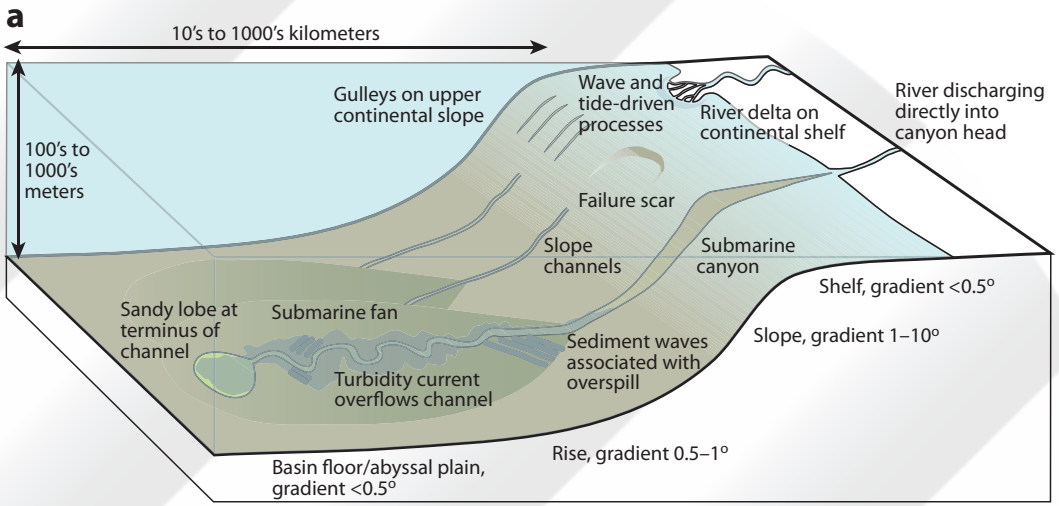


Green WATER

BR PETROBRAS

Turbidity Currents Simulation

PETROBRAS



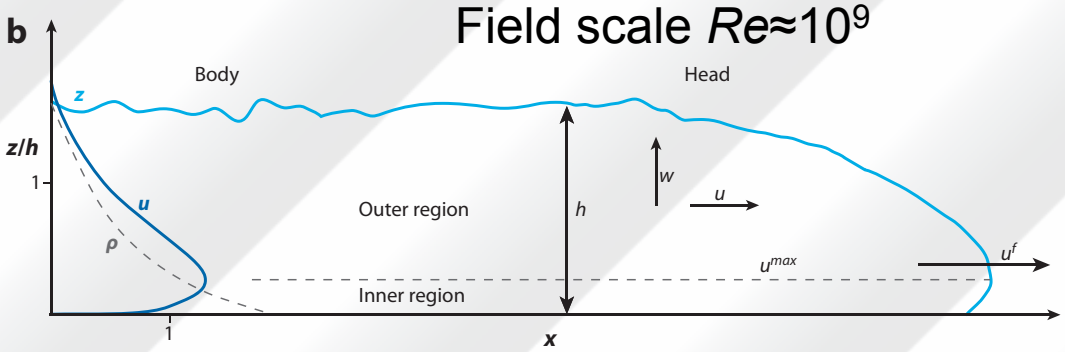
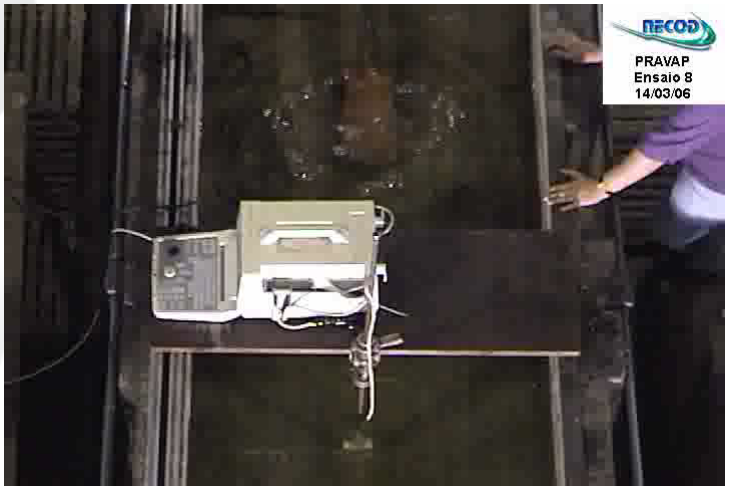


Figure 1

(*a*) Context of turbidity currents on the margins of continents and intracontinental basins, including deep lakes. (*b*) Schema of a turbidity current showing generalized velocity and density profiles based on integral length scale for current thickness, $h = \frac{\int_0^\infty u dz}{\bar{u}}$, where $\bar{u} = \frac{\int_0^\infty u^2 dz}{\int_0^\infty u dz}$.

Underwater turbidite flow



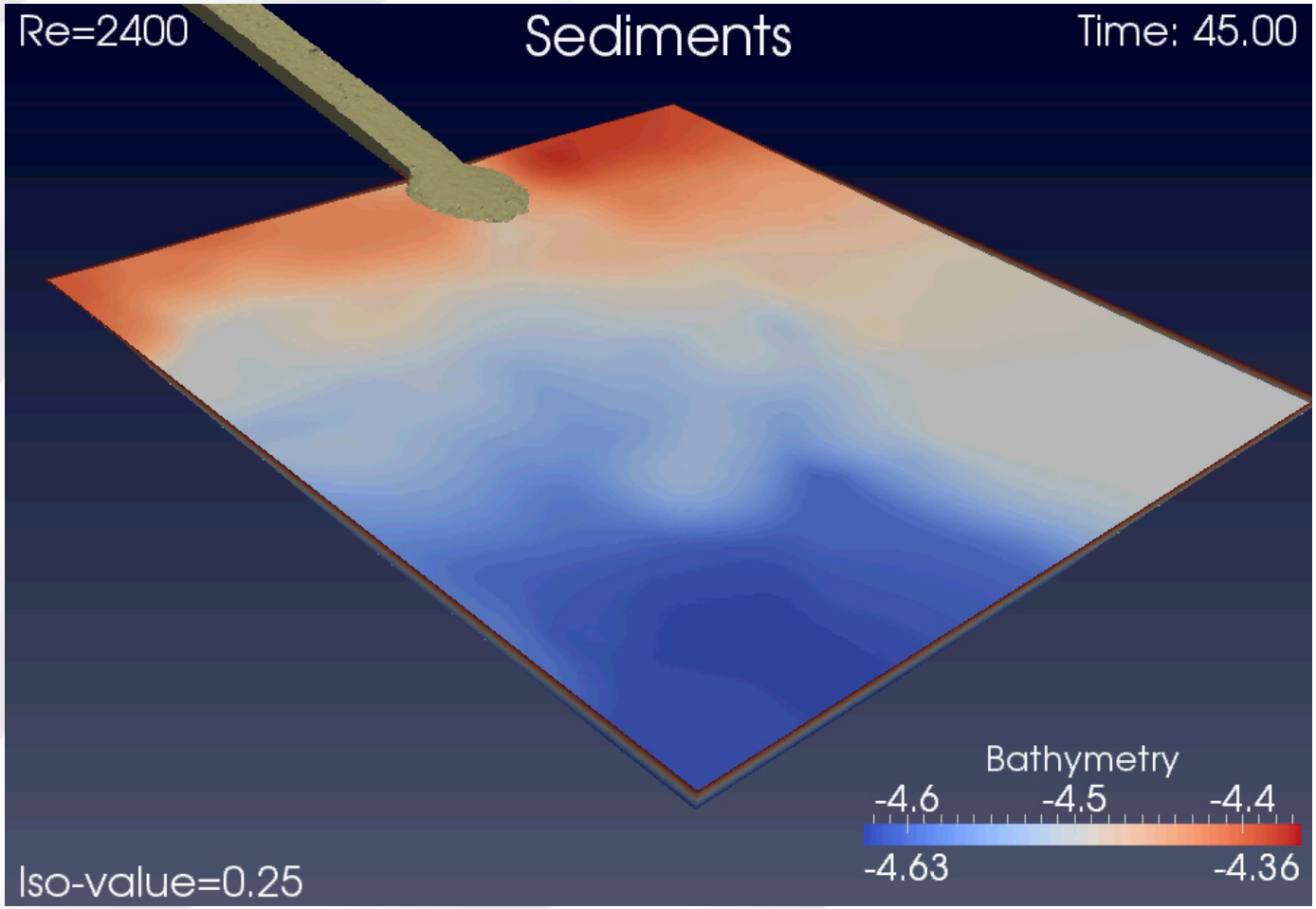
From: Meiburg and Kneller, Annu. Rev. Fluid Mech. 2010. 42:135–56

Turbidity Currents are a Multiphysics/Multiscale Process

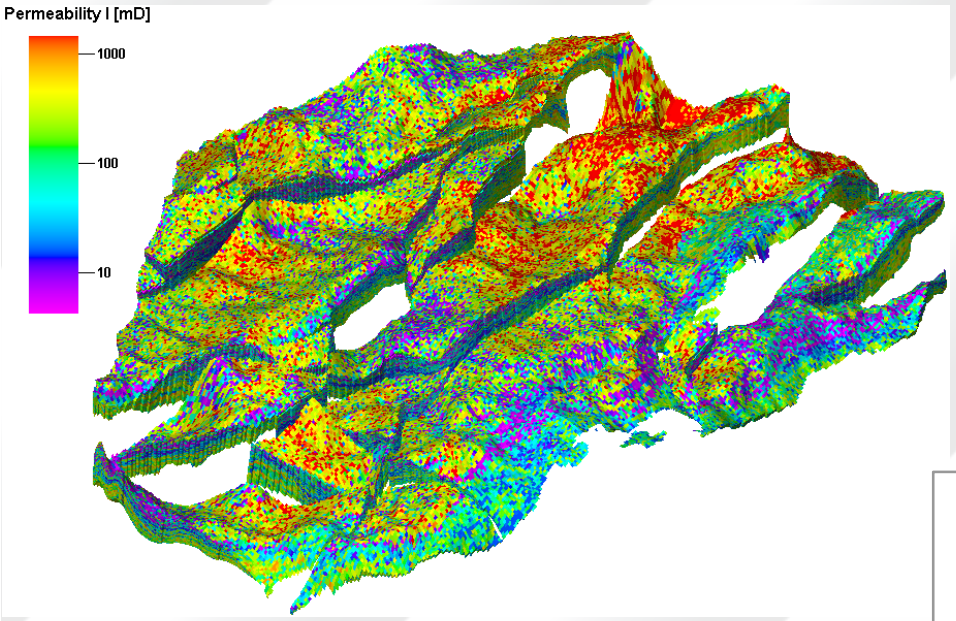
Name	Fluid Phase	Solid Phase	Coupling	Scale
Two-Fluid Model (TFM)	Eulerian	Eulerian	Polydisperse mixtures	Engineering (1 m)
Unresolved Discrete Particle Model (UDPM)	Eulerian (unresolved)	Lagrangian	Fluid-Particle drag closures	Laboratory (10^{-1} m)
Resolved Discrete Particle Model (RDPM)	Eulerian (resolved)	Lagrangian	Boundary conditions at particle surface	Laboratory (10^{-2} m)
Molecular Dynamics (MD)	Lagrangian	Lagrangian	Elastic collisions at particle surface	Mesoscopic ($< 10^{-3}$ m)

Adapted from: Meiburg and Kneller, , Annu. Rev. Fluid Mech. 2010. 42:135–56 and M.A. van der Hoef, et al Annu. Rev. Fluid Mech. 2008. 40:47–70

Real Test Case on a Paleobathymetry

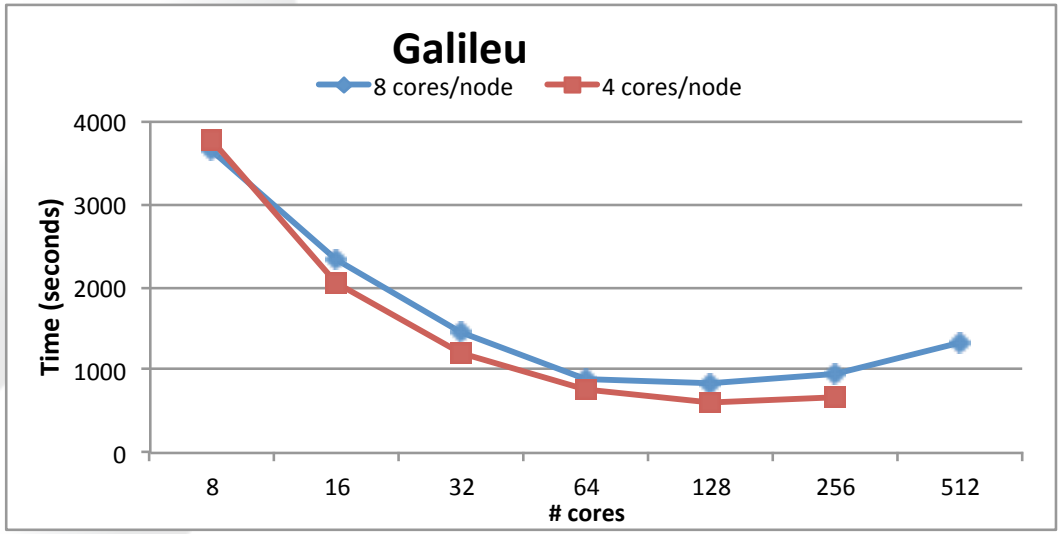


Profiling Reservoir Simulators



Horizontal permeability distribution in B2

1. SLB Iintersect (IX) Multiphase flow in porous media
2. B2 Benchmark
   - ❑ Compositional model
   - ❑ Realistic reservoir
   - ❑ 2 million cell
3. IX Numerics
   - ❑ Multistage parallel linear solver framework
   - ❑ Two-stage CPR1 (Constraint Pressure Residual) scheme for large-scale parallel runs
   - ❑ Parallel Algebraic Multigrid solver with a F-GMRES outer iteration



Schlumberger

ALGORITHMS AND SOFTWARE

What is large?

- **Large in size ➔ unstructured grids with 10^8-10^9 elements (cells)**
- **Large in coupling ➔ many fields interacting**
- **Large in physical parameters ➔ different viscosities, densities, …**
- **Large in control parameters ➔ tolerances, solver options, etc**
- **Large in complexity ➔ several softwares, human intervention, reproducibility, uncertainty quantification, data provenance.**
- **What do we have to worry about besides multiphysics itself?**
  - Efficient algorithms and solvers
  - Data structures, data storage and management
  - Parallel mesh generation and adaptivity
  - Visualization

Challenges on Multiphysics Algorithms and Software* (1)

- **Modern simulation software is complex:**
  - Implicit numerical methods
  - Massively parallel computers
  - Adaptive methods
  - Multiple, coupled physical processes
- **There are a host of existing software libraries that excel at treating various aspects of this complexity.**
- **Leveraging existing software whenever possible is the most efficient way to manage this complexity.**

* Adapted from libMesh library: libmesh.sourceforge.net

Challenges on Multiphysics Algorithms and Software (2)

- **Modern simulation software is multidisciplinary:**
  - Physical Sciences
  - Engineering
  - Computer Science
  - Applied Mathematics
  - Etc …
- **It is not reasonable to expect a single person to have all the necessary skills for developing & implementing high-performance numerical algorithms on modern computing architectures.**
- **Teaming is a prerequisite for success.**

Single Simulation Code: Major Components

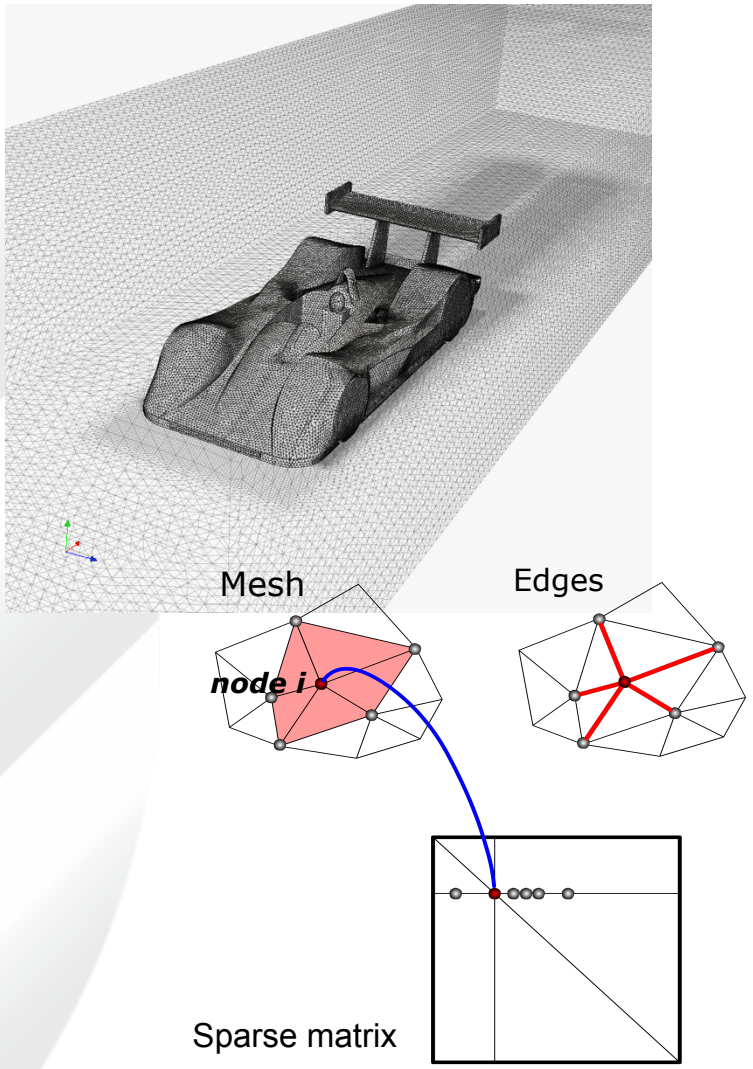
```
Pre-processing
Input data
        Time integration loop
            Nonlinear iteration loop
                  Form system of linear equations
                  Solve system of linear equations
            End NL loop
            Update time step
            Output results
         End time loop
Post-processing
Visualization
```

Complexity $O(n^{4/3})$, n #unknowns

Optimal solvers require O(n) work per time step, and time accurate integration often implies $O(n^{1/3})$ time steps.

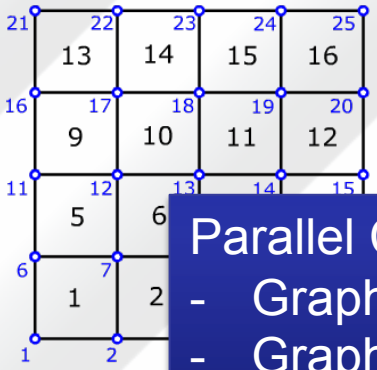
Finite Element Method

- **Unstructured grid method characterized by:**
  - Discontinuous data
  - Gather/scatter operations
  - Random memory access
  - Data dependencies
- **Main Computational Kernels for Implicit Time Marching**
  - Forming system matrix and RHS
  - Solving linearized systems by preconditioned Krylov solvers

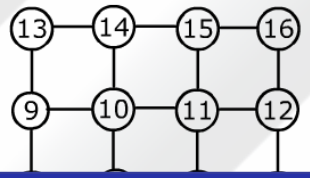


Meshes, Graphs and Matrices

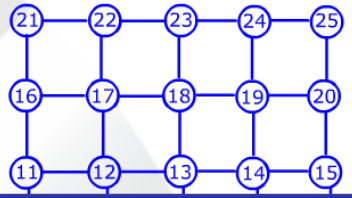
Finite difference grid



Dual graph (cells)

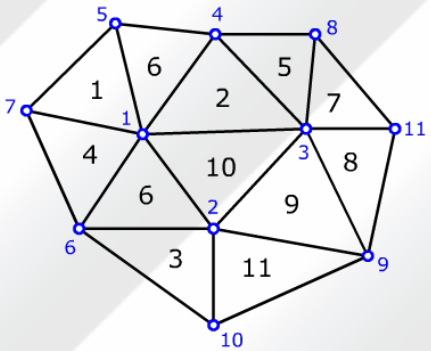


Grid points graph

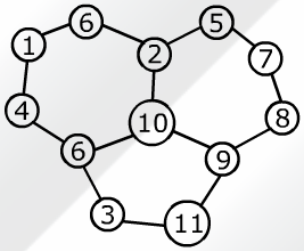


Parallel Computing:
- Graph partitioning, that is, minimizing edge cuts – MPI runs
- Graph coloring: threads

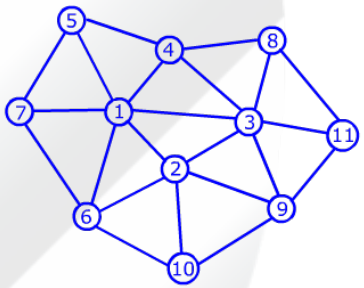
Unstructured mesh



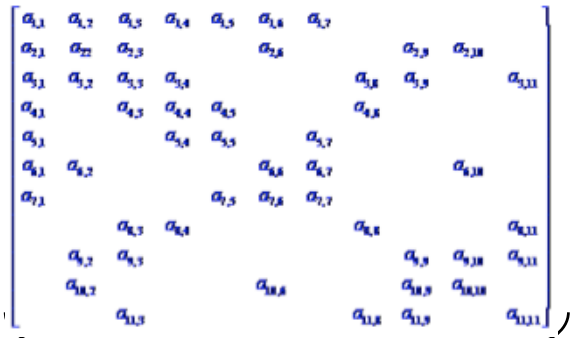
Dual graph (elements)



Nodal graph



Sparse Matrix



geometrical representation

relational representation

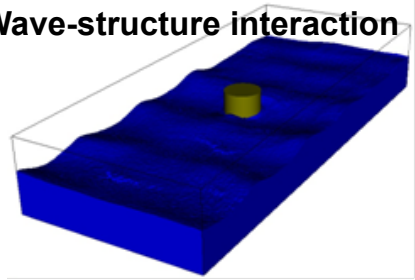
algebraic representation

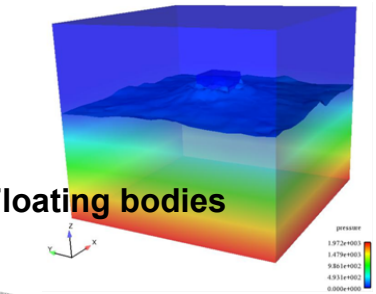
SIMULATION SOFTWARE

EdgeCFD®
Fluid Flow/Free-surface/FSI Solver

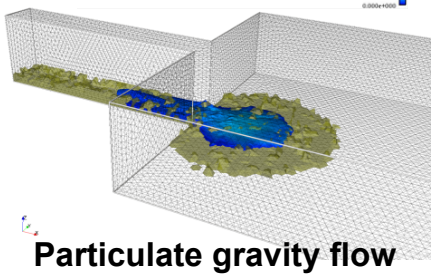
- **General:**
  - Edge based data structure. *"EDE has been proving to be more efficient than other FEM data structures like CSR or EBE"*
  - Segregated predictor-multicorrector time marching;
  - Adaptive time stepping with PID controller;
  - Supports hybrid parallelism (MPI, OpenMP or both at the same time);
  - Unstructured grids with linear tetrahedra for velocity, pressure and scalar transport;
  - Mesh partitioning performed by Metis or ParMetis;
  - Best data reordering defined by EdgePack® in a preprocessing phase;
  - Thermal-flow coupling with Boussinesq approximation; FSI
  - Input/Output file formats: ANSYS/Ensight/Paraview, neutral files, Xdmf/hdf5
- **Incompressible Flow:**
  - SUPG/PSPG/LSIC stabilized finite element method in Eulerian or ALE frames
  - Fully coupled **u**-*p* system (4-dofs per node/non-symmetric);
  - Inexact Newton-GMRES;
  - LES (Smagorinsky, Dynamic Smagorinsky), ILES, RB-VMS
  - Newtonian or non-Newtonian flows (Power Law, Bingham and Hershel-Buckley)
- **Transport:**
  - SUPG/CAU/YZBeta stabilized finite element method in Eulerian and ALE frames
  - Supports free-surface flows through Volume-Of-Fluid and Level-Sets.
  - (UFMM) Unstructured Fast Marching Method for fast computation of signed distance functions
  - PDD: Parallel dynamic deactivation. "*Restrict the computation only in regions with high solution gradients*"

Wave-structure interaction





Floating bodies



Particulate gravity flow

EdgeCFD Software Stack

- **Ansys Classic, ICEM-CFD, CFX and/or GMSH**
  - Computational model
  - Mesh Generation
- **Preprocessor (EdgeCFDPre)**
  1. Takes a serial mesh;
  2. Creates partitions with Metis (could be Scotch…)
  3. Extracts edges and reorders data with EdgePack
  4. Stores data prepared to solver
- **Solver (EdgeCFDSolver)**
- **ParaView, VisIt, Ensight**
  - Visualization: Ensight, Xdmf/HDF5 or Parallel VTK

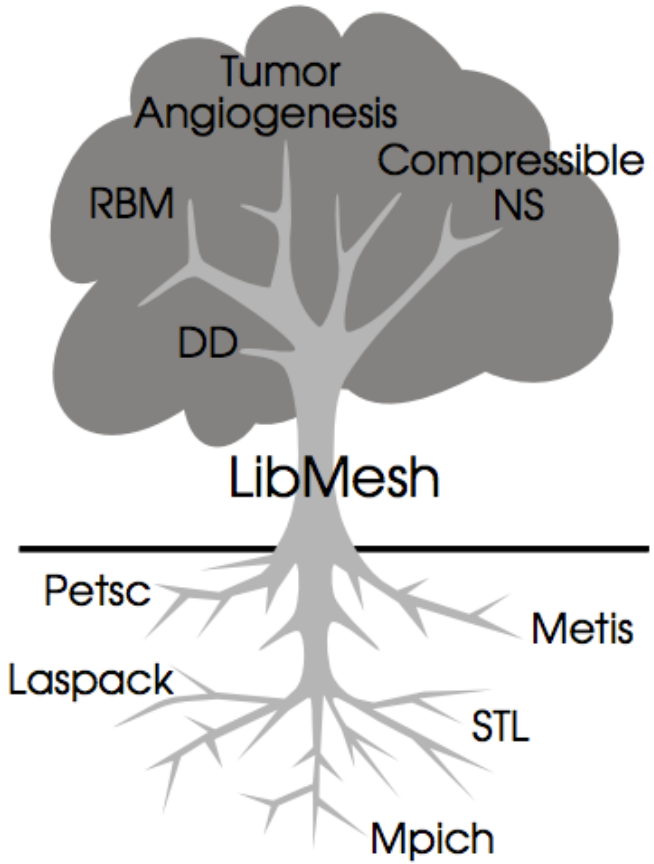
***Blue**: "Home made" code*
***Green**: Third party code*

*Workflow management and provenance by **Chiron***

libMesh

- High level interface for finite element analysis (Kirk et al., Eng. with Computers, 2006)
- Keep focus on the physical problem instead of the computational aspects related to the adaptive mesh refinement/coarsening and parallel computing
- Developed initially at UT Austin
- Available at:

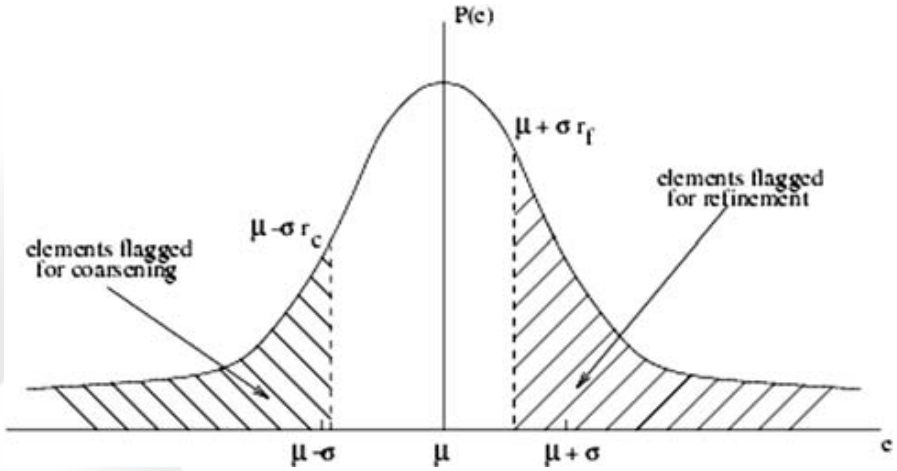
http://libmesh.sourceforge.net/



libMesh interfaces

AMR/C

- libMesh utilizes a statistical scheme based on Kelly's error estimator
- As the simulation goes on, the statistical distribution of the error spreads and then the refinement and coarsening begin
- As the solution reaches equilibrium, the error distribution reaches steady state and then the adaptive process stops



Probability density function. Kirk et al., 2006

Nonlinear solvers and adaptive time step control

SMART ALGORITHMS

Advanced Nonlinear Solver

Algorithm 2: Inexact Krylov-Newton Backtracking Method - INKB

Set η_0;
$k = 0$;
$\tau_{NL} = \tau_{res} \|\mathbf{F}(\mathbf{x}^k)\|_2$;
while $\|\mathbf{F}(\mathbf{x}^k)\|_2 > \tau_{NL}$ **do**
    Compute $\mathbf{J}(\mathbf{x}^k)$;
    Solve $\mathbf{J}(\mathbf{x}^k)\mathbf{s} = -\mathbf{F}(\mathbf{x})$ by a Krylov method with tolerance η_k ;
    Set $\lambda_1 = 1$;
    Compute $\mathbf{x}^{k+1} = \mathbf{x}^k + \lambda_1 \mathbf{s}$;
    $i = 0$;
    $\mathbf{x}^t = \mathbf{x}^{k+1}$;
    while $\|\mathbf{F}(\mathbf{x}^t)\|_2 > (1 - \alpha\lambda_i)\|\mathbf{F}(\mathbf{x}^{k-1})\|_2$ **and** $i \leq nbt$ **do**
        Choose λ_{i+1};
        Update $\mathbf{x}^t = \mathbf{x}^k + \lambda_{i+1}\mathbf{s}$;
        $i = i + 1$;
    endw
    if $i < nbt$ **then**
        Update $\mathbf{x}^{k+1} = \mathbf{x}^t$;
    else
        backtracking loop rejected ;
    endif
    $k = k + 1$;
    Select η_k;
endw

Adaptive Time Step Control

- **How to speed-up time marching schemes?**
  - Adapting time step according to the solution evolution;
  - Track the solution with interventions (feedback) when needed.
  - Particularly useful in stiff systems
- **Key idea:**
  - Tracking × Feedback → Controllers
  - Theory on ODE solvers
    - *G. Soderlind, Automatic control and adaptive time-stepping. Numerical Algorithms, 2002; 31:281–310*
  - Application on PDE's:
    - *A. M. P. Valli, G. F. Carey and A. L. G. A. Coutinho, Control strategies for timestep selection in finite element simulation of incompressible flows and coupled reaction–convection–diffusion processes, IJNMF, 2005.*

Coupled Fluid Flow and Heat Transfer
Rayleigh-Benard 4:1:1 Container

MESH

Eleme

Nodes

Edges

Flow

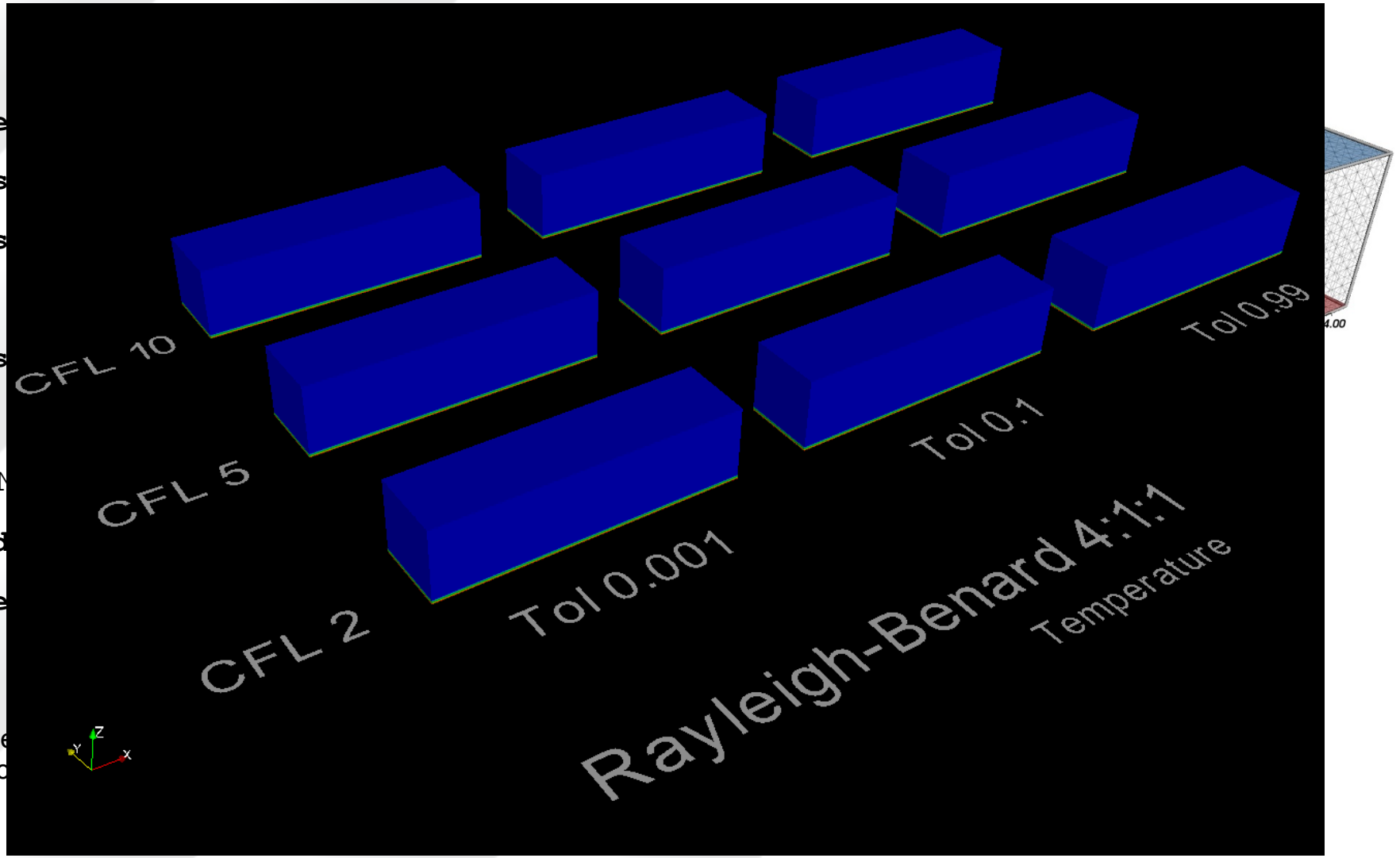
Trans

DIMEN

Prand

Rayle

Se
co

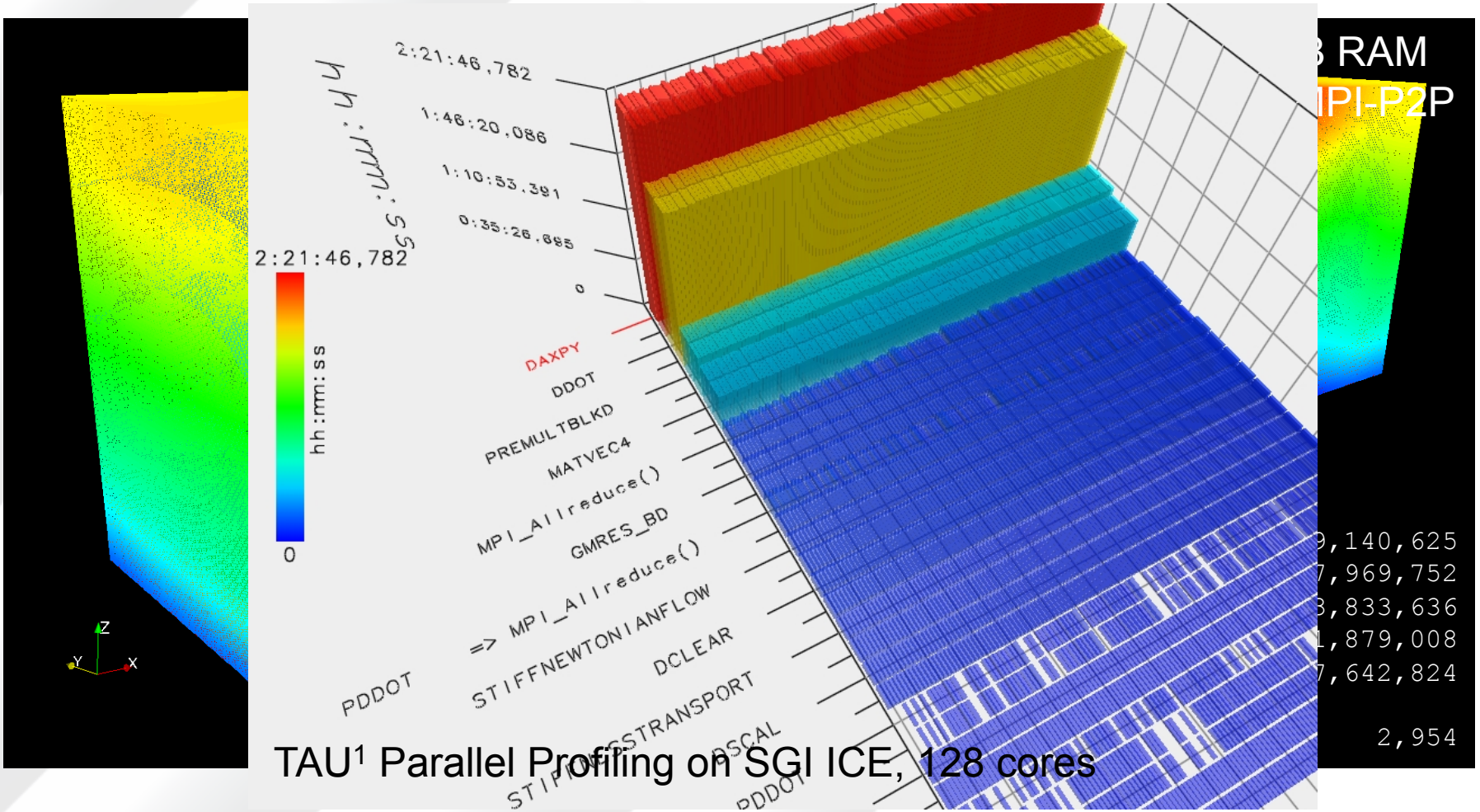


Adaptive Time Stepping and IN Performance

Max IN Tol	CFL min	CFL max	SS time	time steps	wall time
0.001	2	10	0.5686	109	1110.49
		5	0.2410	90	855.42
		2	0.1829	171	1184.81
0.1	2	10	0.5635	108	803.89
		5	0.2418	90	637.71
		2	0.1840	172	1017.02
0.99	2	10	0.7436	142	1652.06
		5	0.2417	90	991.96
		2	0.1839	172	1501.97

SS Tolerance: 10^{-5} Transport P-GMRES Tol: 10^{-3} # Krylov Space Vectors: 25

Rayleigh-Benard 4:1:1 - 501×125×125 mesh

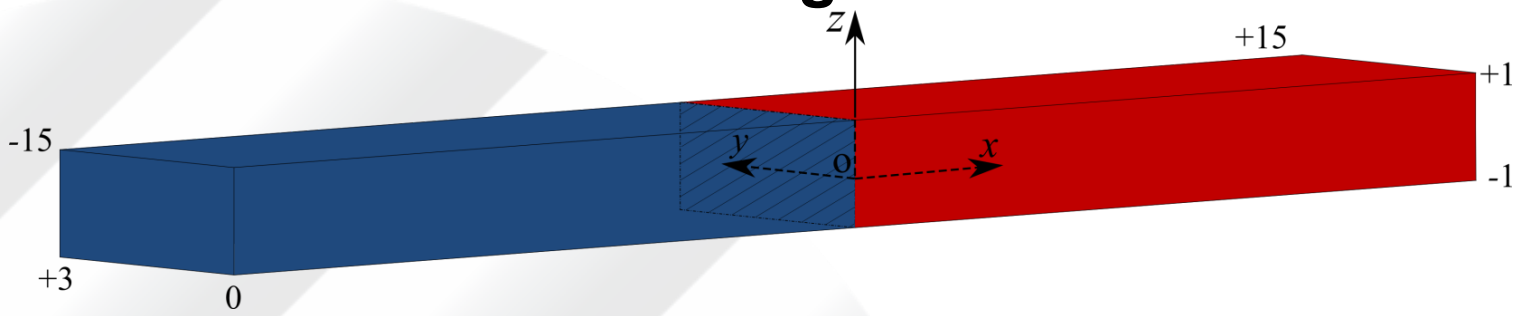


TAU[1] Parallel Profiling on SGI ICE, 128 cores

[1]http://www.cs.uoregon.edu/research/tau

AMR/C COMPUTATIONS

3D Lock-Exchange with AMR/C



Geometry, boundary and initial conditions:

- ✓ Simulation Domain: $\Omega = [-15,15]\text{x}[0,3]\text{x}[-1,1]$
- ✓ No-slip on top and bottom walls; slip on all others
- ✓ Half right filled with “heavy” fluid $(\rho_h = 1)$ and left half filled with “light” fluid $(\rho_l = 0)$

Dimensionless parameters:

$\text{Gr} = 1.5 \times 10^6 \qquad Sc = 0.71$

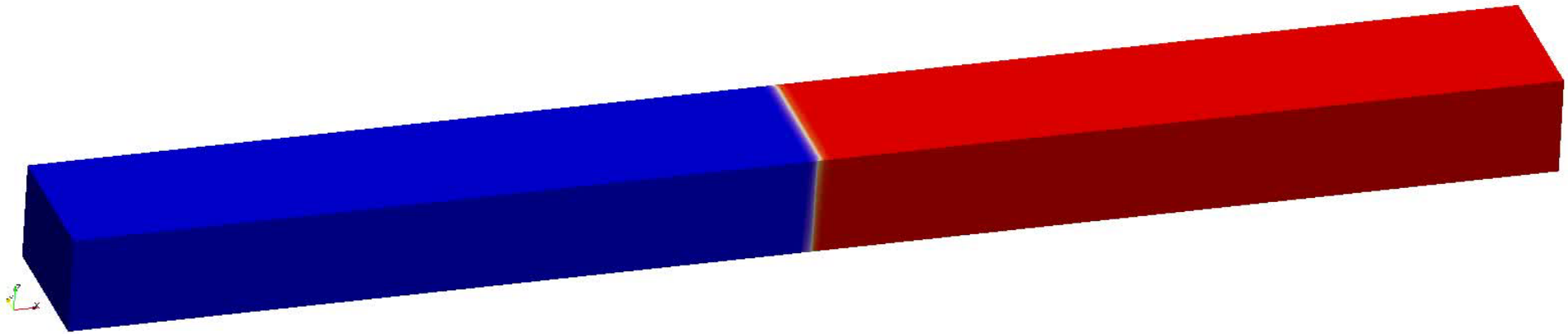
$\Delta t = 0.025$

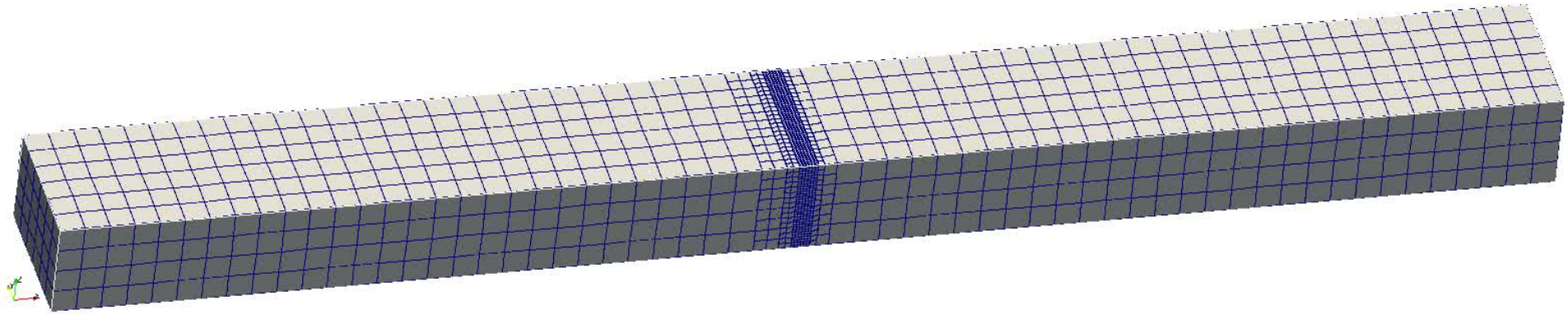
Run on 240 cores, Viz with ParaView, parallel rendering

See also Camata et al, IJNMF, 2012

GMRES(30)
BILU(1)
RCM reordering

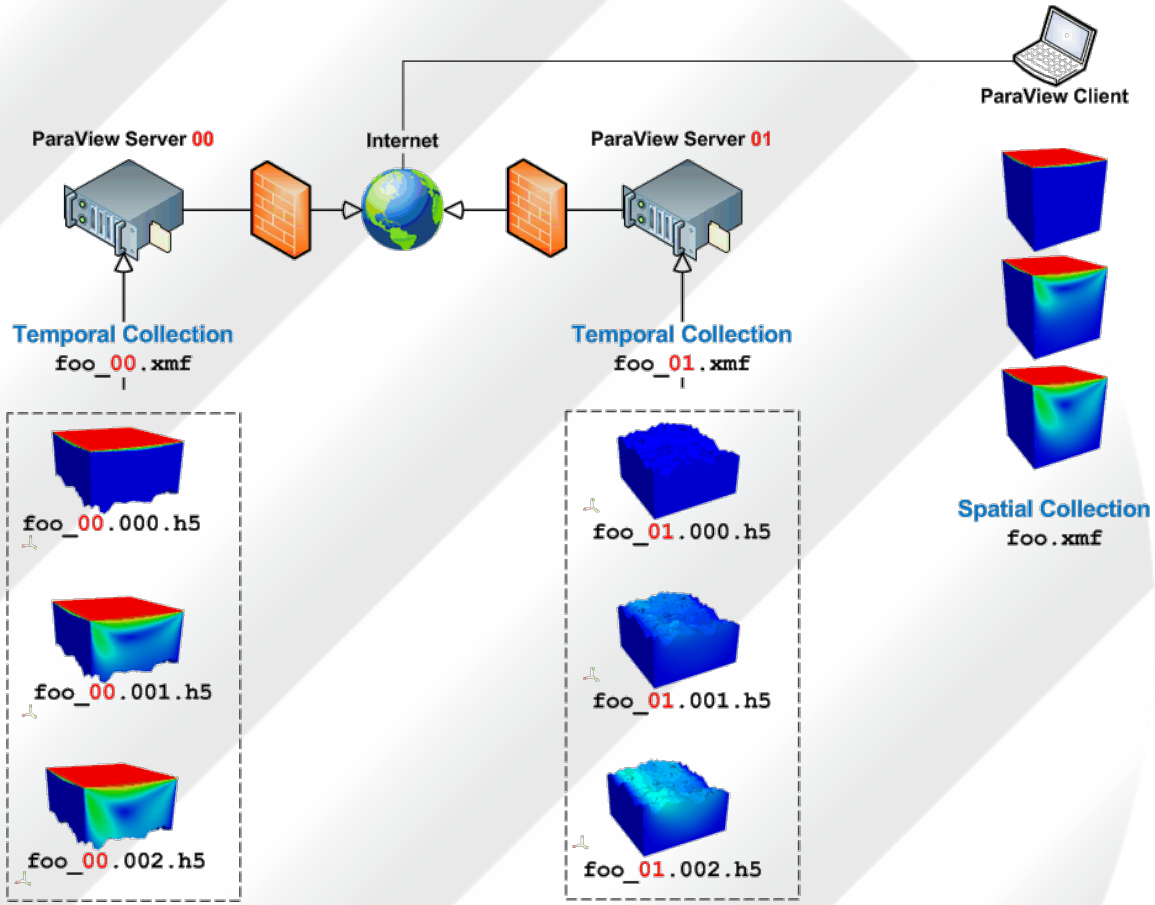
Planar 3D Lock-Exchange with AMR/C





Rossa & Coutinho, Parallel adaptive simulation of gravity currents on the lock-exchange problem, Computer and Fluids, 2013

Parallel Visualization

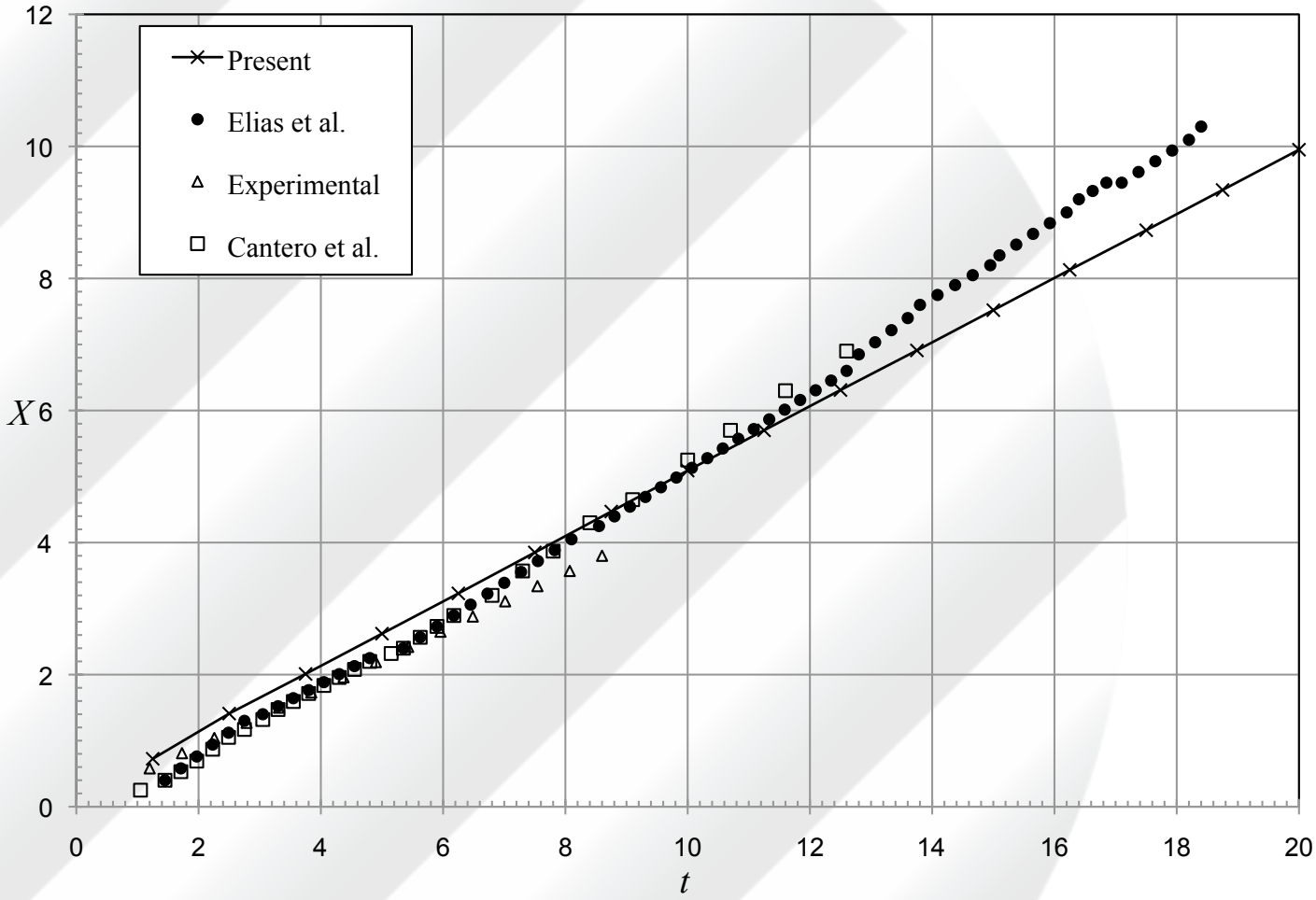


Data for ParaView Parallel Remote Visualization in XDMF/HDF5 formats
Spatial collection of temporal collections

More details: Elias et al, ParCFD09

Planar 3D Lock-Exchange with AMR/C

$X = |x|$ Front evolution



Froude Number

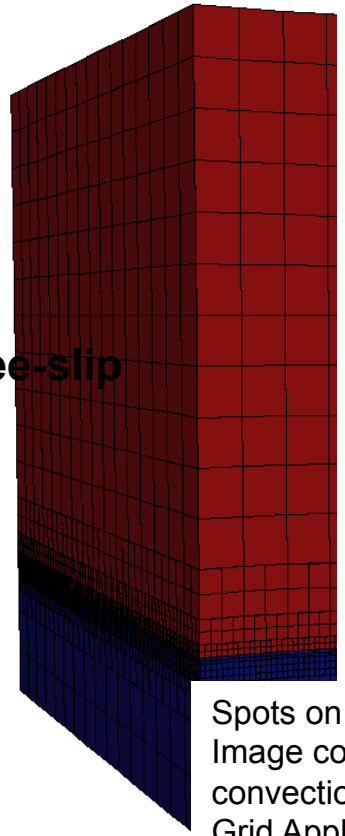
$$Fr = \frac{u_f}{u_b}$$

Work	Fr
Present	0.505
Elias *et al.*	0.579
Härtel *et al.*	0.576
Cantero *et al.*	0.570
Experimental	~0.48

Chemical convection benchmark w/ AMR/C: Stokes flow

Initial mesh: locally

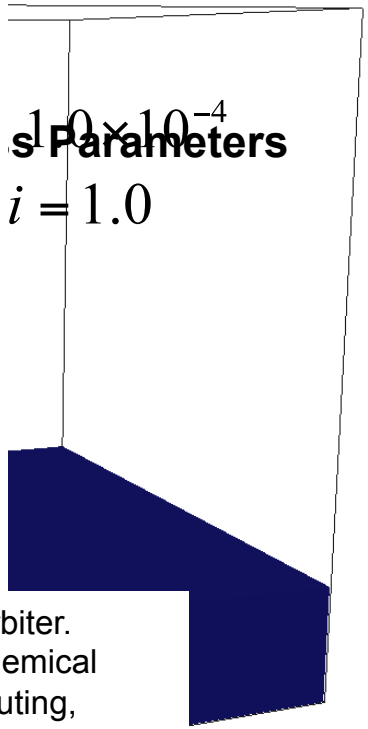
Equivalent uniform r



Spots on the surface of Europa, an icy satellite of Jupiter, as imaged by the Galileo orbiter. Image copyright JPL/NASA, apud Freeman J, Stegman DR, May DA, Moresi L. 3d chemical convection. Proceedings of the APAC Conference and Exhibition on Advanced Computing, Grid Applications and eResearch September 2005

terface

$$).02\cos\left(\frac{\pi x}{\lambda}\right)$$



1.0×10^{-4}

s Parameters

$i = 1.0$

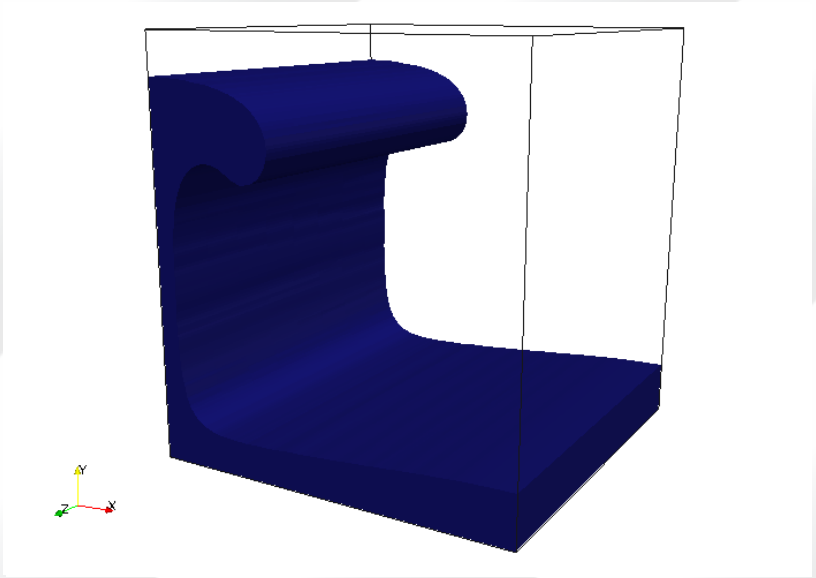
No-slip **Domain**

$$([0,\lambda]\times[0,1]\times[0,\lambda])$$

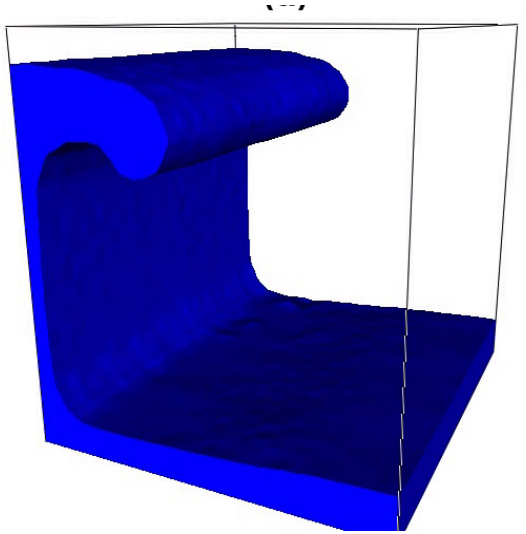
D. F. M. Vasconcelos, A. L. Rossa and A. L. G. A. Coutinho, A residual-based Allen–Cahn phase field model for the mixture of incompressible fluid flows, IJNMF, 2014.

Chemical convection benchmark w/ AMR/C

- **Heavy fluid below the interface**



Present Solution, AMR, RB-AC

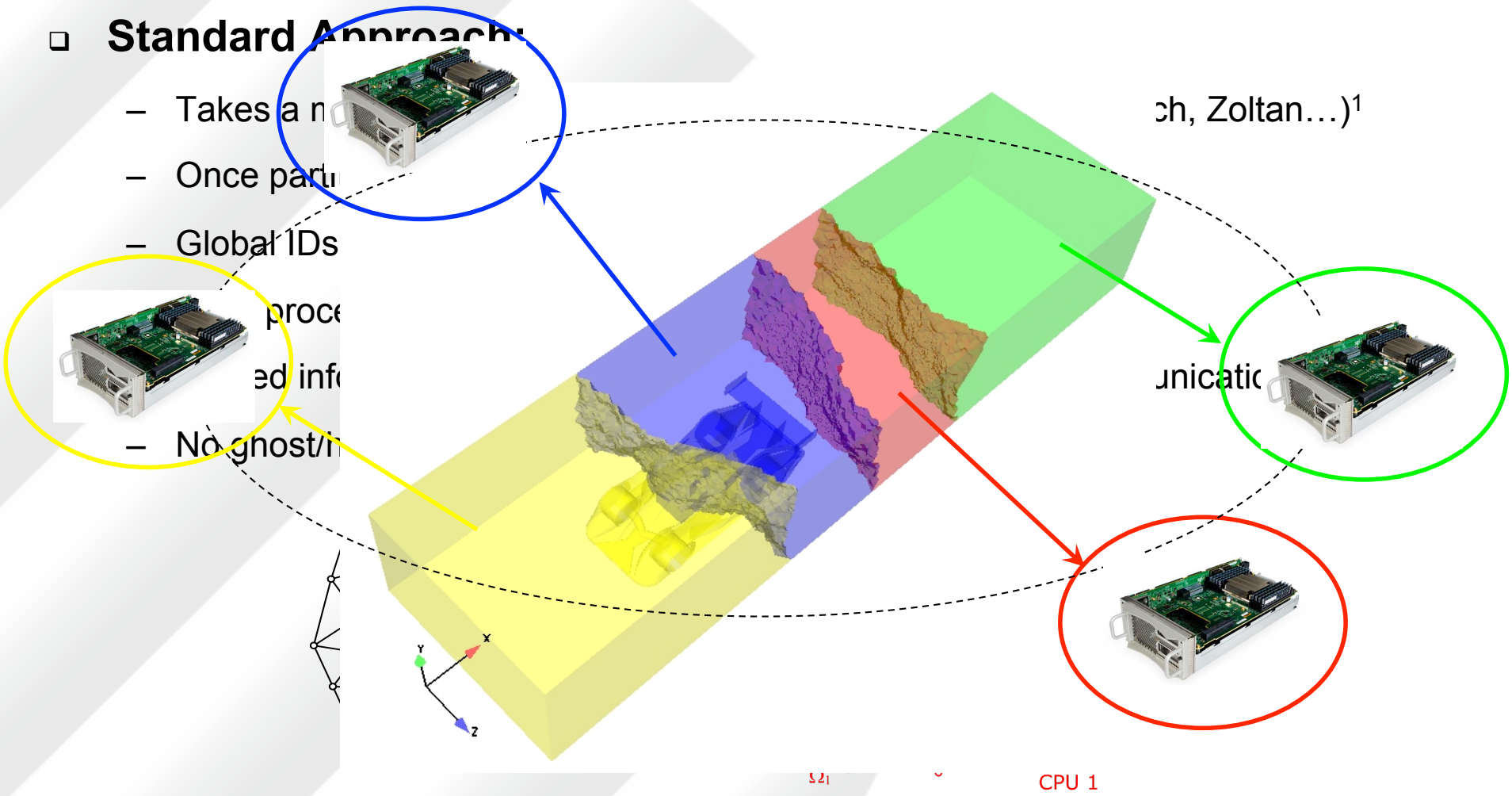


Freeman et al, 2005, mesh 32^3

PARALLEL COMPUTATIONS

Distributed Memory Parallelism

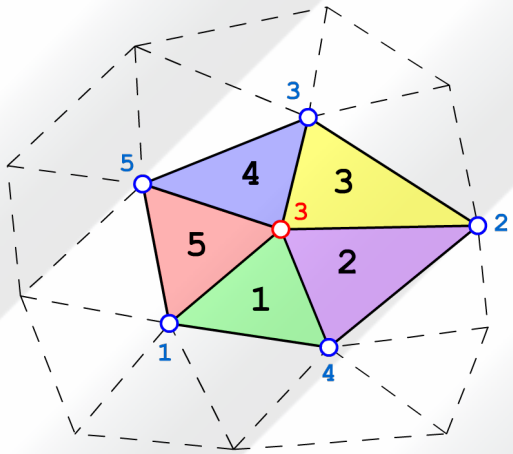
- **Standard Approach:**
  - Takes a n… ch, Zoltan…)[1]
  - Once parti…
  - Global IDs
  - proce…
  - ed inf… unicatio…
  - No ghost/h…

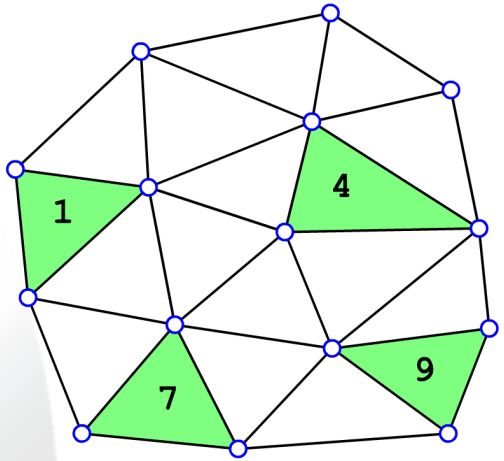


[1]for a recent study see: Z. Shang, Impact of mesh partitioning methods in CFD for large scale parallel computing, Computers & Fluids 103 (2014) 1–5

Threaded Parallelism

- Standard blocked loops to remove memory dependency






```
!$OMP PARALLEL DO
do i=1,nel
   ! retrieve element nodes
   x(no) = x(no) + a
enddo
!$OMP END PARALLEL DO
```

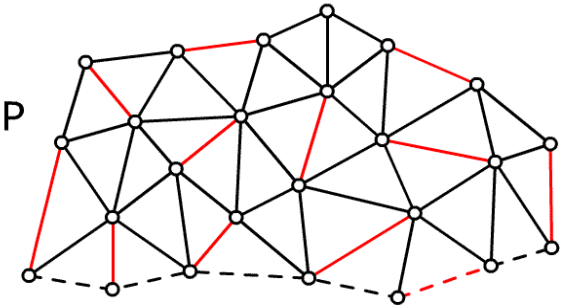
```
ielm = 0
do icor = 1, ncores
   nvec = ielblk(icor)
!$OMP  PARALLEL DO
   do i = ielm+1, ielm+nvec
      ! Retrieve element nodes
      x(no) = x(no) + a
   enddo
!$OMP END PARALLEL DO
   ielm = ielm+nvec
enddo
```

Hybrid Matrix-Vector Product

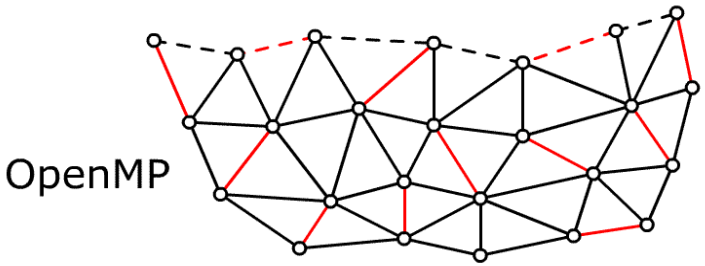
```
iside = 0
DO iblk = 1, nedblk
nvec  = ia_edblk(iblk)
!dir$ ivdep
!$OMP PARALLEL DO
   DO ka = iside+1, iside+nvec, 1
      ...MATVEC computations...
   ENDDO
!$OMP END PARALLEL DO
ENDDO

...over interface nodes...
#ifdef MPICODE
call MPI_AllReduce
#endif
```

Edge-by-Edge

OpenMP



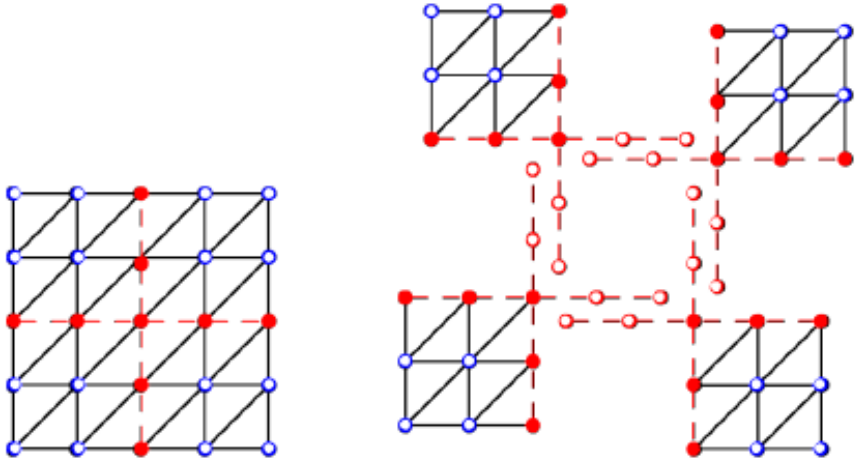
OpenMP



MPI Collective Communications

MPI collective

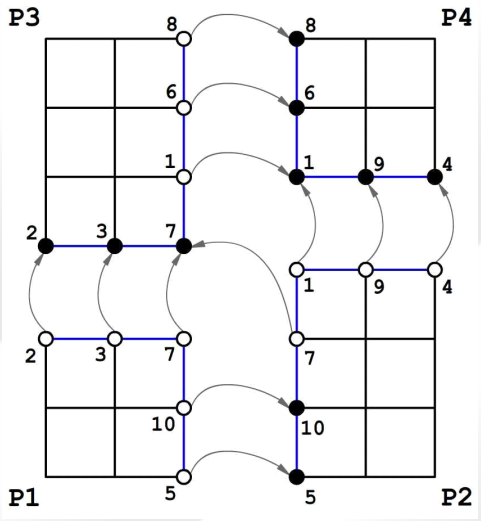
1. All shared equations are synchronized in just one collective operation;
2. Easy to implement;
3. Poor performance for massive parallelism;
4. Some (small) improvements can be done (... *but there's no miracle...*).



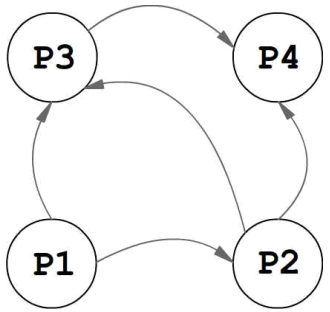
(a) Original mesh (b) Redundant communication

P2P Subdomain Communication

Master-Slave subdomain relationship



(a) Mesh Partition



(b) Communication Graph

Exchange information between neighboring processors implemented in two stages:
(i) slaves processes send their information to be operated by masters
(ii) solution values are copied from masters to slaves.

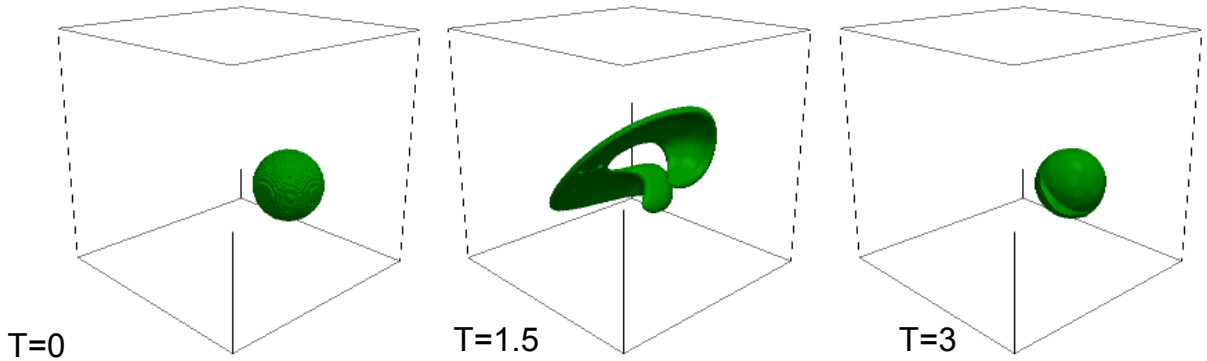
EdgeCFD uses non-blocking send and receive MPI primitives

See also: Karanam, Jansen, Whiting, Geometry based pre-processor for parallel fluid dynamic simulations using a hierarchical basis, Engineering with Computers (2008)

AMR/C and Solver Performance

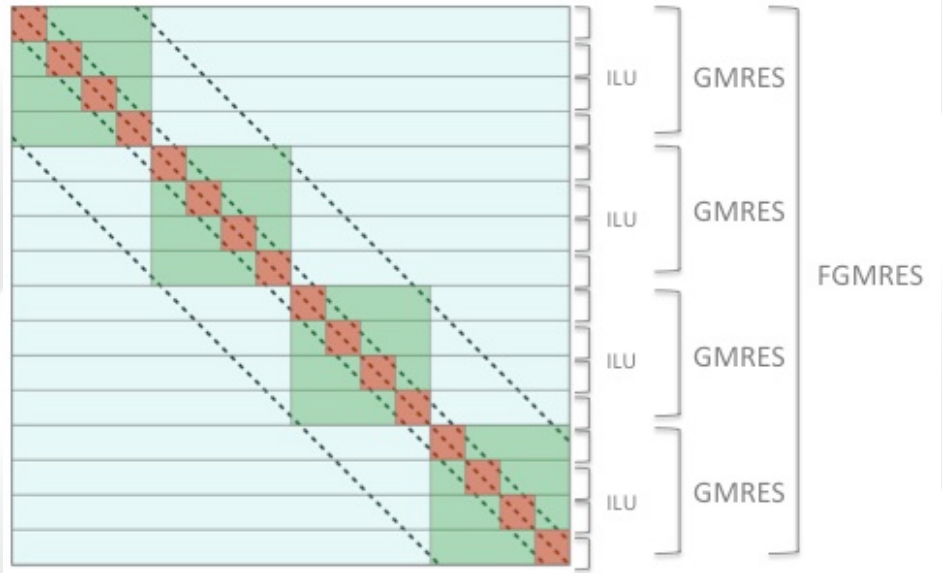
- **Implicit time integration schemes require a solution of large and sparse linear system**
- **Krylov subspace methods + preconditioning strategies**
  - Incomplete LU factorizations methods
  - Powerful method in terms to improve convergence
  - Complex parallel implementation
  - New hierarchical and nested versions
  - Improved version of BiCGSTAB with independent and overlapped dot products
- **Making ILU more suitable for parallel architectures**
  - Block preconditioners with local ILU factorizations
  - Domain decomposition preconditioners: Additive-Schwartz and
  - Block Jacobi
- **How the choice of block ILU affects the simulation performance using AMR/C using different unknowns orderings**

Three-dimensional deformation problem

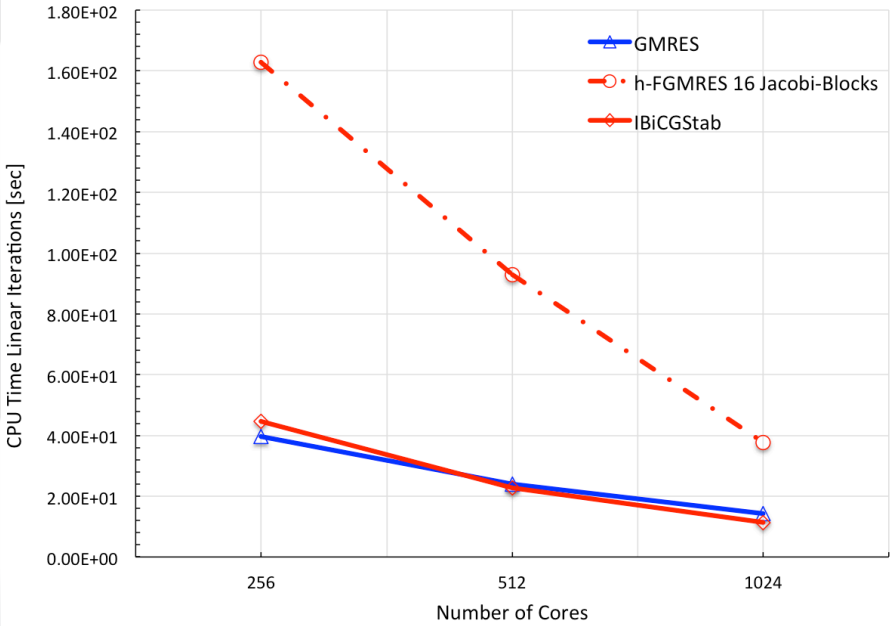


$$u_x = 2sin^2(\pi x)sin(2\pi y)sin(2\pi z)g(t)$$
$$u_y = -sin(2\pi x)sin^2(\pi y)sin(2\pi z)g(t)$$
$$u_z = -sin(2\pi x)sin(2\pi y)sin^m 2(\pi z)g(t)$$

Mesh: 150X150X150 Hex 8
3,375,000 elements
SUPG FEM libMesh+Petsc



2-level h-FGMRES for a sparse N × N matrix partitioned over 16 cores, with 4 Jacobi Blocks at each level for preconditioning.

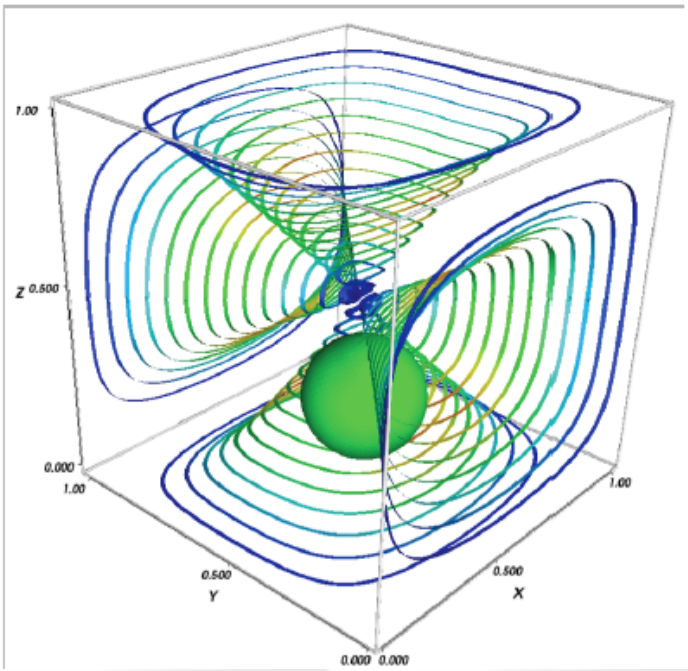


Performance of GMRES(30), h-FGMRES(30) with 16 Jacobi-Blocks and IBiCGStab (all with ILU(0))

McInnes, L. C., Smith, B., Zhang, H., & Mills, R. T. (2014). Hierarchical Krylov and nested Krylov methods for extreme-scale computing. Parallel Computing, 40(1), 17–31.

Three-dimensional deformation problem with AMR/C

- The sphere is deformed by vortices and stretched out very thinly.
- The velocity field is given by



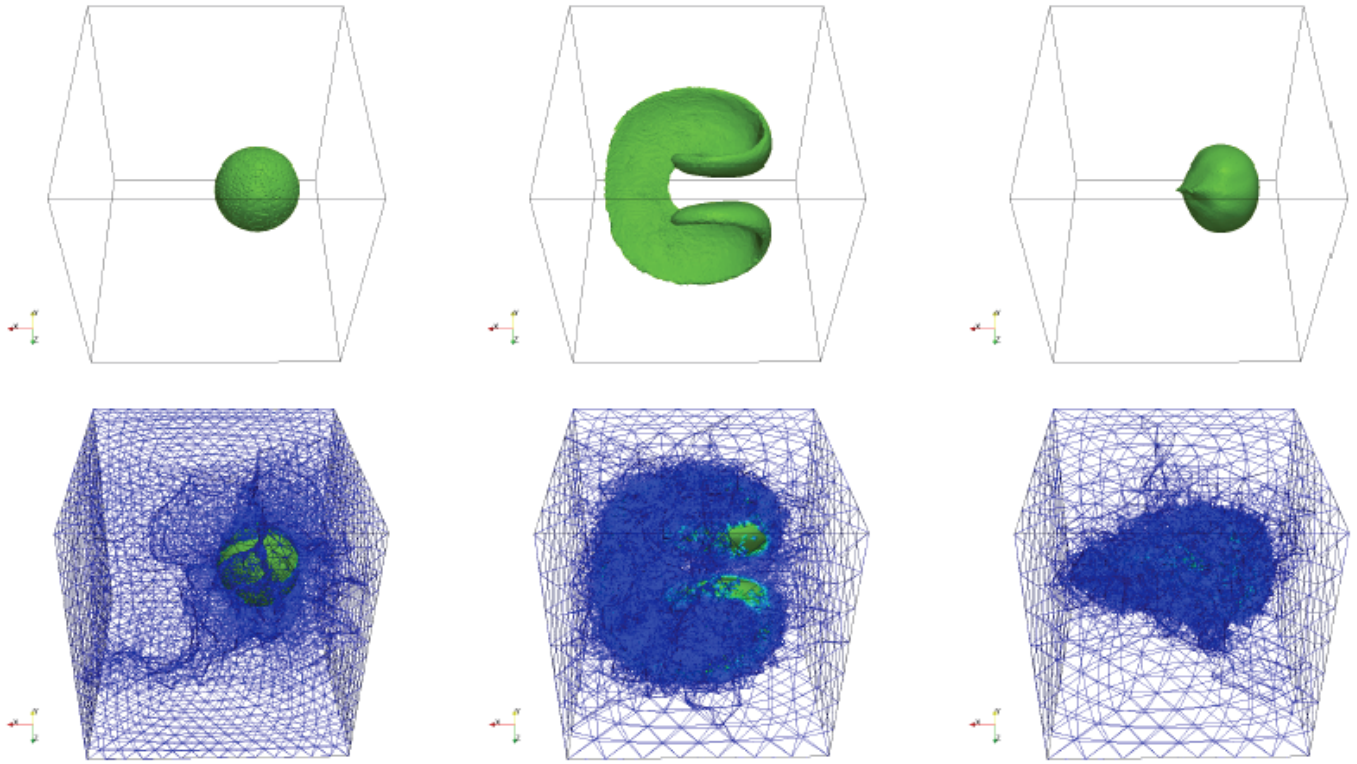
- AMR Parameters:
  - Initial Mesh: 1,130,949 linear tetrahedra
  - Refine fraction: 0.9
  - Coarse fraction: 0.1
  - $h-level$: 4
  - AMR every 5 time steps
- Solver Parameters:
  - GMRES(30) - relative tolerance 10^{-6}.
  - Preconditioners: ASM $(1-level)$ and Block-Jacobi
  - Local ILU(0) and ILU(1)

SUPG FEM with linear tets

See Camata et al, IJNMF, 2012

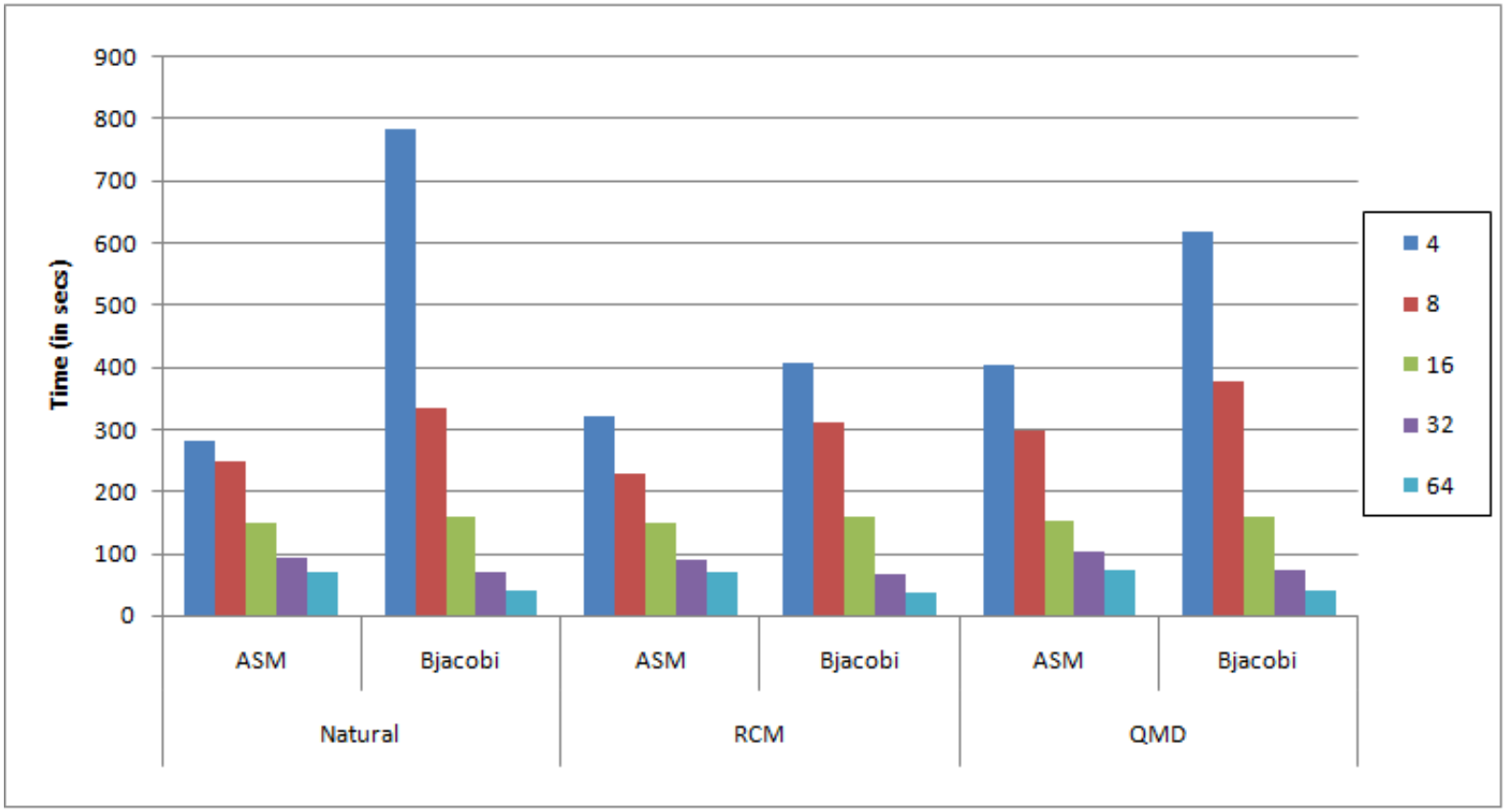
Results 3D deformation problem

- Solution and mesh configuration of the 3D deformation problem at three times, $T = 0$, $T = 1.5$, $T = 3.0$ time units.



Results 3D deformation problem

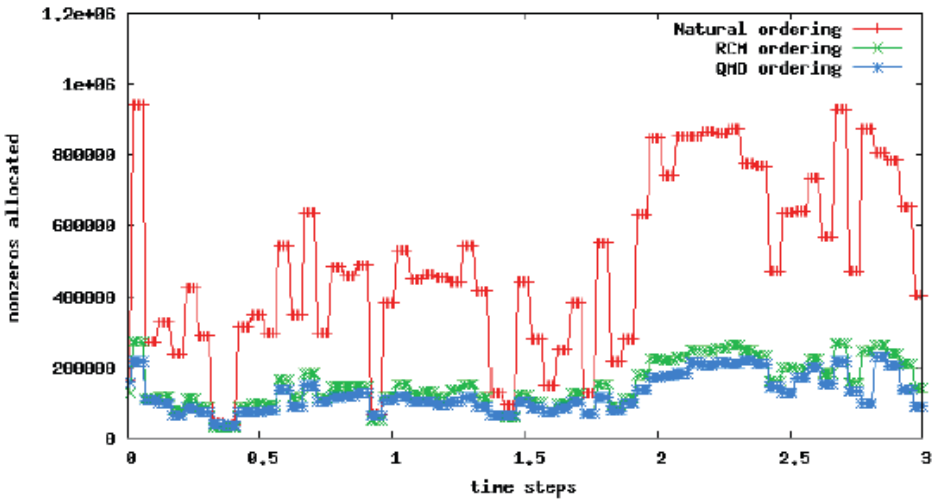
GMRES(30) + local ILU(0) performance using Block-Jacobi and ASM on Altix ICES 8200 [1]



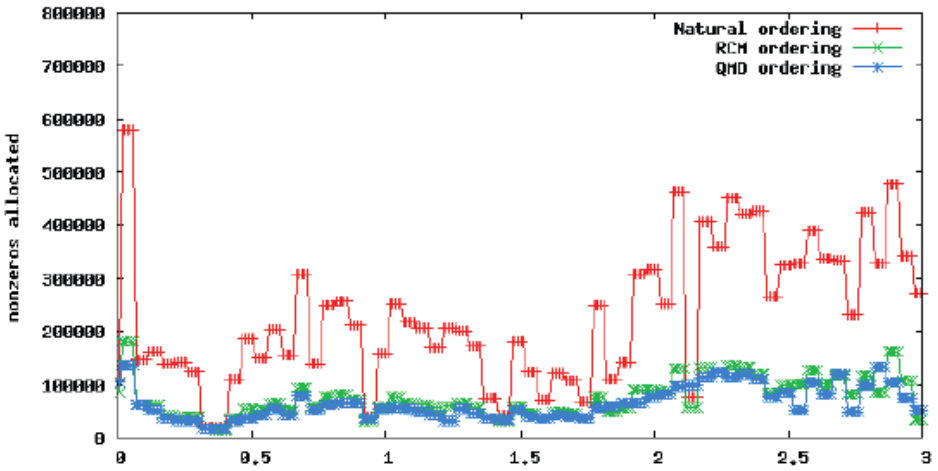
[1]Each compute node has two Intel Quad-Core 2.66GHz, 8MB L2 cache on-die and 16GB of memory

Nonzeros allocated using ILU(1) preconditioner and 64 processors

ASM



Block-Jacobi



Pushing the Limits

PARALLEL OCTREE MESH GENERATION

Parallel Octree Mesh Generation

- **Growing availability of parallel machines**
- **Improvements on scalability of numerical solutions (FEM, FVM)**
- **Scalable Octree meshing and solver strategies**
  - Dendro lib: scales oup to 4,000 cores (Sundar et al, 2007)
  - Tu et al, runs in 67,000 cores (2005)
  - p4est: scales up to 220,320 cores, Burstedde et al (2011)
  - Park & Shin (2012): octree meshes using GPGPU
- **Our objectives**
  - Present a parallel octree generator able to representing arbitrary surfaces
  - Extract conforming tetrahedral meshes from the resulting octree
  - Perform a parallel scalability analysis

Linear Octree

- An octree is a tree data structure in which internal nodes has exactly eight children [3] (Figure 1)
- Often used to partition a three dimensional domain space by recursively subdividing it into eight octants
- Linear Octree
  - ✓ complete list of leaf nodes
  - ✓ octants are encoded by a scalar key: Morton Code
- Advantages:
  - ✓ do not require to store internal nodes
  - ✓ reduce overhead associated with pointers use

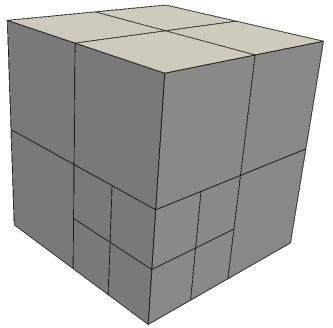


Figure 1: Octree – sudivision of a cube into octants

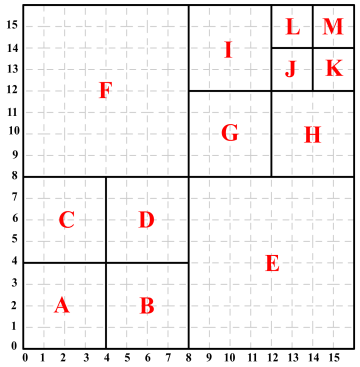


Figure 2: Quadtree - 2D structure

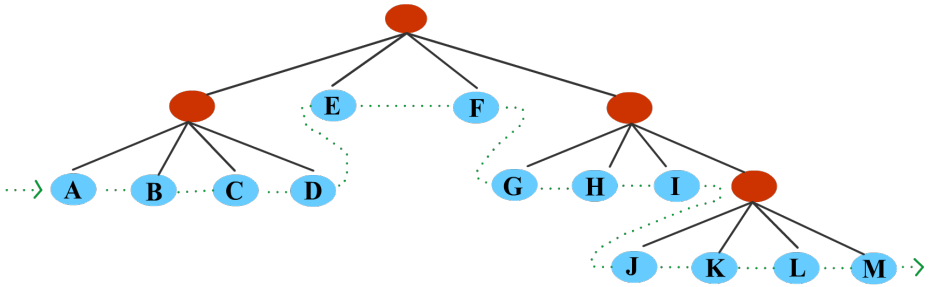
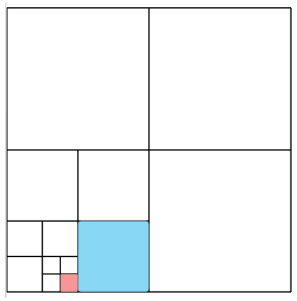


Figure 3: tree structure

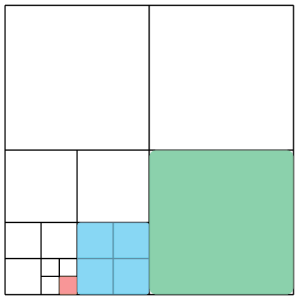
Morton code
- ✓ D's left lower corner: (4,4)
- ✓ Binary representation: (100,100)
- ✓ Interleave the bits: 110000
- ✓ Append D`s level: 110000 ← **10**

Octree Algorithms

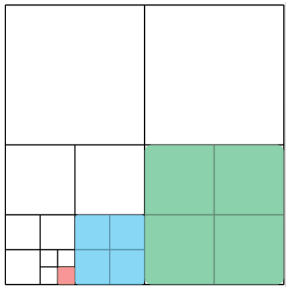
- Octree Construction
  - ✓ generate a list of octants equally distributed between the cores.
- Octree partition
  - ✓ redistribution of the octants among processes with the objective to reach load balance
  - ✓ each process stores a contiguous chunk of leaf octants
- Octree Refinement
  - ✓ non-recursive refinement algorithm transverses all leaf octants for each local octree replacing an octant with its eight children
- 2:1 Balancing
  - ✓ no leaf octants sharing at level l shares an edge or face with another leaf at level greater than l + 1



Unbalanced octree



Riplet octree



balanced octree

- Meshing
  - ✓ Find anchors and hanging nodes
  - ✓ Find sharing nodes
  - ✓ Get element connectivity

Manipulating STL surfaces

- Necessary to represent complex solid boundaries present in computational solid and fluid mechanics applications
  - ✓ Surfaces are given by a triangulation, typically obtained from CAD package as STL files
- Finding interceptions is based on bounding box threes (BBT)
  - ✓ Advantage: Allows fast overlap rejection test
  - ✓ Computational cost: O(*nlogn*)
- Finding octants located inside the solid boundaries is based on ray tracing

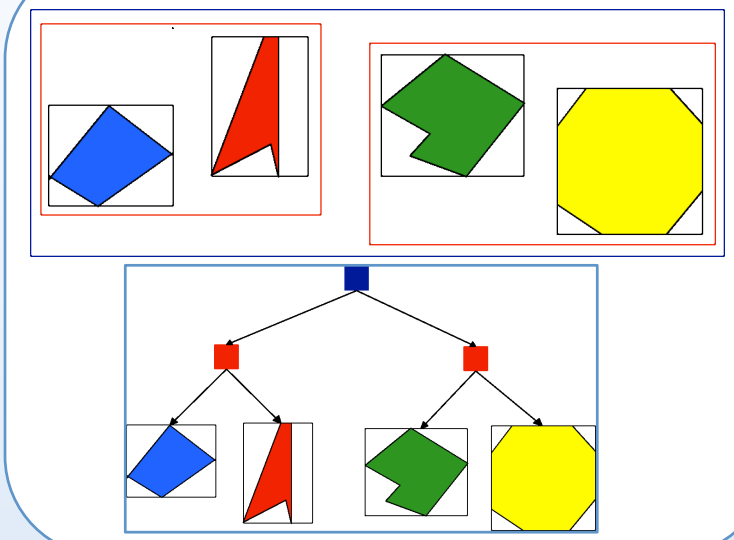


Figure 4: Bounding Box Tree scheme

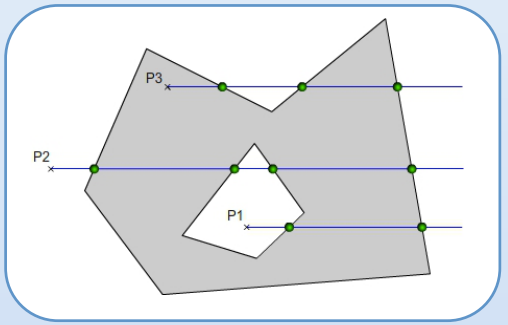
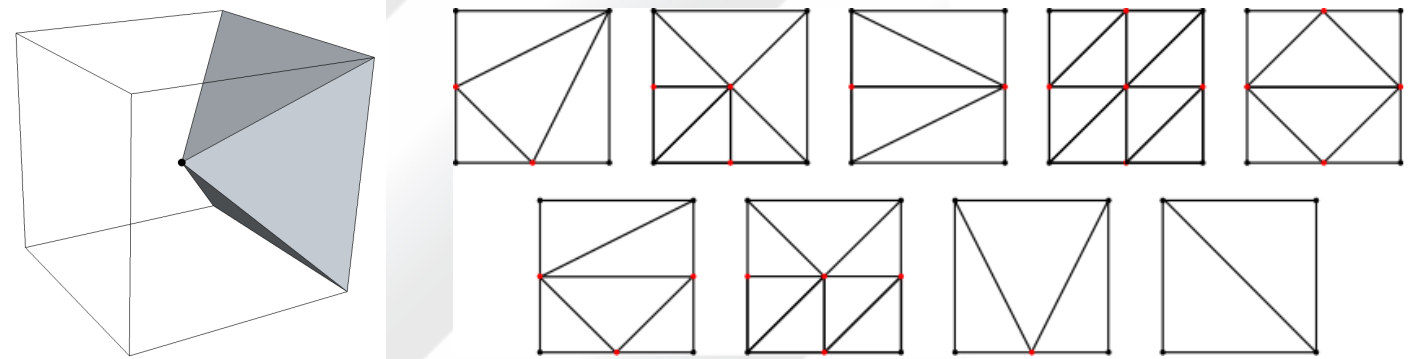


Figure 5: Ray trancing

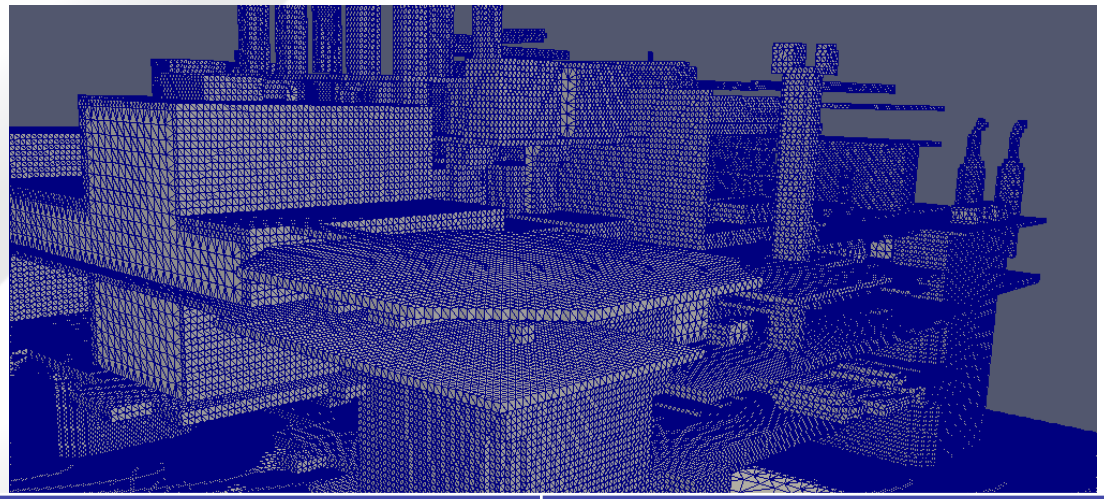
Conforming Techniques

- **FREY e GEORGE (2000) / BERG et. al (1998)**
  - Decompose octants in 6 pyramidal elements by inserting a central node
  - Define 9 templates for face triangulation
  - Connect all face nodes with the central node



  - Does not require modifications in the octree construction
  - Embarrassing parallel (does not need neighboring info)
  - Templates for all possible hanging nodes configuration

High Fidelity Parallel Mesh Generation



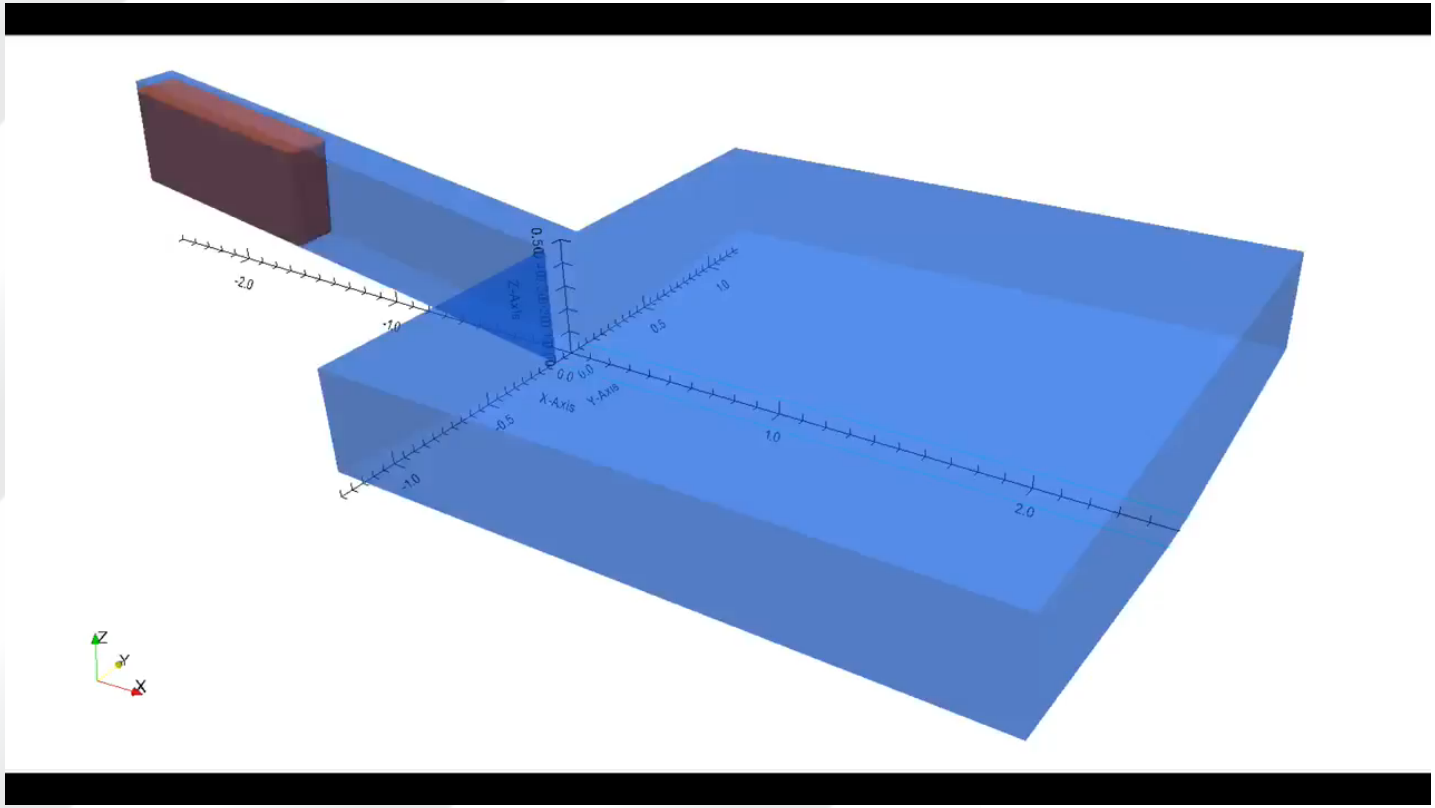
Data	Storage (TB)
Solver Edge-by-Edge	3.5
Solver EBE	17.5
Solver CSR+ILU(0)	5.83
Solution (10^4 saved time steps)	524.0

Experiments conducted in Stampede at TACC UT-Austin, USA

Pushing the limits: Uncertainty Propagation on Particle Laden Flows

EXPLORING THE STOCHASTIC SPACE

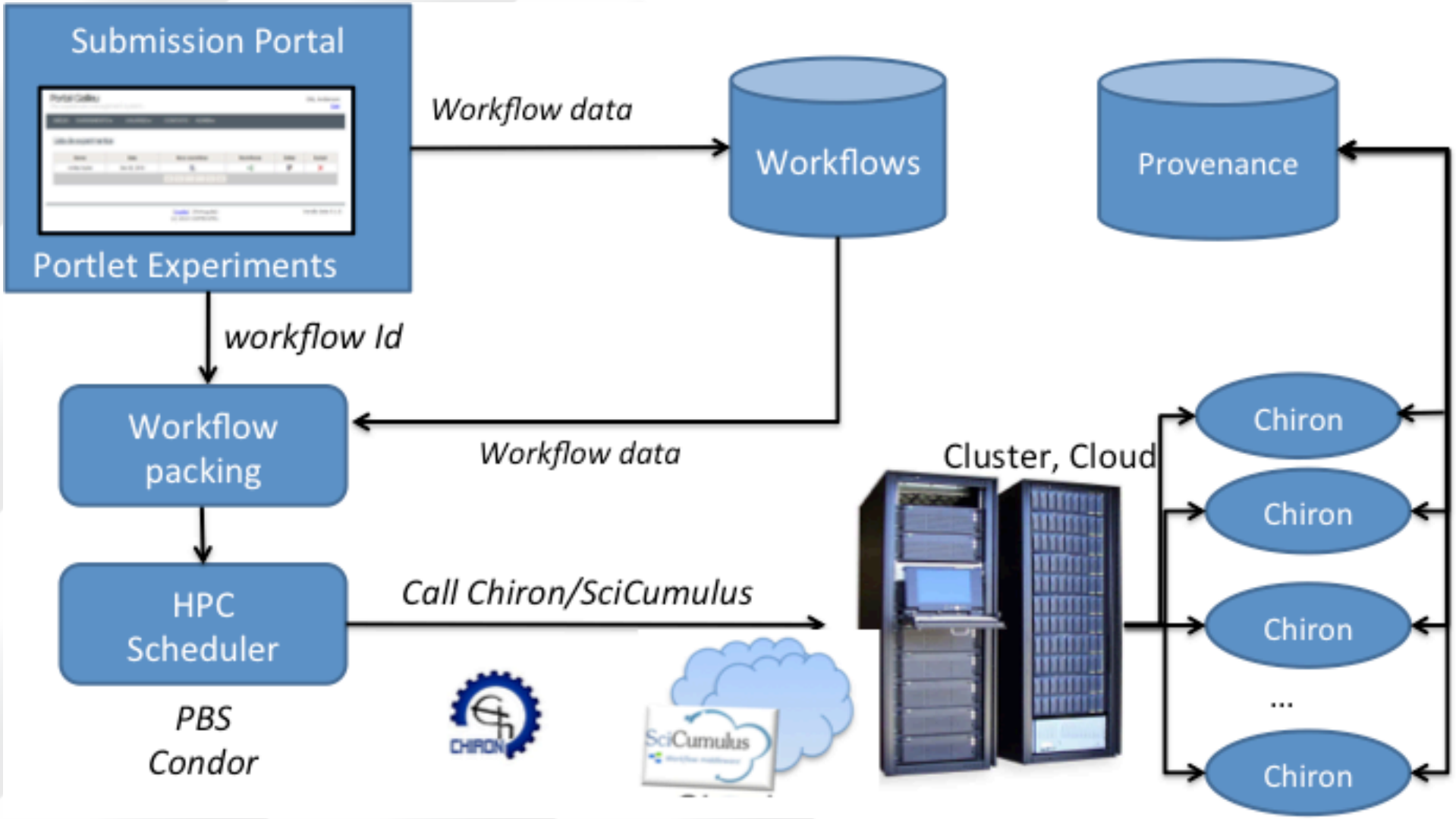
Why currents move and deposit?



Investigation on how uncertainty in initial conditions and settling velocities propagate
Initial concentrations and settling velocities considered random fields
Simulations done with EdgeCFD[1] with Stochastic Collocation Method

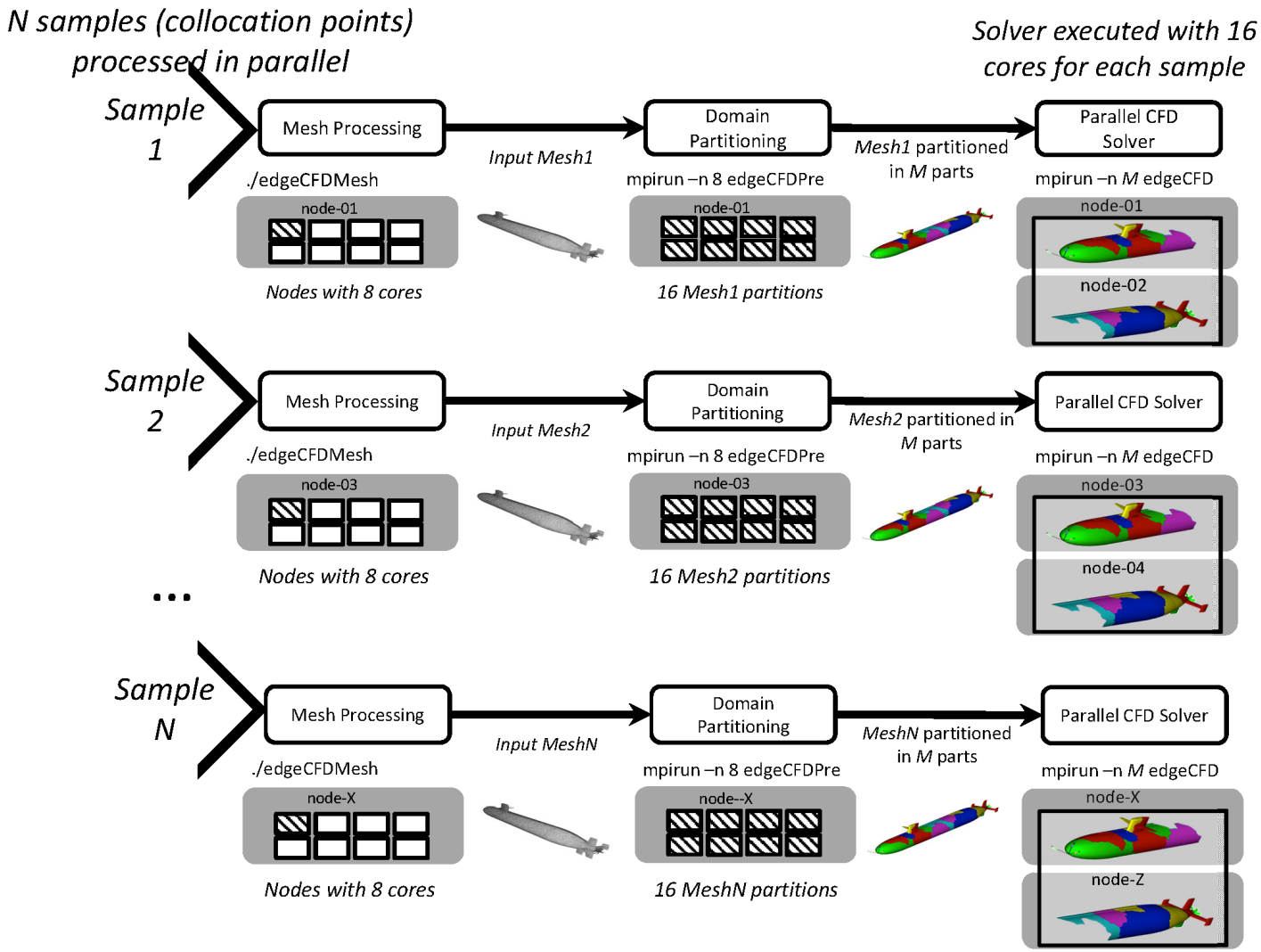
[1]G. M. Guerra, et al, Numerical simulation of particle-laden flows by the residual-based variational multiscale method, IJNMF 73 (8) (2013) 729–749.

Computational Infrastructure for Exploring the Stochastic Space



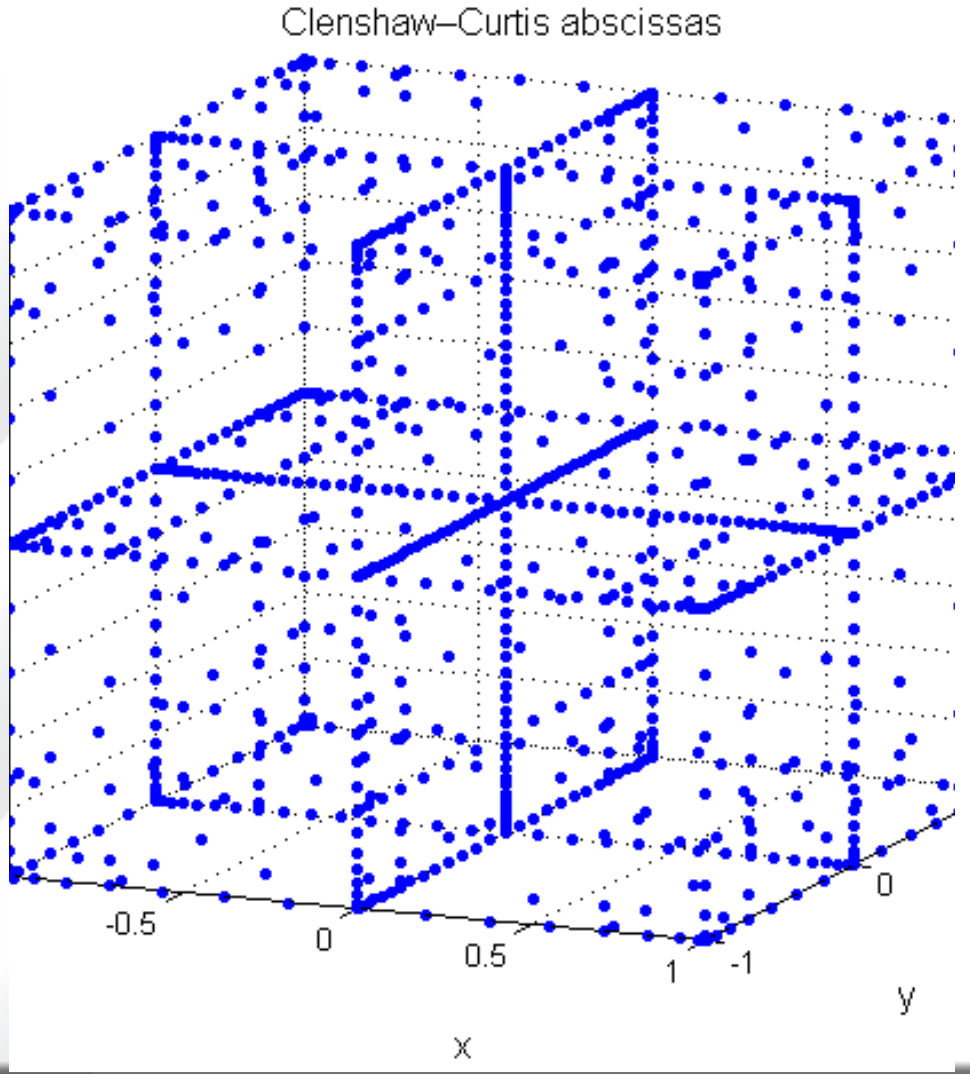
E. S. Ogasawara, et al: Chiron: a parallel engine for algebraic scientific workflows, Concurrency and Computation: Practice and Experience 25 (16) (2013) 2327–2341.

Two-level Parallel Strategy

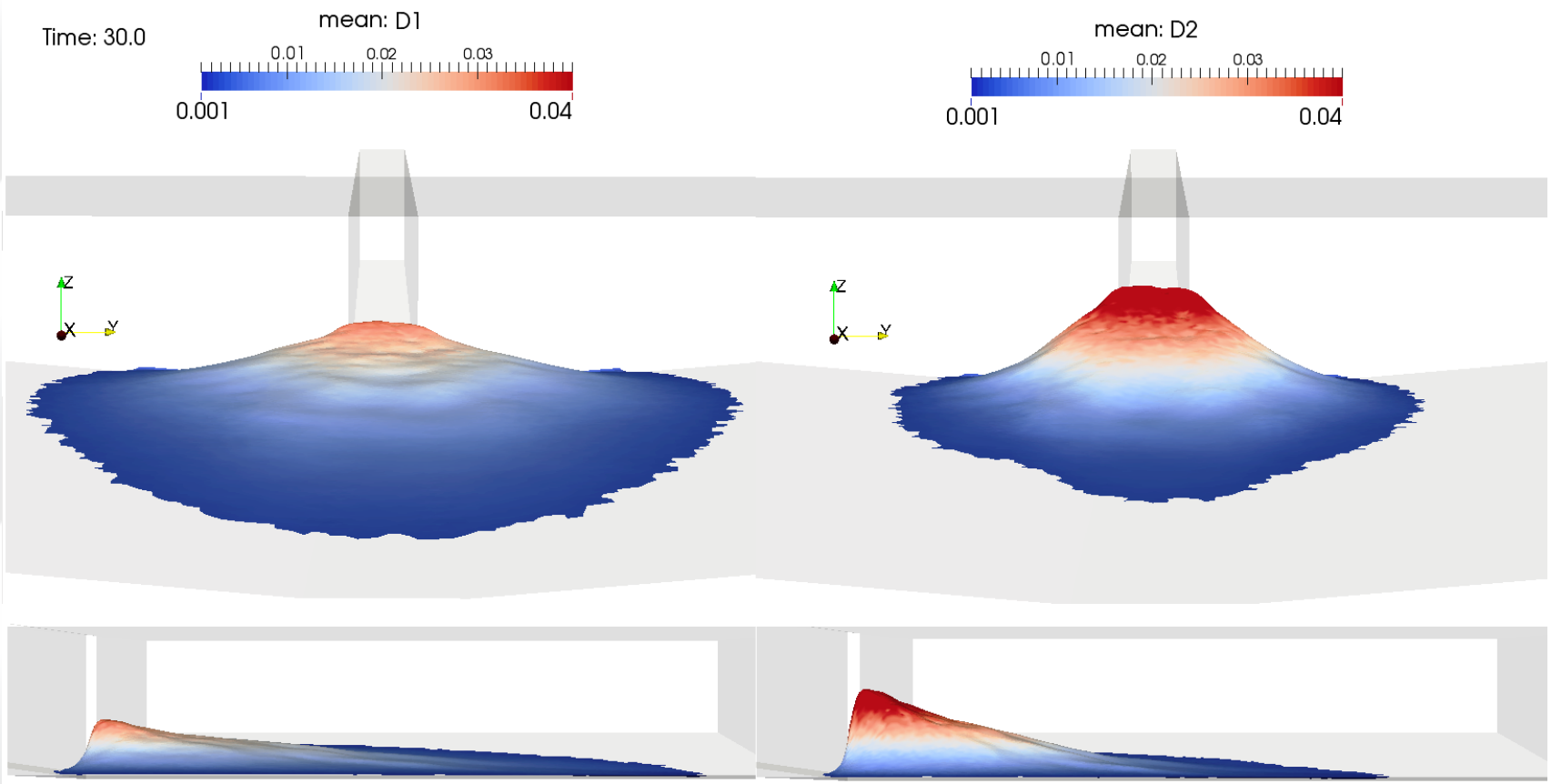


Computational Data

- Level 6 Sparse Grid Stochastic Collocation Method
- 1073 samples (each one a parallel job) run on HPC machine. Average run time for each sample: 5h
- 108 jobs with 10 samples managed with Chiron
- Sequential time: 202 days
- Two-level parallel run: 24 days

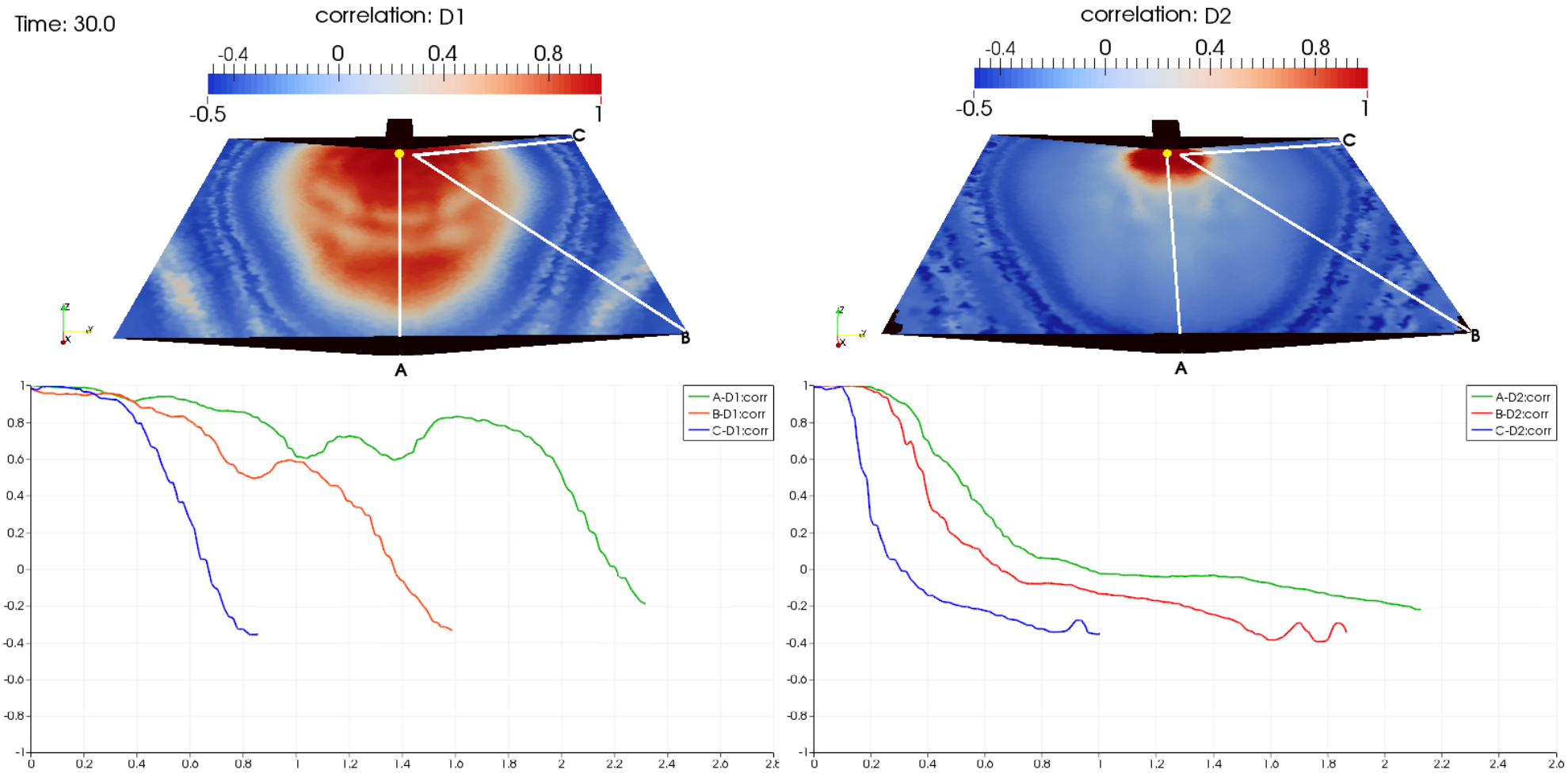


What geologists want to know



3D representation of the deposition for both constituents at *t*=30: top and lateral views.
Colors on the surfaces correspond to the deposits thickness.

What geologists want to know



Correlation map for the depositions

CONCLUDING REMARKS AND DISCUSSION

- **Multiphysics problems have been addressed**
  - Prospect of predicting the behavior of complex systems combining multiple physical phenomena is one of the main motivations for extreme computing
  - Scaling-up components is challenging. How to effectively combine them? How to generate, manage and visualize data?
  - Need of an integrated infrastructure
- **Advanced Algorithms and Solvers**
  - Smart algorithms (Inexact Newton, timestep control) really pays off
  - Solvers are still challenging
- **Parallel Computations**
  - Communication issues of paramount importance
  - Adaptivity introduces an extra complexity layer (needs repartition)
  - Mesh generation still a problem
- **Stochastic Multiphysics: New Frontier**
  - Managing the complexity of sampling stochastic space
  - Issues: data management, fault tolerance, data provenance, etc

Acknowledgements:

- Faculty:
  - HPC: A. Coutinho (PI), R.Elias
  - Civil Engineering: J. Alves
  - Mechanical Engineering: F. Rochinha
  - Computer Science: M. Mattoso
- Pos-Docs, Research Staff and Students:
  - A. Aveleda, C. Barbosa, D. Barcarolo, D. de Bruyckner, O. Caldas, J. Camata, A. Cortes, J. Dias, I. Ghisi, M. Goncalves Jr, J. Gonzalez, G. Guerra, N. Guevara Jr, F. Horta, E. Lins, A. Mendonça, E. Ogasawara, D. Oliveira, A. Rossa, F. Seabra, C. Silva, D. Vasconcelos, S. Zio
- Funding: Petrobras, ANP, MCT/CNPq, MCT/FINEP, Petrobras
- Computer Resources: NACAD/COPPE/UFRJ, TACC/UT Austin, Intel

Visit us at www.nacad.ufrj.br