FIFTH Brazil - France
Workshop on High Performance Computing an Scientific Data Management Driven by Highly Demanding Applications

A review of NACAD's developments within HOSCAR and future perspectives on H2020

Alvaro Coutinho
alvaro@nacad.ufrj.br

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

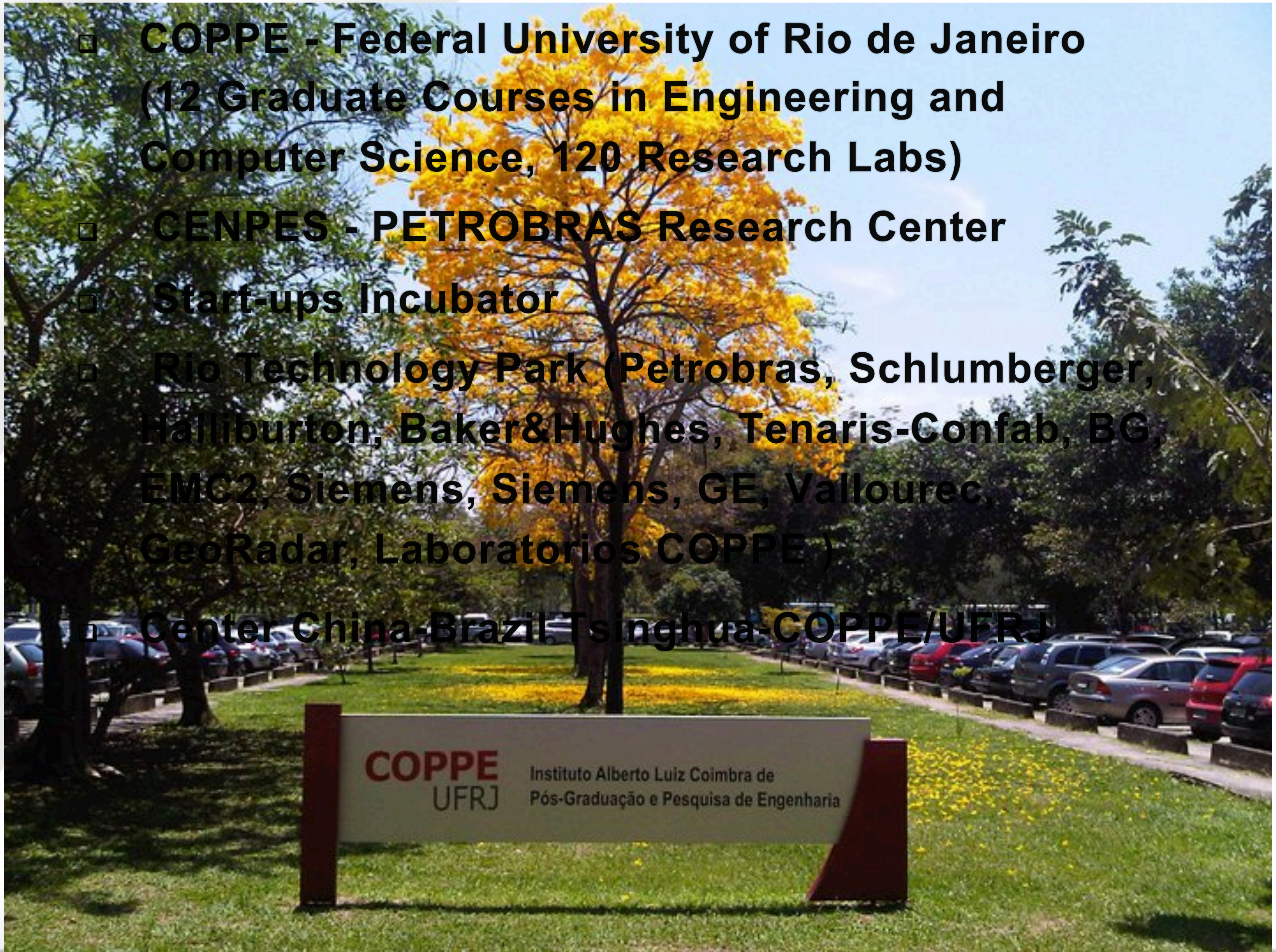
COPPE UFRJ 50

Contents

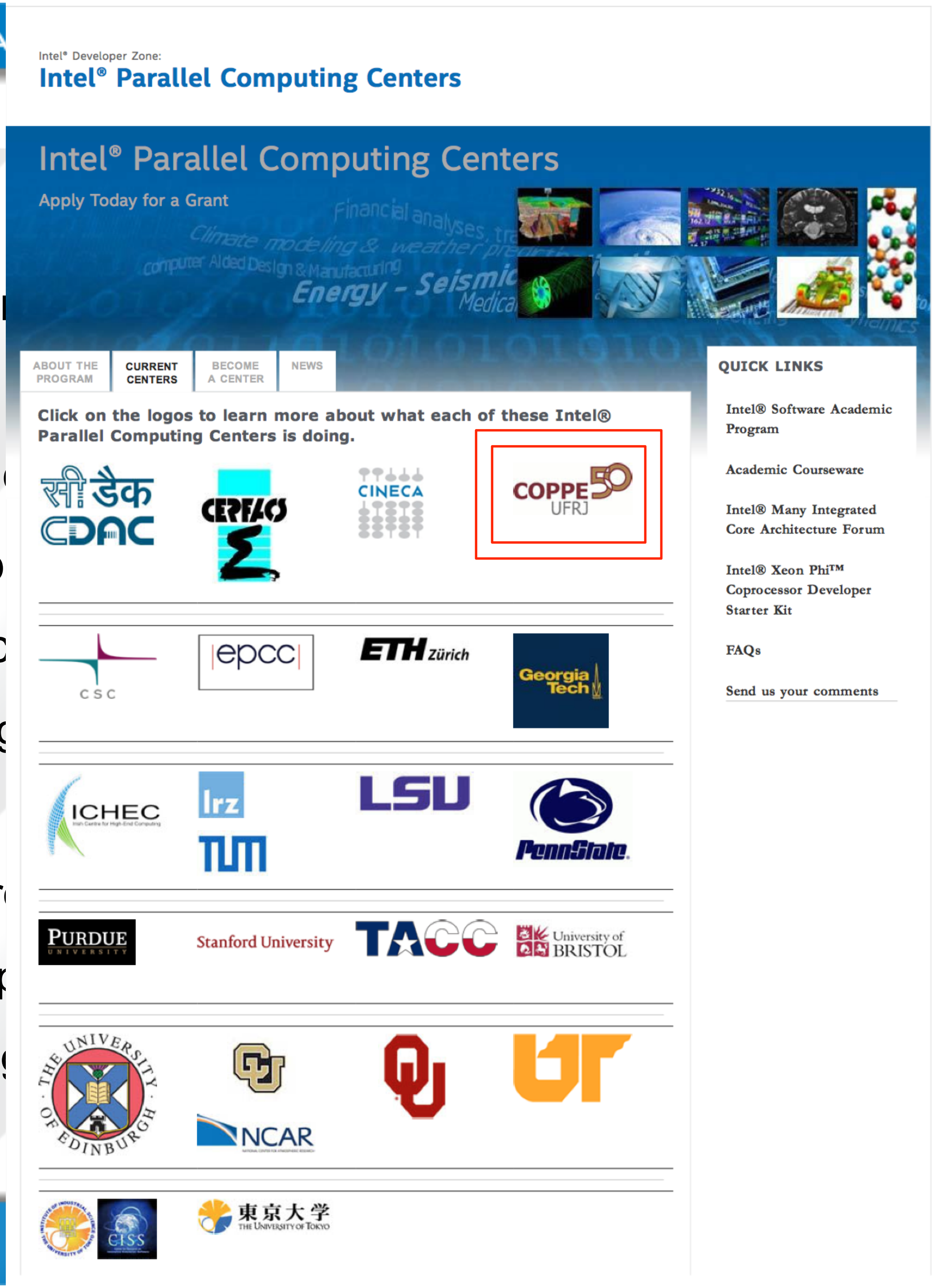
WHO WE ARE AND WHAT WE DO

COPPE/UFRJ Innovation Ecosystem

- COPPE - Federal University of Rio de Janeiro (12 Graduate Courses in Engineering and Computer Science, 120 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 Per ... mputer Science
- To provi ... omputing
- Develop ... of relevanc
  - Energ
  - Civil,
  - Envir
  - Comp
  - Biolog



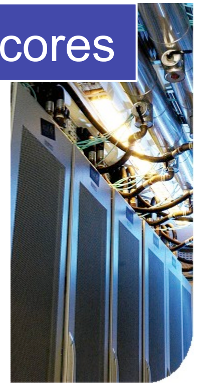
SINAPAD

Timeline

Machines

Sun Galileu

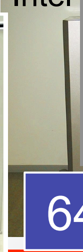
65Tflops, 7K cores



SGI Altix ICE

COPPE NCP I

Intel



COPPE's new machine 2015-206
253 Compute Nodes 6072 Cores (506 Processadors
Intel Xeon Haswell E5-2670V3 12-Cores
2.3GHz)16.192GB Mem DDR4

r vendor

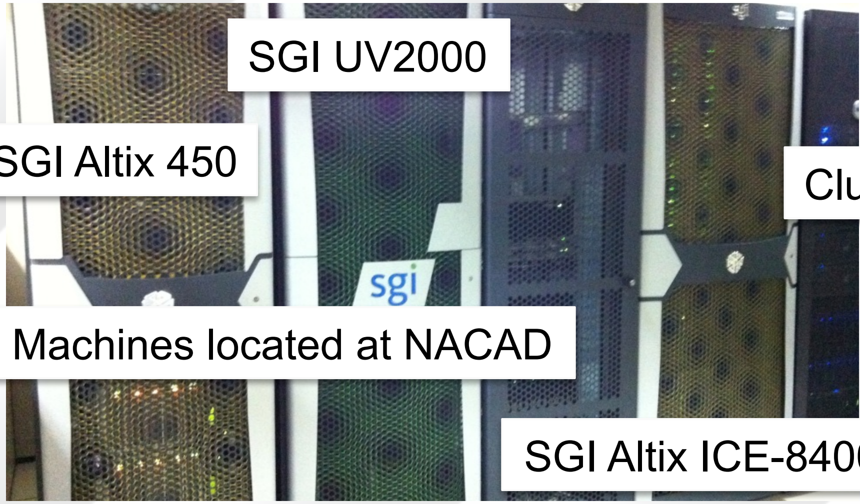
Year

640Mflops, 8 procs

1988 1990 1995 1999 2005 2010

1st Supercomputer built in Brazil

HPC Center Systems



SGI UV2000

SGI Altix 450

Cluster Dell

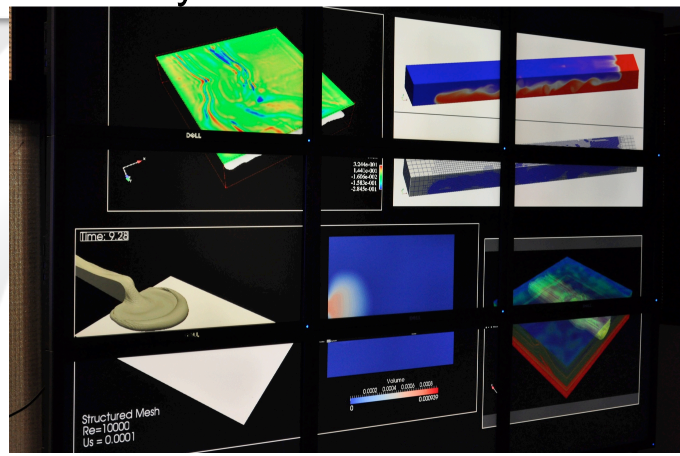
Machines located at NACAD

SGI Altix ICE-8400

NetApp Storage 500TB

Galileu, Oracle Server #1 LA 2010
Memory 21TB, Storage 200TB at Lamce, Tech Park

Dell's Tiled Wall Display+
RV Facility at Lamce at the Tech Park



NACAD's Main Academic Partners

TACC

Laboratório Nacional de Computação Científica

inria informatiques mathématiques

BSC

.inf INSTITUTO DE INFORMÁTICA UFRGS

LAMCE PEC UFRJ COPPE

LASPOT Programa de Engenharia Eletrica COPPE / UFRJ

CentraleSupélec

VIRTUS IMPAVIDA UFPE

ntt Núcleo de Transferência de Tecnologia

ITA

NACAD's Industrial Partners

BR

DELL

Adaptive COMPUTING

intel

ORACLE

CISCO

sgi accelerating results™

NetApp

Eletrobras Cepel

EMC²

PETREC PETROLEUM RESEARCH AND TECHNOLOGY

Schlumberger

ikon Science

HALLIBURTON

中国中化集团公司 SINOCHEM GROUP

Itautec

NACAD's on HOSCAR

Mattoso's talk

- **App Track**
  - Multiphysics
  - Uncertainty Quantification
  - Enabling Technologies
    - *Mesh generation, solvers, visualization*

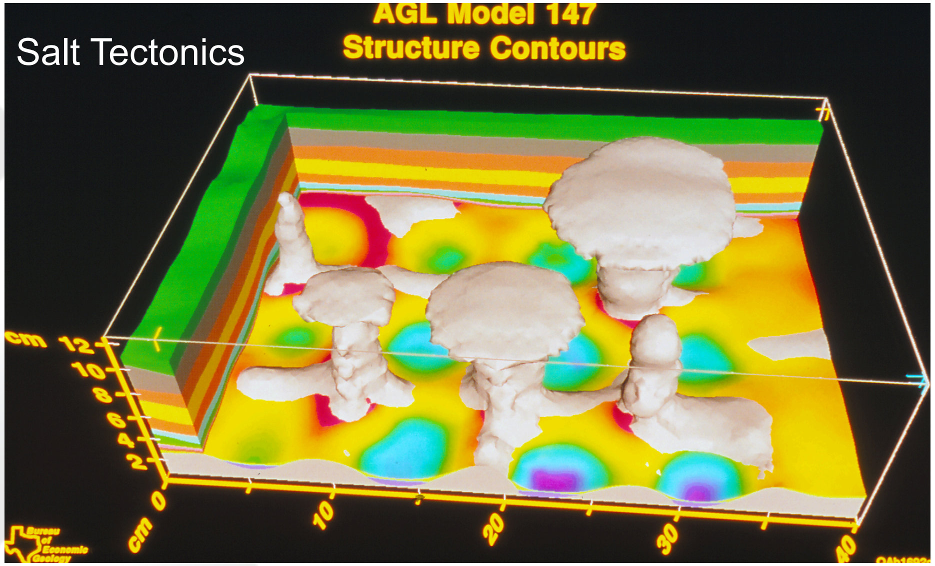
This talk

- **Big Data Track**
  - Scientific Workflows
  - Databases
  - Provenance
  - User steering
  - Dynamic Loops

MULTIPHYSICS

Motivation


Underwater turbidite currents



Salt Tectonics


Green water on ship decks

Governing PDE's: Fluids

- **Incompressible Flows**

Momentum balance in Eulerian frame

$$\boxed{\rho(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}\cdot\boldsymbol{\nabla}\mathbf{u} - \mathbf{f}) - \boldsymbol{\nabla}\cdot\boldsymbol{\sigma} \;= 0 \qquad \text{on } \Omega}$$

$$\boxed{\boldsymbol{\nabla}\cdot\mathbf{u} \;= 0} \quad \text{on } \Omega$$

Incompressibility constraint

$$\boldsymbol{\sigma}(p,\mathbf{u}) = -p\mathbf{I} + 2\mu\boldsymbol{\varepsilon}(\mathbf{u}) \;, \qquad \boldsymbol{\varepsilon}(\mathbf{u}) = \frac{1}{2}((\boldsymbol{\nabla}\mathbf{u}) + (\boldsymbol{\nabla}\mathbf{u})^T)$$

$$\mathbf{u} \;=\; \mathbf{g} \text{ on } \Gamma_{\mathrm{g}} \;, \qquad \mathbf{n}\cdot\boldsymbol{\sigma} = \mathbf{h} \text{ on } \Gamma_{\mathrm{h}}$$

- **Incompressible Navier-Stokes equations**
  - Hypothesis: NS is valid for ALL scales

Governing Equations for Solids and Structures

- **Equation c**

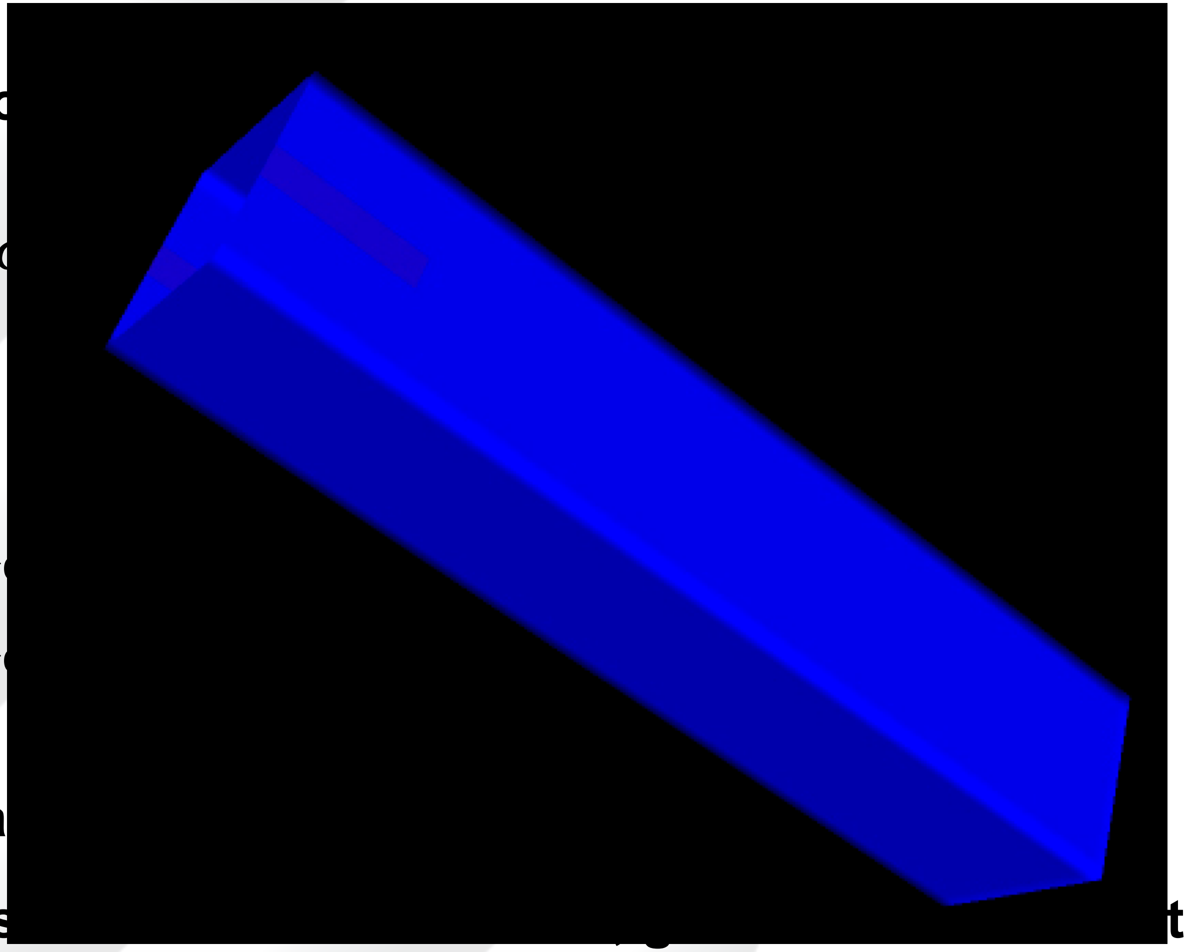
$$\rho\frac{\partial^2\mathbf{u}}{\partial t^2}-\nabla.$$

$$\mathbf{u}=\mathbf{g}$$

$$\sigma\cdot\mathbf{n}=\mathbf{h}$$

$$\mathbf{u}(\mathbf{x},0)=\mathbf{u}$$

$$\dot{\mathbf{u}}(\mathbf{x},0)=\dot{\mathbf{u}}$$



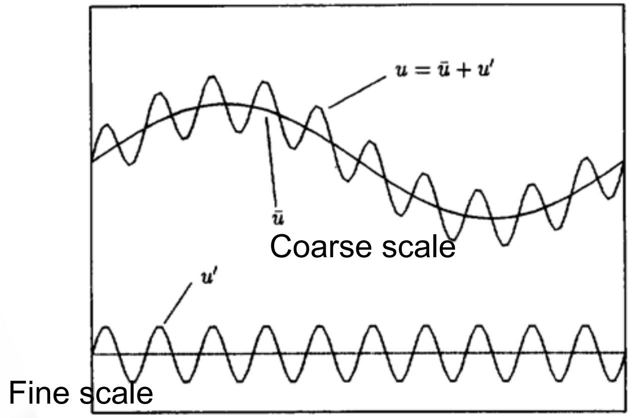
- **Material la**
- **Difficulties ... ties, contact, absorbing bcs**

Courtesy of J. Alves, COPPE

General Weak Formulation for Governing Equations

$$\textit{Find} \quad \mathbf{U} \in \mathcal{V} \textit{ such that } \forall \mathbf{W} \in \mathcal{V}:$$

$$\boxed{B(\mathbf{W},\mathbf{U}) = (\mathbf{W},\mathbf{F})}$$



Remarks:

- Boundary conditions are built in $\mathcal{V}$
- (.,.) is the standard notation for inner product ➔ eqs weakly satisfied
- B for Lagrangian equations is the standard Galerkin method
- For Eulerian equations B has to be modified
- Modifications account for convection, velocity-pressure coupling, shocks
- Weak formulation amenable to multiscale decomposition $\mathcal{V} = \mathcal{V}^h \otimes \mathcal{V}'$

$$B(\mathbf{W}^h, \mathbf{U}^h + \mathbf{U}') = (\mathbf{W}^h, \mathbf{F}) \qquad \mathbf{U} = \mathbf{U}^h + \mathbf{U}'$$

$$B(\mathbf{W}', \mathbf{U}^h + \mathbf{U}') = (\mathbf{W}', \mathbf{F}) \qquad \mathbf{W} = \mathbf{W}^h + \mathbf{W}'$$

RBVMS Formulation for the Incompressible Navier-Stokes Equations

Find $\mathbf{u}^h \in \mathcal{S}_u^h$ and $p^h \in \mathcal{S}_p^h$, such that $\forall \mathbf{w}^h \in \mathcal{V}_u^h$ and $q^h \in \mathcal{V}_p^h$:

$$
\underbrace{\int_\Omega \mathbf{w}^h \cdot \rho \left(\frac{\partial \mathbf{u}^h}{\partial t} + \mathbf{u}^h \cdot \nabla \mathbf{u}^h - \mathbf{f}^h \right) \, d\Omega + \int_\Omega \boldsymbol{\varepsilon}\left(\mathbf{w}^h\right) : \boldsymbol{\sigma}\left(\mathbf{u}^h, p^h\right) \, d\Omega - \int_{\Gamma_h} \mathbf{w}^h \cdot \mathbf{h}^h \, d\Gamma + \int_\Omega q^h \nabla \cdot \mathbf{u}^h \, d\Omega}_{\text{Galerkin}}
$$

$$
\underbrace{\underbrace{+ \sum_{e=1}^{n_{el}} \int_{\Omega^e} \tau_{\text{SUPS}} \left(\mathbf{u}^h \cdot \nabla \mathbf{w}^h + \frac{\nabla q^h}{\rho} \right) \cdot \mathbf{r}_{\text{M}}\left(\mathbf{u}^h, p^h\right) \, d\Omega + \sum_{e=1}^{n_{el}} \int_{\Omega^e} \rho \nu_{\text{LSIC}} \nabla \cdot \mathbf{w}^h r_{\text{C}}(\mathbf{u}^h, p^h) \, d\Omega}_{\text{SUPG/PSPG (SUPS)}}
- \sum_{e=1}^{n_{el}} \int_{\Omega^e} \tau_{\text{SUPS}} \mathbf{w}^h \cdot \left(\mathbf{r}_{\text{M}}\left(\mathbf{u}^h, p^h\right) \cdot \nabla \mathbf{u}^h \right) \, d\Omega
- \sum_{e=1}^{n_{el}} \int_{\Omega^e} \frac{\nabla \mathbf{w}^h}{\rho} : \left(\tau_{\text{SUPS}} \mathbf{r}_{\text{M}}\left(\mathbf{u}^h, p^h\right) \right) \otimes \left(\tau_{\text{SUPS}} \mathbf{r}_{\text{M}}\left(\mathbf{u}^h, p^h\right) \right) \, d\Omega}_{\text{Multiscale}} = 0
$$

ALE-RBVMS Formulation for the Incompressible Navier-Stokes Equations

Find $\mathbf{u}^h \in \mathcal{S}_u^h$ and $p^h \in \mathcal{S}_p^h$, such that $\forall \mathbf{w}^h \in \mathcal{V}_u^h$ and $q^h \in \mathcal{V}_p^h$:

Galerkin

$$\int_{\Omega_t} \mathbf{w}^h \cdot \rho \left(\left. \frac{\partial \mathbf{u}^h}{\partial t} \right|_{\hat{x}} + \left(\mathbf{u}^h - \hat{\mathbf{u}}^h\right) \cdot \boldsymbol{\nabla} \mathbf{u}^h - \mathbf{f}^h \right) \mathrm{d}\Omega + \int_{\Omega_t} \boldsymbol{\varepsilon}\left(\mathbf{w}^h\right) : \boldsymbol{\sigma}\left(\mathbf{u}^h, p^h\right) \mathrm{d}\Omega$$
$$- \int_{(\Gamma_t)_\mathrm{h}} \mathbf{w}^h \cdot \mathbf{h}^h \, \mathrm{d}\Gamma + \int_{\Omega_t} q^h \boldsymbol{\nabla} \cdot \mathbf{u}^h \, \mathrm{d}\Omega$$

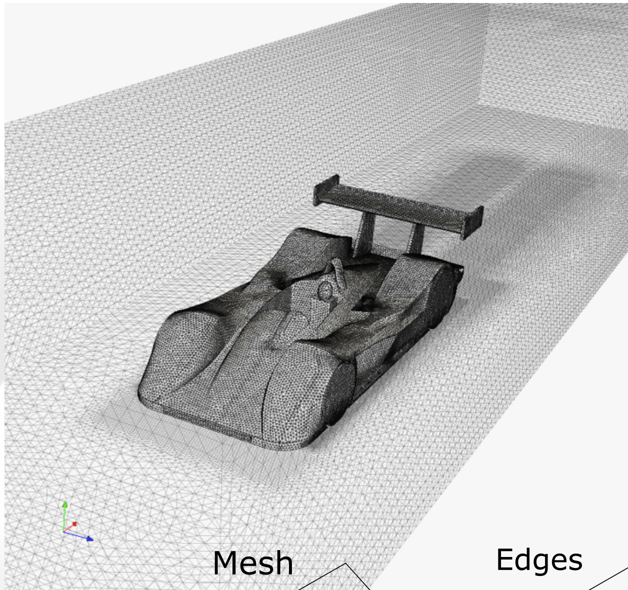
Multiscale

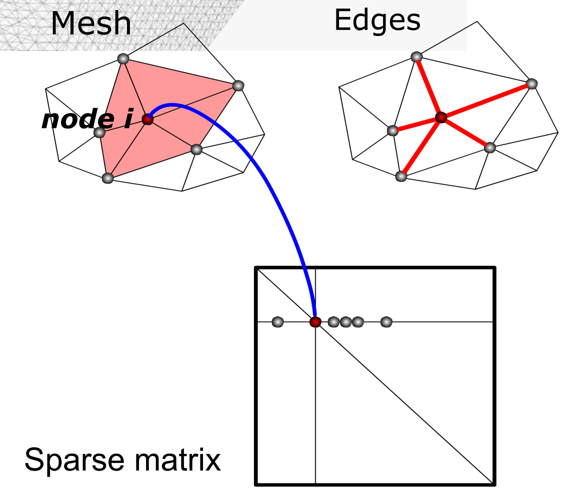
SUPG/PSPG (SUPS)

$$+ \sum_{e=1}^{n_\mathrm{el}} \int_{\Omega_t^e} \tau_\mathrm{SUPS} \left(\left(\mathbf{u}^h - \hat{\mathbf{u}}^h\right) \cdot \boldsymbol{\nabla} \mathbf{w}^h + \frac{\boldsymbol{\nabla} q^h}{\rho} \right) \cdot \mathbf{r}_\mathrm{M}\left(\mathbf{u}^h, p^h\right) \mathrm{d}\Omega$$
$$+ \sum_{e=1}^{n_\mathrm{el}} \int_{\Omega_t^e} \rho \nu_\mathrm{LSIC} \boldsymbol{\nabla} \cdot \mathbf{w}^h r_\mathrm{C}\left(\mathbf{u}^h, p^h\right) \mathrm{d}\Omega$$
$$- \sum_{e=1}^{n_\mathrm{el}} \int_{\Omega_t^e} \tau_\mathrm{SUPS} \mathbf{w}^h \cdot \left(\mathbf{r}_\mathrm{M}\left(\mathbf{u}^h, p^h\right) \cdot \boldsymbol{\nabla} \mathbf{u}^h \right) \mathrm{d}\Omega$$
$$- \sum_{e=1}^{n_\mathrm{el}} \int_{\Omega_t^e} \frac{\boldsymbol{\nabla} \mathbf{w}^h}{\rho} : \left(\tau_\mathrm{SUPS} \mathbf{r}_\mathrm{M}\left(\mathbf{u}^h, p^h\right) \right) \otimes \left(\tau_\mathrm{SUPS} \mathbf{r}_\mathrm{M}\left(\mathbf{u}^h, p^h\right) \right) \mathrm{d}\Omega = 0$$

Finite Element Method

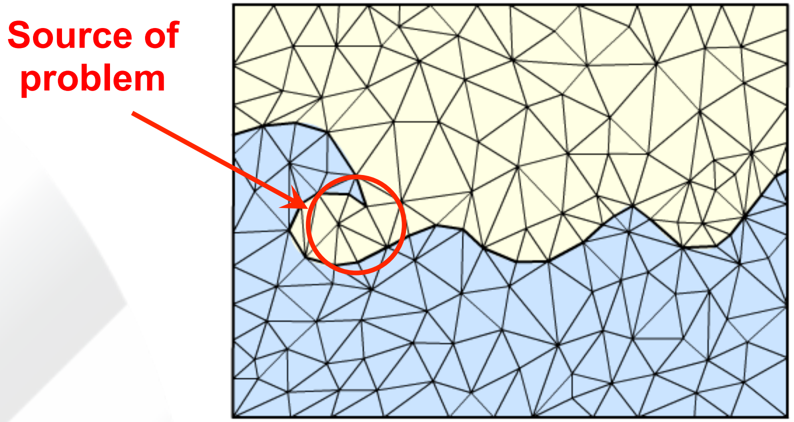
- **Unstructured grid method characterized by:**
  - Discontinuous data
  - Gather/scatter operations
  - Random memory access
  - Data dependencies
- **Main Computational Kernels for Implicit Time Marching**
  - Forming system matrix and RHS
  - Solving linearized systems by preconditioned Krylov solvers

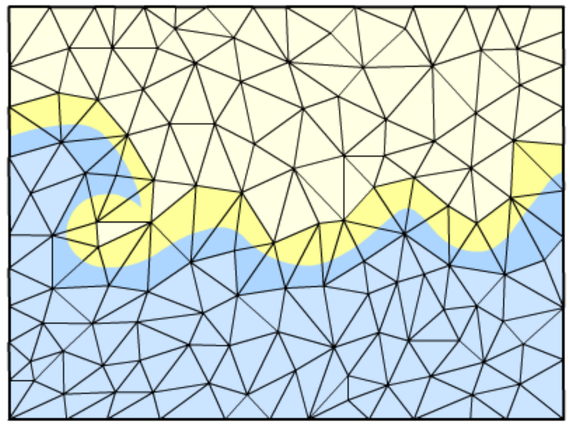




Mesh-based Solution Techniques for Moving Boundaries

- **Interface Tracking (ALE)**
  - Accurate
  - Coarse mesh computation
  - Small deformation
  - Mesh moving
  - Mesh distortion/remeshing
- **Interface Capturing (VOF, LS, PF)**
  - Mesh fixed
  - No remesh needed
  - Flexible
  - Large Deformation, breaks and cusps
  - Additional equation
  - Refined mesh





Scalar Transport

$$\frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \boldsymbol{\nabla} \phi - \boldsymbol{\nabla} \cdot (\nu \boldsymbol{\nabla} \phi) = 0 \text{ on } \Omega$$

Boundary conditions

$$\begin{aligned} \phi &= g \qquad \text{on } \Gamma_g \\ \mathbf{n} \cdot \nu \boldsymbol{\nabla} \phi &= h \qquad \text{on } \Gamma_h \end{aligned}$$

ϕ is a scalar being transported (marker function: VoF, LS, Phase-Field)
u is a given velocity field
ν is the diffusion coefficient
VoF and LS: pure advection for marker function

Fluid properties:

$$\boxed{\begin{aligned} \rho &= \rho_0 + (\rho_1 - \rho_0)\mathbb{H}(\phi) \\ \mu &= \mu_0 + (\mu_1 - \mu_0)\mathbb{H}(\phi) \end{aligned}}$$

FEM for Transport

$$\int_\Omega w^h \left(\frac{\partial \phi^h}{\partial t} + \mathbf{u}^h \cdot \boldsymbol{\nabla} \phi^h \right) d\Omega + \int_\Omega \boldsymbol{\nabla} w^h \cdot \nu \boldsymbol{\nabla} \phi^h d\Omega$$

$$+ \sum_{e=1}^{n_{el}} \int_{\Omega^e} \tau_{\text{SUPG}} \mathbf{u}^h \cdot \boldsymbol{\nabla} w^h \left(\frac{\partial \phi^h}{\partial t} + \mathbf{u}^h \cdot \boldsymbol{\nabla} \phi^h - \boldsymbol{\nabla} \cdot \left(\nu \boldsymbol{\nabla} \phi^h \right) \right) d\Omega$$

$$= \int_{\Gamma_h} w^h h d\Gamma$$

SUPG stabilization

Stabilization prevents spurious oscillations in convection dominated flows

Discontinuity capturing for transport equation

CAU and YZβ discontinuity capturing

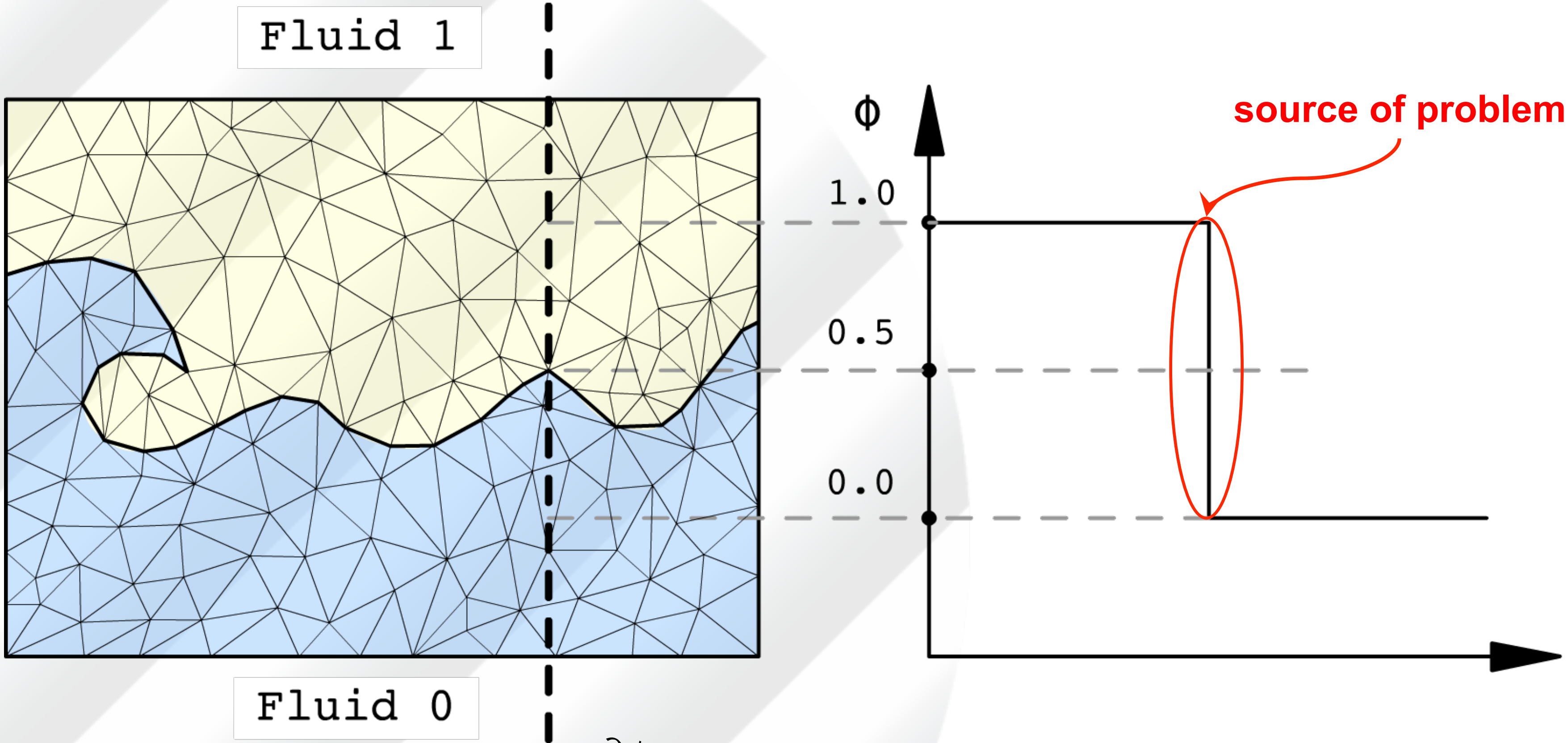
From Bazilevs et al, IJNMF, 2007

$$\delta = \frac{h_e}{2} \frac{\left| r\left(\phi^h\right) \right|}{\left\| \nabla \phi^h \right\|} \quad ; \quad \delta^{YZ\beta} = \left(\frac{h_e}{2} \right)^{\beta} \left| \hat{\phi} r\left(\phi^h\right) \right| \left(\sum_{i=1}^{3} \left| \left(\hat{\phi}\right)^{-1} \frac{\partial \phi^h}{\partial x_i} \right|^2 \right)^{\frac{\beta}{2}-1}$$

From Galeão and do Carmo CMAME, 1988

$$\boxed{\beta = 1 \quad \text{and} \quad \hat{c}_i = 1 \Rightarrow \text{CAU} \equiv \text{YZ}\beta}$$

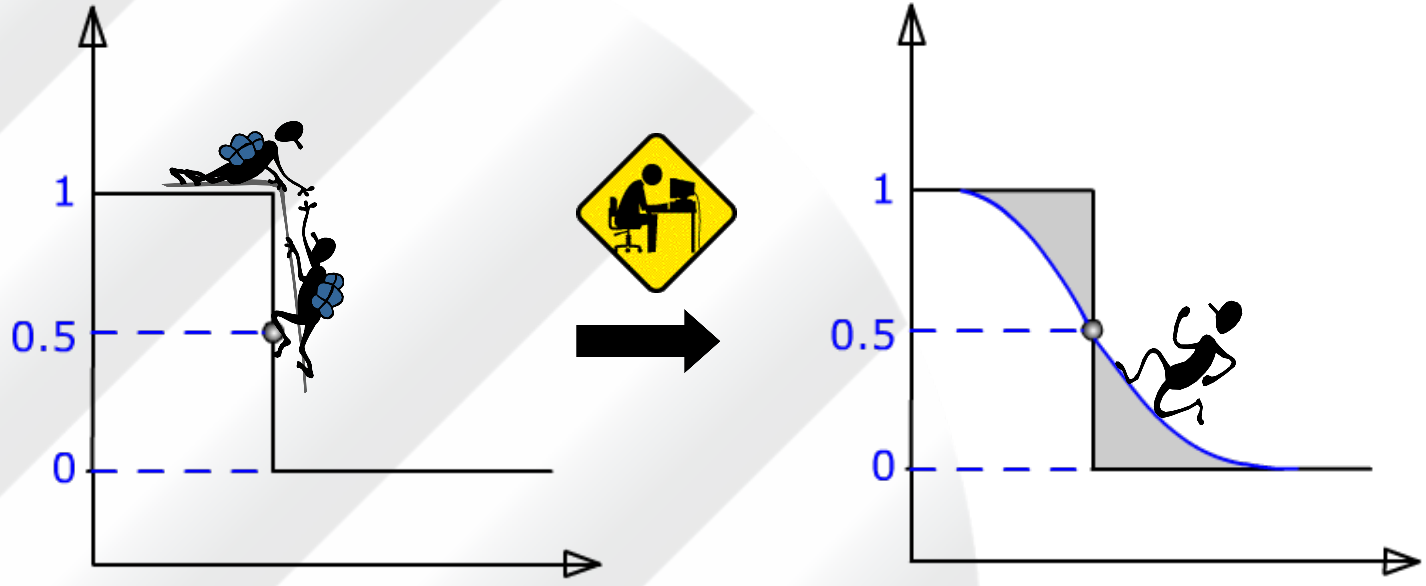
Interface capturing: VOF or Pseudo-Concentration



$$\frac{\partial \phi}{\partial t} + \mathbf{u}.\nabla \phi = 0$$

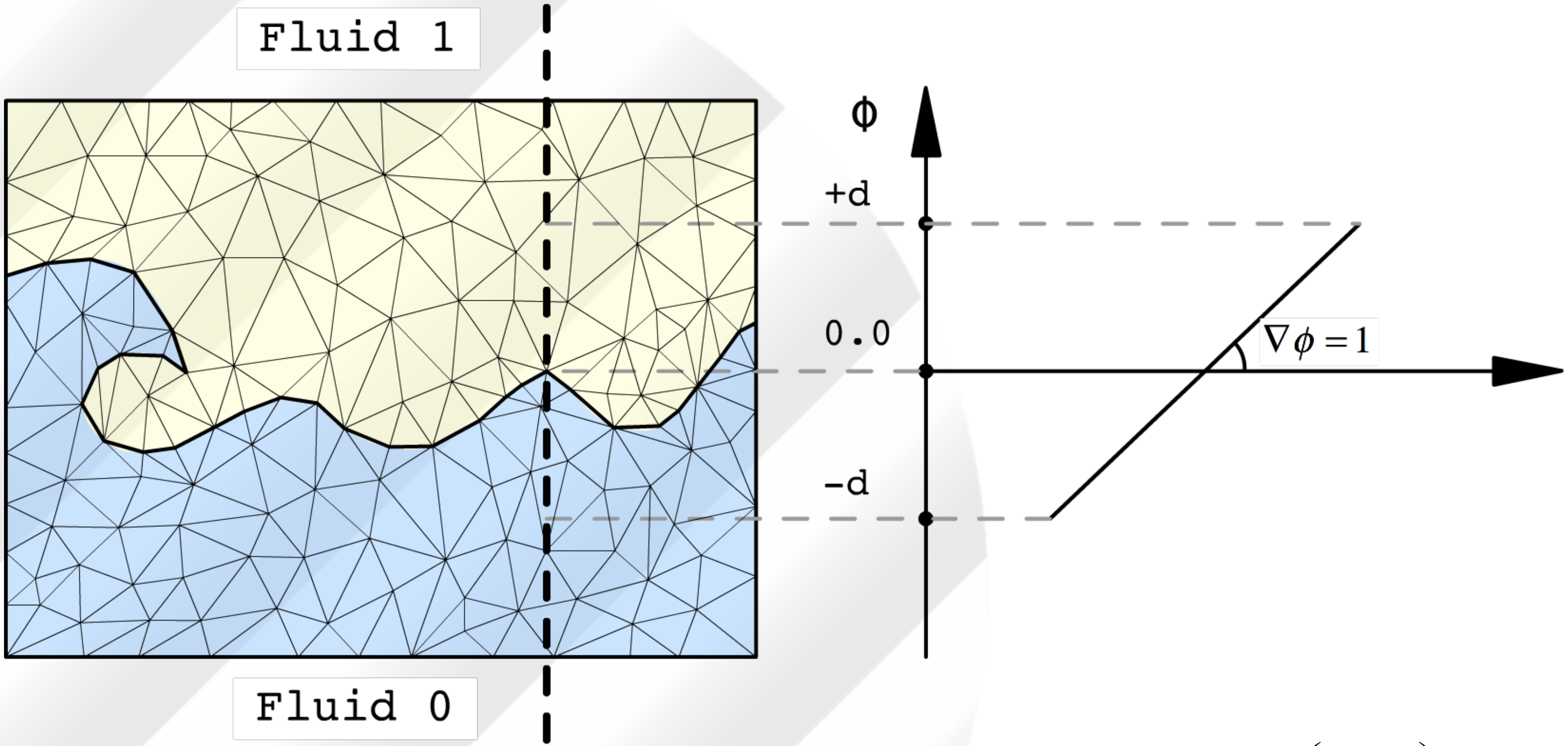
from Navier-Stokes

VOF Challenges



- Interface tends to smooth according to the marking function transport
  - Stabilized formulation helps to diminish this effect → **but does not completely solve**
- **Surface tension is not easy to be modeled in VOF methods since it depends on the interface curvature**

Interface capturing: Level Set



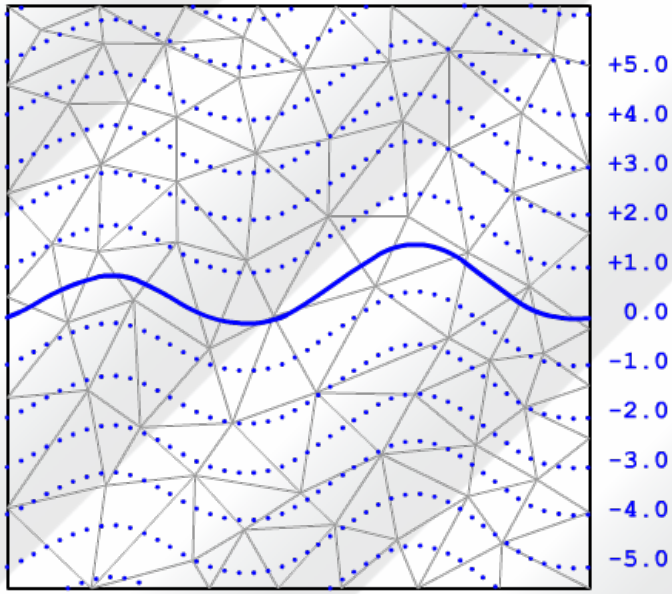
$$\frac{\partial\phi}{\partial t} + \mathbf{u}.\nabla\phi = 0$$

from Navier-Stokes

Interface Curvature: $\kappa = \nabla.\left(\frac{\nabla\phi}{\|\nabla\phi\|}\right)$

Level set challenges

- Computing curvature is not that difficult
- Computing and keeping sign distance functions in unstructured grids is not easy! Redistancing is necessary.
- **Is possible to avoid redistancing by solving directly Eikonal's equation**



We propose a new method based on the Fast Marching Method concepts for the Eikonal equation.

$$\left\|\nabla \phi\right\| = 1$$

More details in: Renato N. Elias, Marcos A. D. Martins, Alvaro L. G. A. Coutinho, *Simple finite element based computation of distance functions in unstructured grids*, IJNME, Volume 72(9): 1095–1110, 2007

Computing signed distance functions in TET4 meshes

- **We impose the satisfaction of Eikonal´s equation at element level, using FEM data structures: algorithm *O*(*nnodes*)**

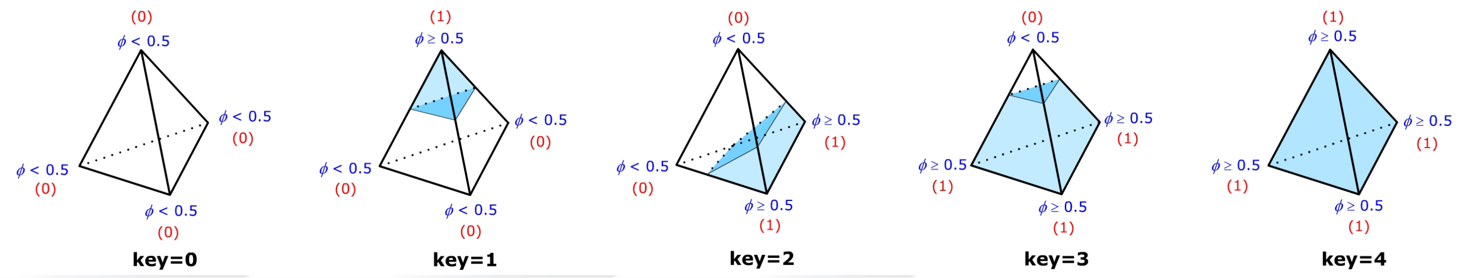
$$\left\|\nabla\phi^{e}\right\| = \left\| \frac{1}{6V} \begin{bmatrix} N_{1,x} & N_{2,x} & N_{3,x} & N_{4,x} \\ N_{1,y} & N_{2,y} & N_{3,y} & N_{4,y} \\ N_{1,z} & N_{2,z} & N_{3,z} & N_{4,z} \end{bmatrix} \begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \end{bmatrix} \right\| = 1$$

- **Small errors (<1%) in a narrow band**
- **Alternative to Ray Tracing in Tomography**

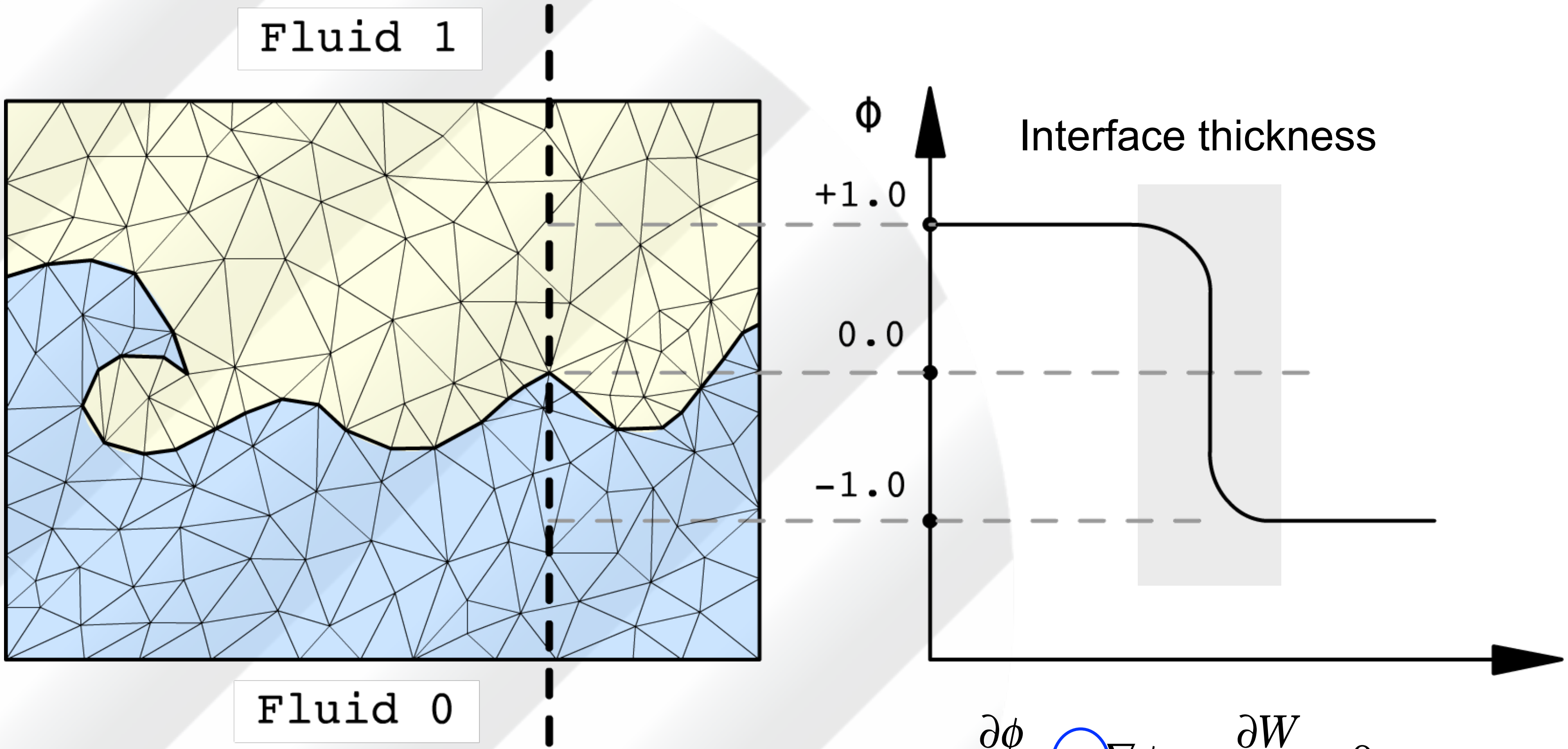
Enforcing Mass Conservation on VoF and LS

- mass lost/gained are found comparing the initial mass plus the inlet and outlet fluxes, at the end of each time step.
- values to be added or removed are made proportional to the absolute value of the normal velocity of the interface given by

$$u_n = \left| \mathbf{u} \cdot \frac{\nabla\phi}{\left\|\nabla\phi\right\|} \right|$$



Interface capturing: Phase Field (Diffuse Interface)



$$\frac{\partial \phi}{\partial t} + \mathbf{u}.\nabla \phi - \gamma \frac{\partial W}{\partial \phi} = 0$$

from Navier-Stokes

W is a total free energy providing the interface dissipation mechanism

Smooth transition region of length η between two bulk phases

General Coupling Scheme

Rigid Body Motion

Newton Euler equations

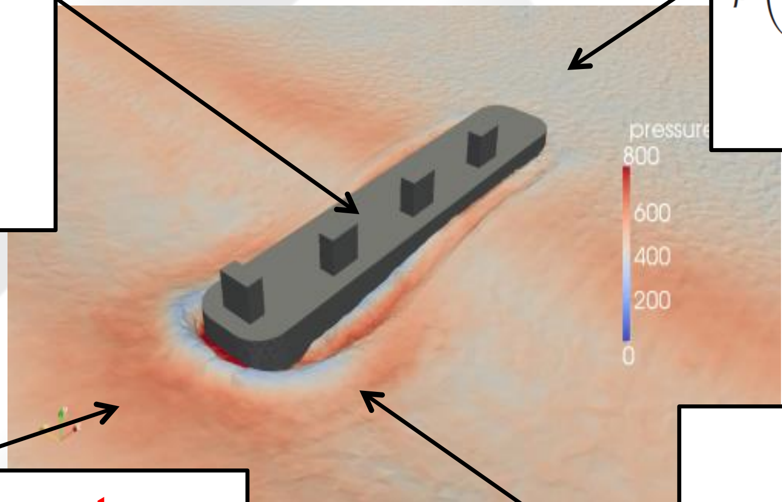
$$\begin{aligned} m\mathbf{a} &= \mathbf{f}_{tot} \\ \mathbf{J}\,\dot{\boldsymbol{\Omega}} + \boldsymbol{\Omega} \times \mathbf{J}\,\boldsymbol{\Omega} &= \mathbf{m} \end{aligned}$$

Fluid flow

NS + Incompressibility

$$\rho\left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} - \mathbf{u}_{mesh}) \cdot \nabla \mathbf{u} - \mathbf{b}\right) - \nabla \cdot \sigma = \mathbf{0},$$

$$\nabla \cdot \mathbf{u} = 0.$$



Mesh Movement

BVP

$$\mathbf{u}_{mesh} = \frac{\mathbf{d}(\mathbf{x}, t)}{\Delta t}$$

$$\nabla \cdot ([1 + \tau]\nabla)\, d_i = 0$$

Interface capturing

VOF/LS Method

$$\frac{\partial \phi}{\partial t} + \nabla \cdot ((\mathbf{u} - \mathbf{u}_{mesh})\,\phi) = 0$$

$$\rho = \phi(\mathbf{x}, t)\rho_A + [1 - \phi(\mathbf{x}, t)]\,\rho_B$$

Coupling Scheme: Overview

Time Loop

Given FLUID $(\mathbf{u}, \mathbf{u}_{\text{mesh}}, \mathrm{P}, \phi)_{t_n}$

Rigid Body $(\mathbf{x}, \theta)_{t_n}$

Compute the FORCE / MOMENT

$(\mathbf{F}, \mathbf{M})_{t_n}$ → Informations needed

Compute BODY MOTION

$(\mathbf{x}, \theta)_{t_{n+1}}$

SOLVE SOLID PART

Compute Mesh velocity BVP $\mathbf{u}_{\text{mesh}} \; t_{n+1}$

SOLVE TRANSPORT PART

Compute Fluid Variables Navier-Stokes $(\mathbf{u}, \mathrm{P})_{t_{n+1}}$

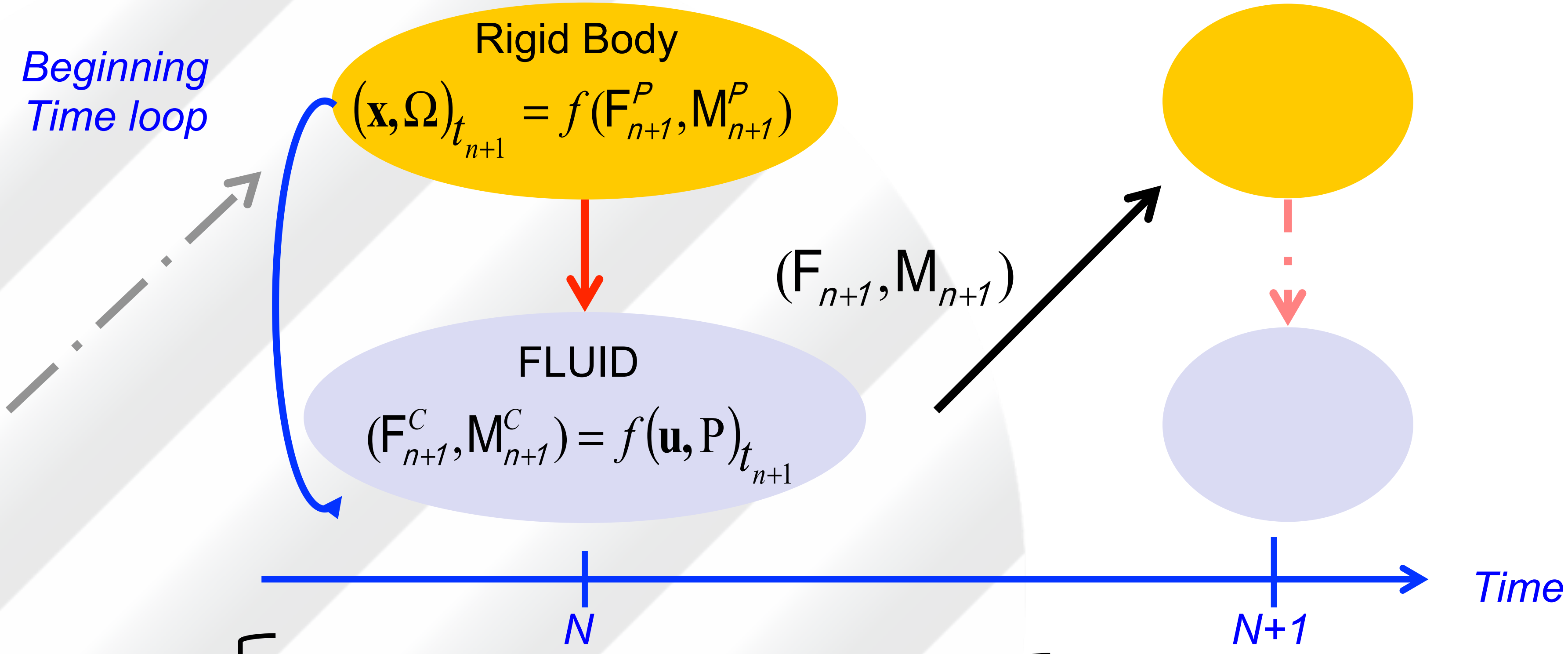
SOLVE FLUID PART

Compute Marking function $\phi_{t_{n+1}}$

VOF/LS

n ← **n+1**

Coupling Scheme: Staggered - Predictor/Corrector



Predictor

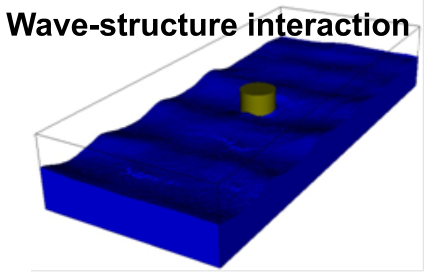
Corrector

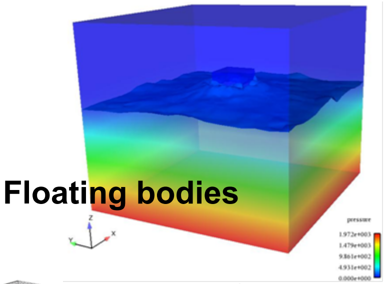
$$\begin{cases} F^P_{n+1} = (1+\tau_P)F_n - \tau_P F_{n-1} \\ F_{n+1} = \tau_C F^C_{n+1} + (1-\tau_C)\, F^P_{n+1} \end{cases} \qquad \begin{cases} \tau_P = 1 \\ \tau_C = 1/2 \end{cases}$$

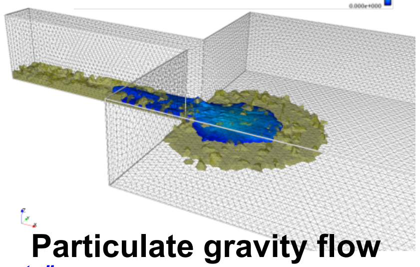
Ref.: *A New Staggered Scheme for Fluid- Structure Interaction – 2013* Dettmer, Peric (Int. J. for Num. Meth. In Eng.)

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/Compressible Flow:**
  - SUPG/PSPG/LSIC stabilized finite element method in Eulerian or ALE frames
  - Fully coupled **u**-*p* system (4-dofs/5-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"*







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

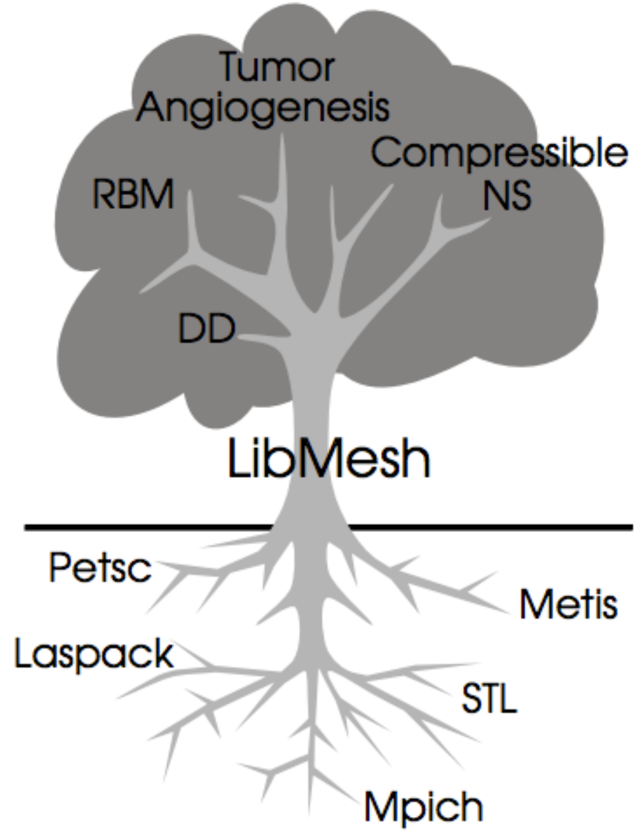
Blue: *"Home made" code*
Green: *Third party code*

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

libMesh

- High level interface for finite element analysis (Kirk et al., Eng. with Computers, 2006)
- Support adaptive mesh refinement and coarsening (AMR/C)
- 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:

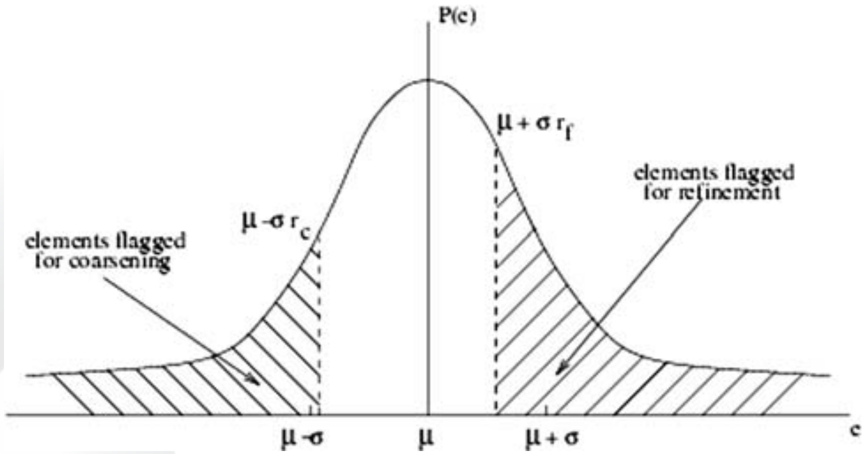
http://libmesh.github.io/



libMesh interfaces

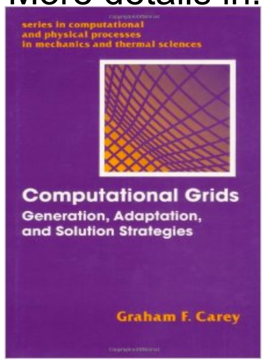
AMR/C

- libMesh utilizes a statistical scheme (Kirk et al., Eng. with Computers, 2006)
  - 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

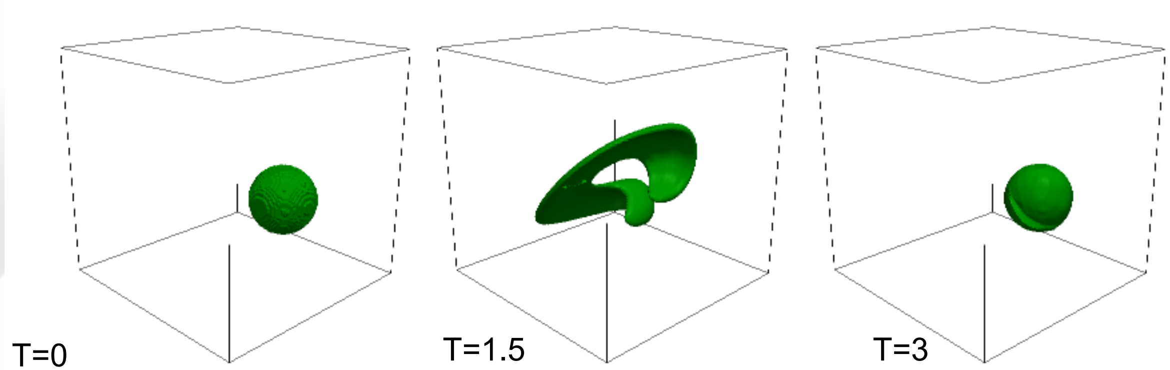


Probability density function. Kirk et al., 2006

More details in:

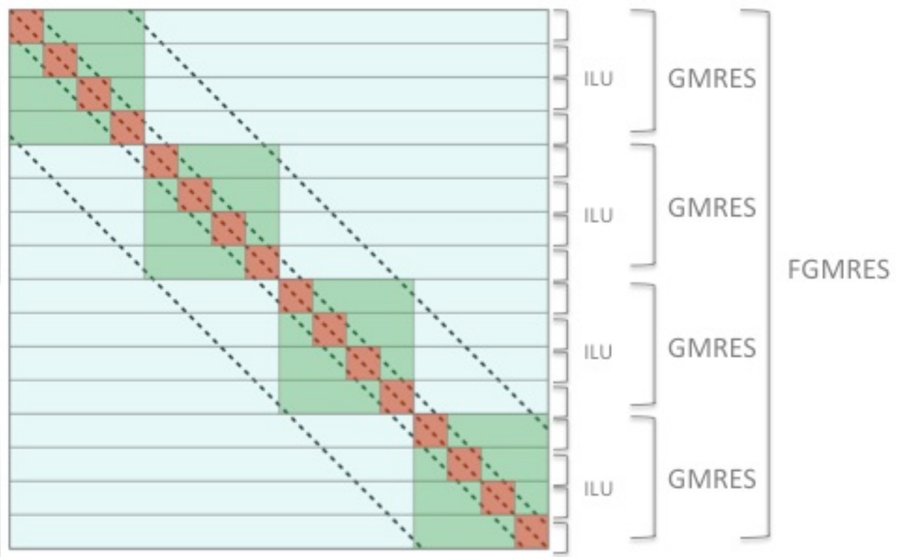


Three-dimensional deformation problem

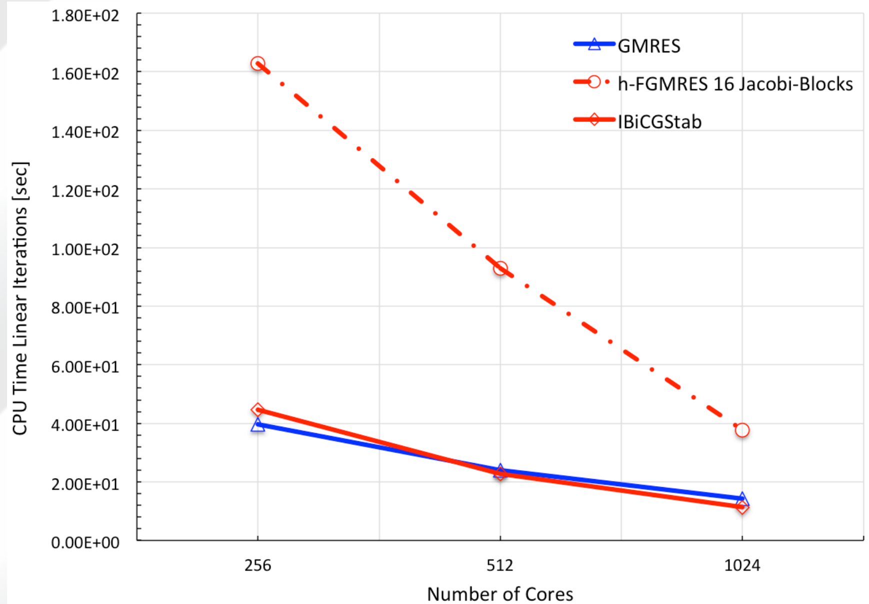


$$u_x = 2sin^2(\pi x)sin(2\pi y)sin(2\pi z)g(t)$$
$$u_y = -sin(2\pi x)sin^2(\pi y)sin(2\pi z)g(t)$$
$$u_z = -sin(2\pi x)sin(2\pi y)sin^m 2(\pi z)g(t)$$

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.

Fixed FPSO Model Test

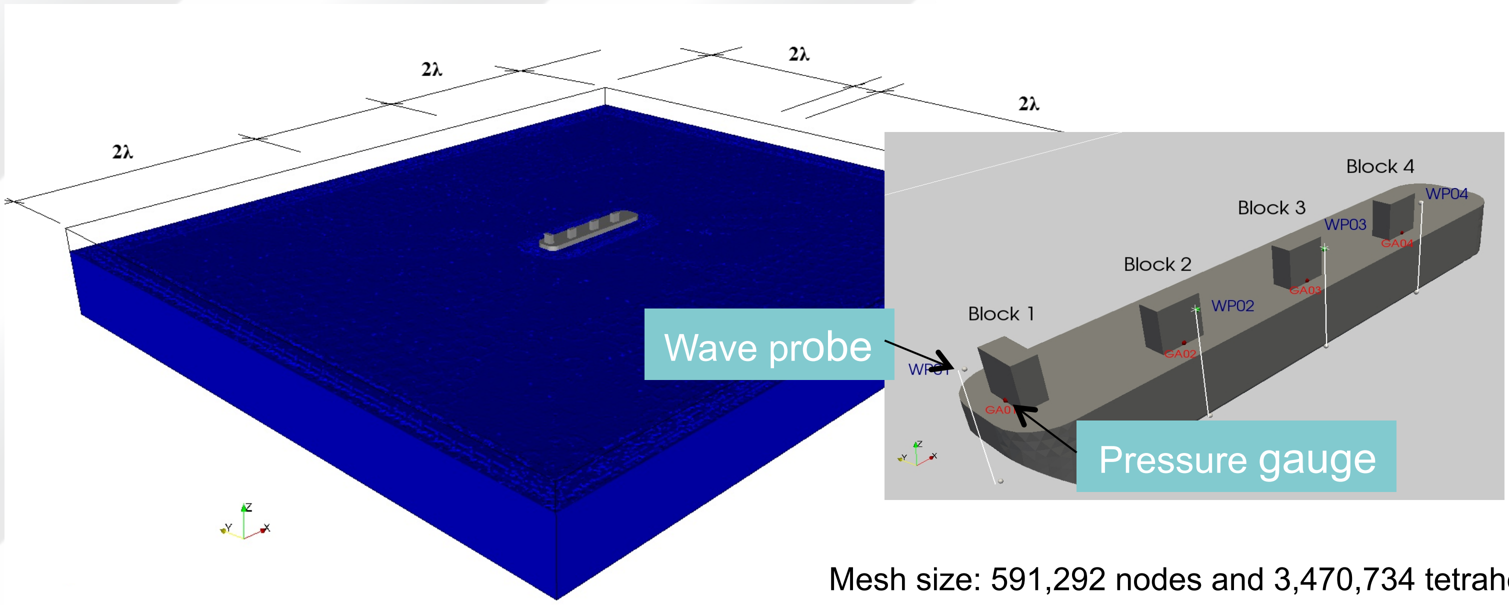
WAVE	TYPE	H_s (m)	T_p (s)	ST	OBSERVATION
Regular 1	Regular	10.50	17.50	0.022	Low H_s, with the **roll natural period**
Regular 2	Regular	12.50	17.50	0.026	Medium H_s, with the **roll natural period**
Regular 3	Regular	15.00	17.50	0.031	High H_s, with the **roll natural period**
Regular 4	Regular	12.50	15.00	0.036	Medium H_s, with the period **under the roll natural period and above the pitch/heave natural period**
Regular 5	Regular	12.50	20.00	0.020	Medium H_s, with the period **above the roll natural period**
Regular 6	Regular	12.50	12.50	0.051	Medium H_s, with the **pitch/heave natural period**
Regular 7	Regular	10.50	12.50	0.043	Low H_s, with the **pitch/heave natural period**
Regular 8	Regular	15.00	12.50	0.061	High H_s, with the **pitch/heave natural period**
Regular 9	Regular	12.50	10.00	0.080	Medium H_s, with the period **under the pitch/heave natural period**
Irregular 1	Jonswap	7.97	13.50	0.028	Based on representative conditions of Santos Basin
Irregular 2	Jonswap	10.79	13.50	0.038	Based on representative conditions of Santos Basin
Irregular 3	Jonswap	6.1	9.0	0.048	Based on a preliminary WAMIT study

- ***Pure waves***
- ***Steepness range (2 to 8%)***
- ***Periods based on the natural periods***
- ***Irregular waves based on Santos Basin conditions***
- ***Transient waves (not shown on the table)***

Fixed FPSO

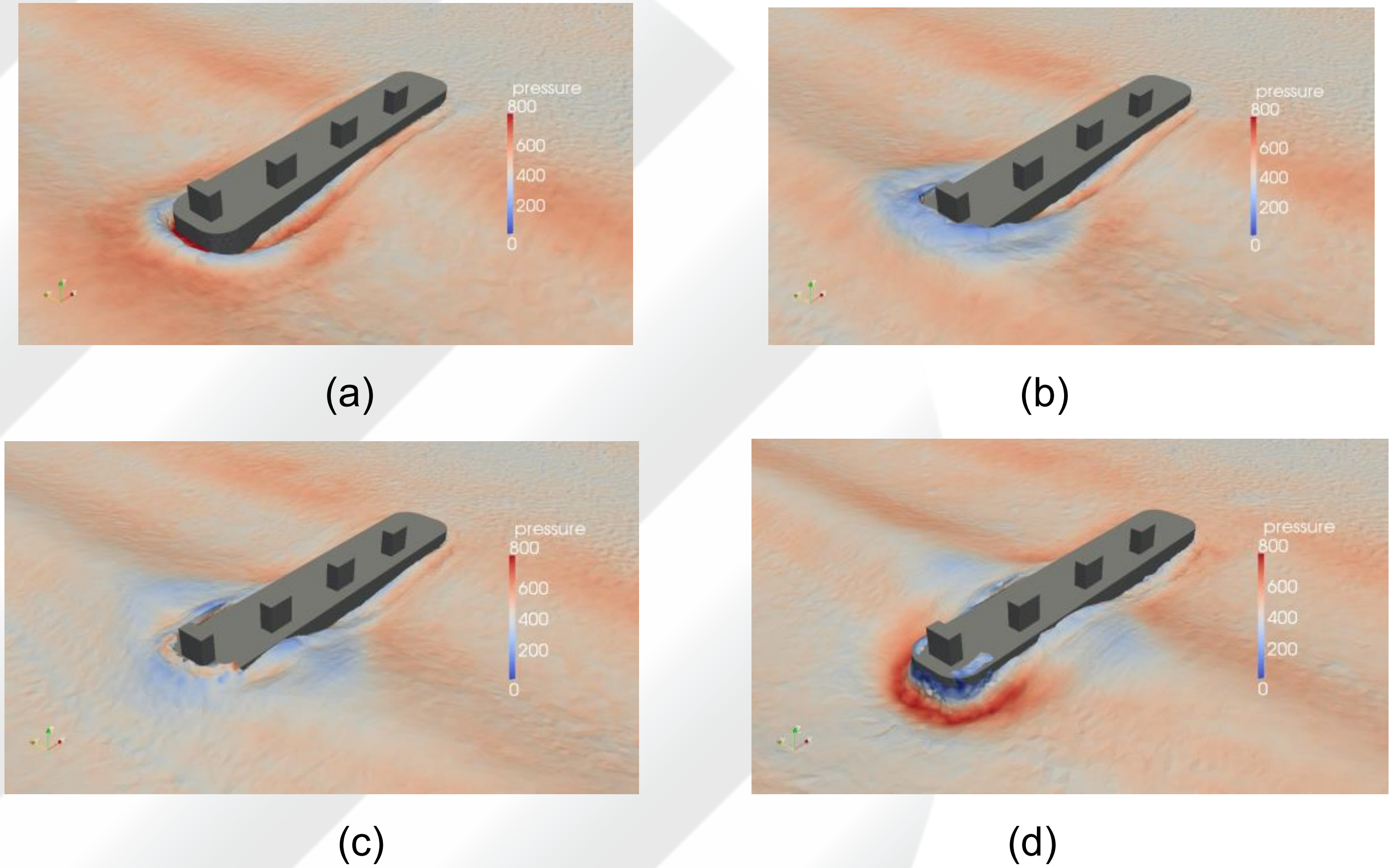
❑ Computational Domain (scale 1:100)

- Tank parameterized from average wave length λ related to waves of period 12,5s (λ=244m) and 15,0s (λ=348m).
- Thus, the computational domain is suitable for simulations up to 15 seconds.



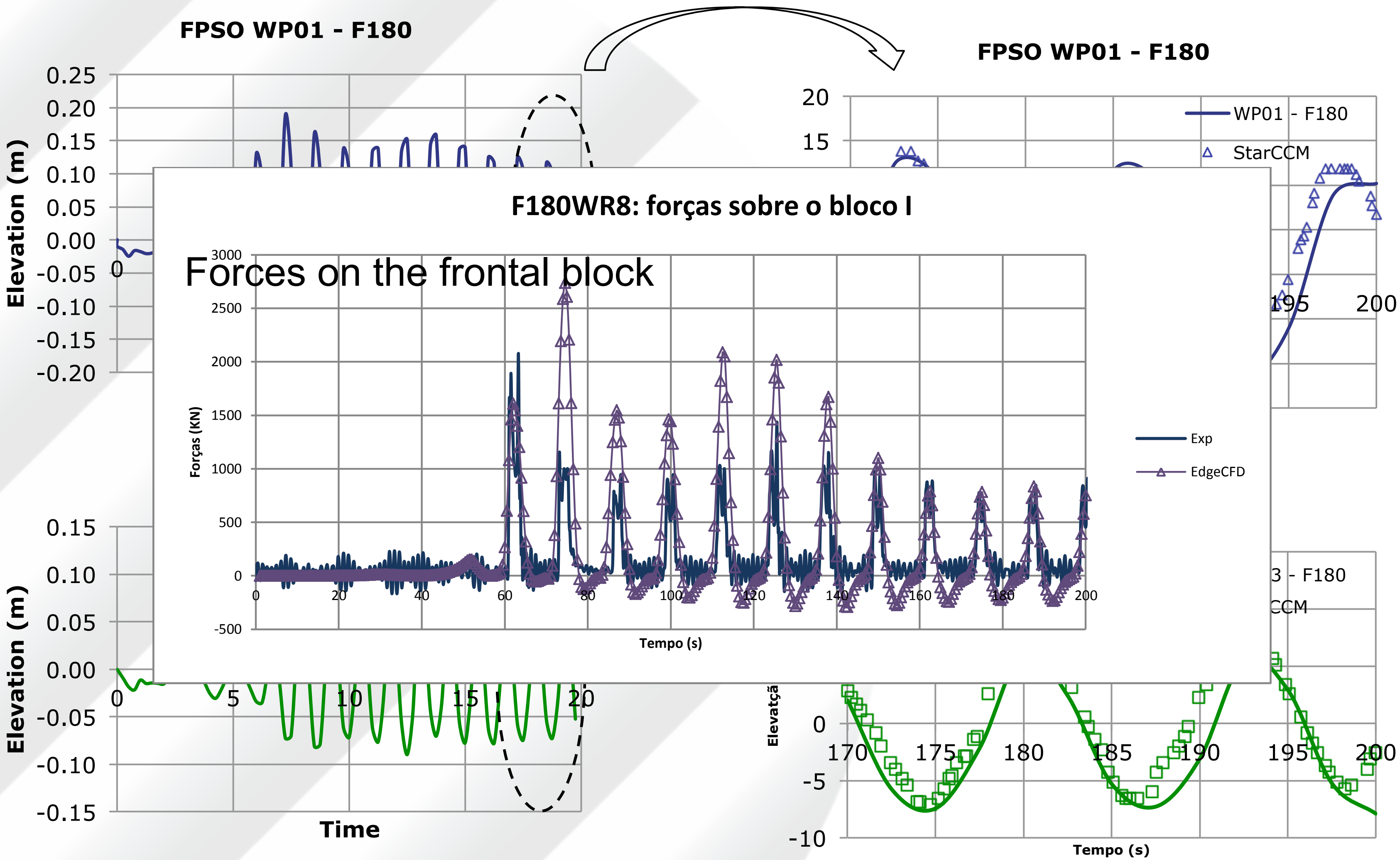
Mesh size: 591,292 nodes and 3,470,734 tetrahedra

Fixed FPSO / Frontal Wave R8



Pressure and wave elevation snapshots at (a) t=7.0s (b) t=7.25s (c) t=7.5s (d) t=7.75s

Fixed FPSO / Frontal WR8

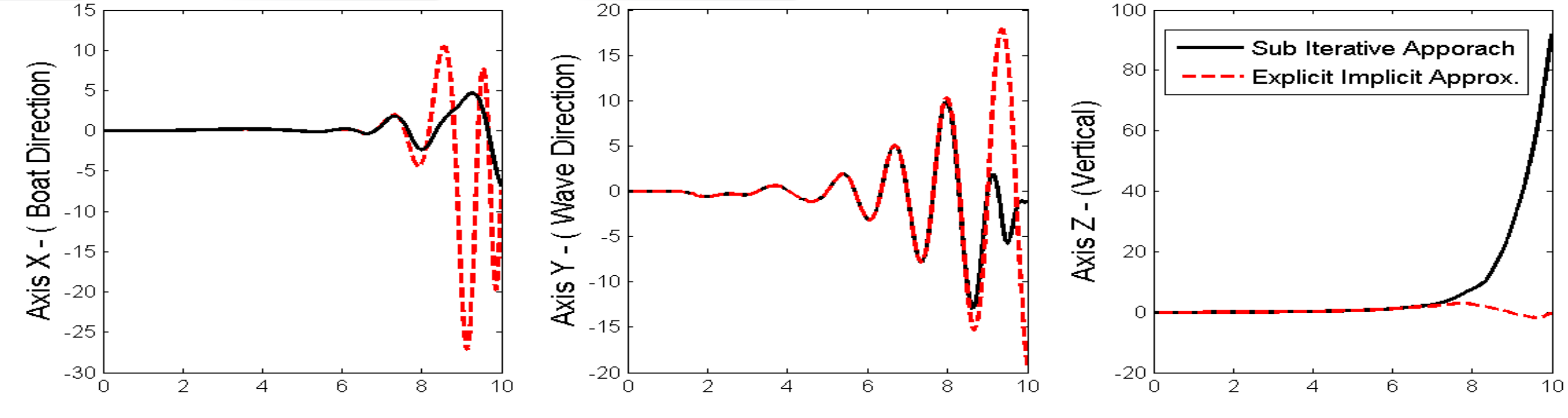


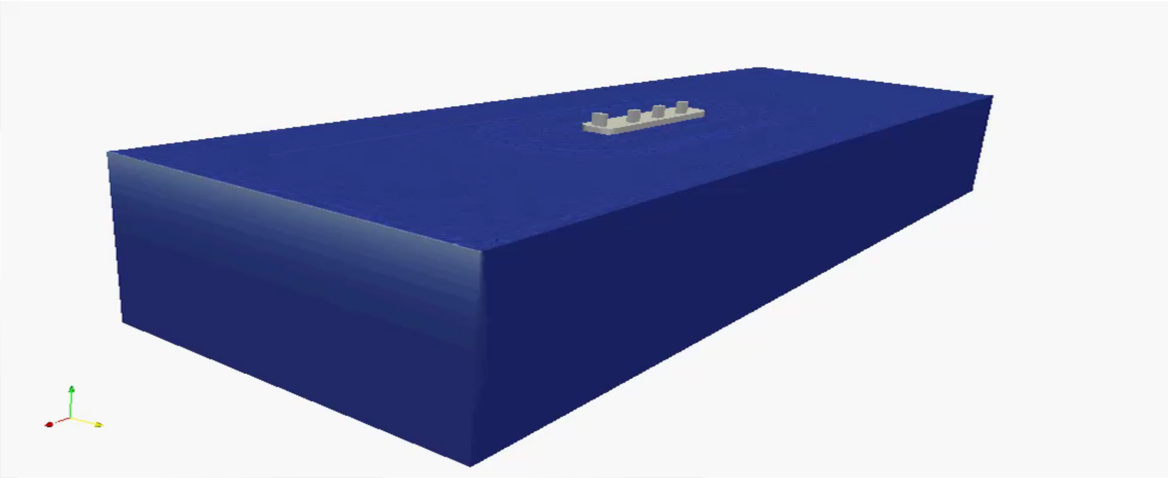
STARCCM Mesh is 4x finer

Test Case: Ship Simulation

Simulation 4 d.o.f.: 3 rotations 1 vertical translation

Comparison between **Subiterative** and **Explicit-Implicit** Schemes





Allen-Cahn Phase Field Model

- **Total free energy in terms phase field variable** $W(\varphi)$
  - Double-well function

$$W = \int_{\Omega} \frac{1}{2}|\nabla\phi|^2 + \frac{1}{4\eta^2} f(\phi)\, d\Omega$$

$$\phi(\mathbf{x},t) \in [-1,1]$$

- **Allen-Cahn model is derived by assuming that the dissipation effect is described through a gradient flow mechanism**
  - interfacial thickness
  - elastic relaxation time

$$\frac{\partial\phi}{\partial t} + \mathbf{u}\cdot\nabla\phi = -\gamma\frac{\partial W}{\partial\phi} = -\gamma\left(\nabla^2\phi + \frac{1}{4\eta^2}F(\phi)\right)$$

Conservative Allen-Cahn Phase Field Model

- Lagrange multiplier related to the constant volume constraint

$$\frac{d}{dt}\int_{\Omega}\phi\, d\Omega = 0$$

$$\frac{\partial\phi}{\partial t} + \mathbf{u}\cdot\nabla\phi = -\gamma\left(\frac{\partial W}{\partial\phi} + \xi(t)\right)$$

$$\frac{\partial\phi}{\partial t} + \mathbf{u}\cdot\nabla\phi = -\gamma\left(\nabla^2\phi + \frac{1}{4\eta^2}f\left(\phi\right) + \xi(t)\right)$$

Allen-Cahn Navier-Stokes Model

- **Navier-Stokes equation:**
  - Body force depending on the phase-field parameter
  - Surface tension too

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}\nabla\mathbf{u} + \nabla p - \frac{1}{Re}\nabla^2\mathbf{u} + \lambda\nabla\cdot\left(\nabla\phi\otimes\nabla\phi\right) = Ri\phi$$

$$\nabla\cdot\mathbf{u} = 0$$

- **Conservative Allen-Cahn equation:**
  - Lagrange multiplier corresponding to the constant volume constraint

$$\frac{\partial\phi}{\partial t} + \mathbf{u}\cdot\nabla\phi = \gamma\left(\nabla^2\phi - \frac{1}{4\eta^2}f\left(\phi\right) + \xi(t)\right)$$

$$\frac{\partial}{\partial t}\int_\Omega \phi\, d\Omega = 0$$

FEM for Allen-Cahn Phase Field Model

- **SUPG Finite Element Formulation**

$$\int_\Omega w^h \left(\frac{\partial \phi^h}{\partial t} + \mathbf{u}^h \cdot \nabla \phi^h \right) d\Omega + \int_\Omega \gamma \nabla w^h \cdot \nabla \phi^h d\Omega + \int_\Omega \gamma w^h (f^h - \xi^h) d\Omega$$
$$+ \sum_{e=1}^{n_{el}} \int_{\Omega^e} \tau_{SUPG}\, \mathbf{u}^h \cdot \nabla w^h \left(\frac{\partial \phi^h}{\partial t} + \mathbf{u}^h \cdot \nabla \phi^h - \gamma \left(\nabla^2 \phi^h - f^h + \xi^h \right) \right) d\Omega^e = 0$$

- **Residual-Based AC FEM Formulation**

$$AC_{DC} = \sum_{e=1}^{n_{el}} \int_{\Omega^e} \gamma \nabla w^h \cdot \nabla \phi^h d\Omega$$

$$\int_\Omega w^h \left(\frac{\partial \phi^h}{\partial t} + \mathbf{u}^h \cdot \nabla \phi^h \right) d\Omega + \sum_{e=1}^{n_{el}} \int_\Omega \gamma^e(\phi^h)(\nabla w^h \cdot \nabla \phi^h - w^h(-f^h + \xi^h)) d\Omega$$
$$+ \sum_{e=1}^{n_{el}} \int_{\Omega^e} \tau_{SUPG}\, \mathbf{u}^h \cdot \nabla w^h \left(\frac{\partial \phi^h}{\partial t} + \mathbf{u}^h \cdot \nabla \phi^h - \gamma^e(\phi^h)(\nabla^2 \phi^h - f^h + \xi^h) \right) d\Omega^e = 0$$

Vasconcelos, Rossa & Coutinho, IJNMF, 2014

Salt Tectonics with AC Phase-Field Model in libMesh

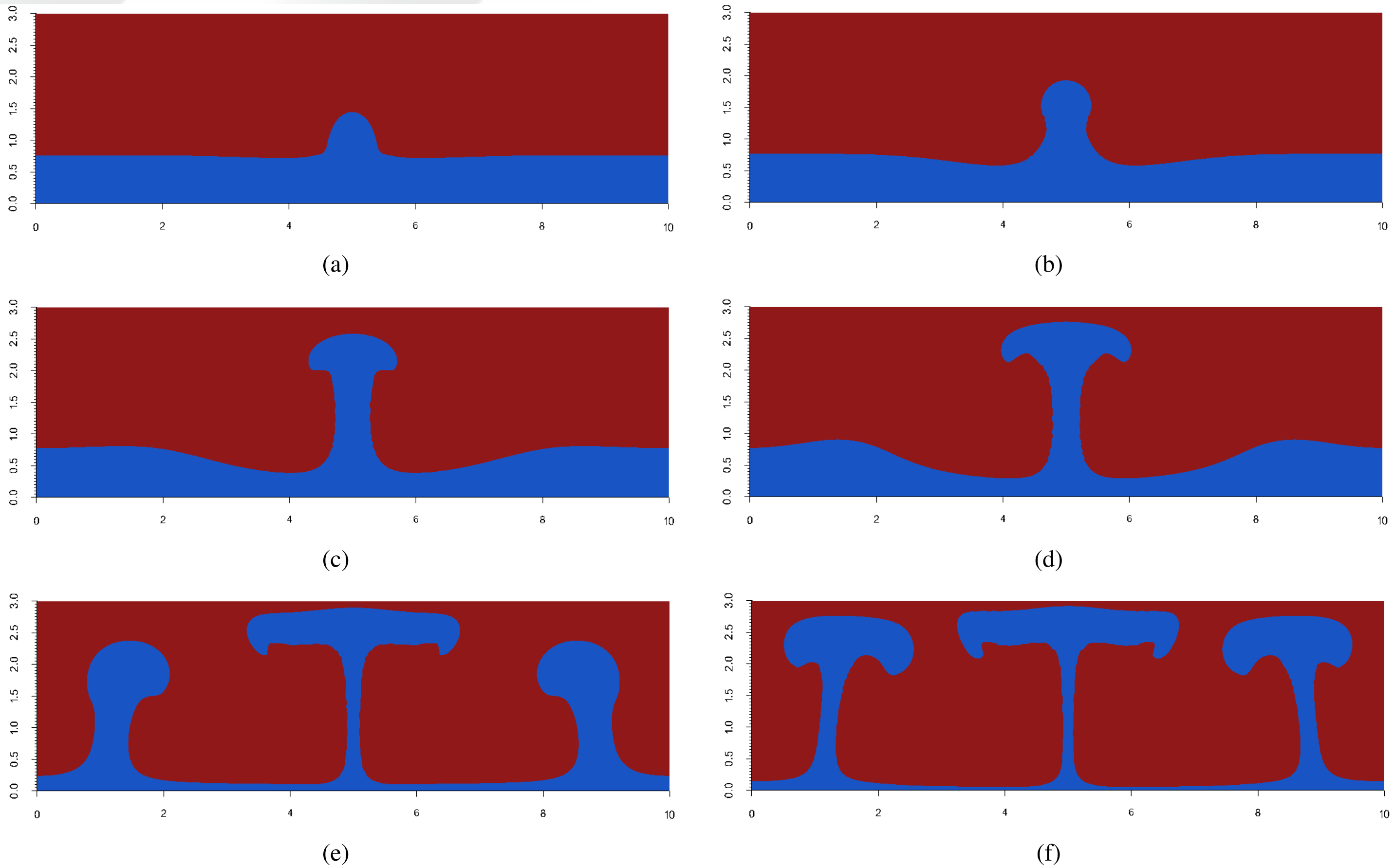


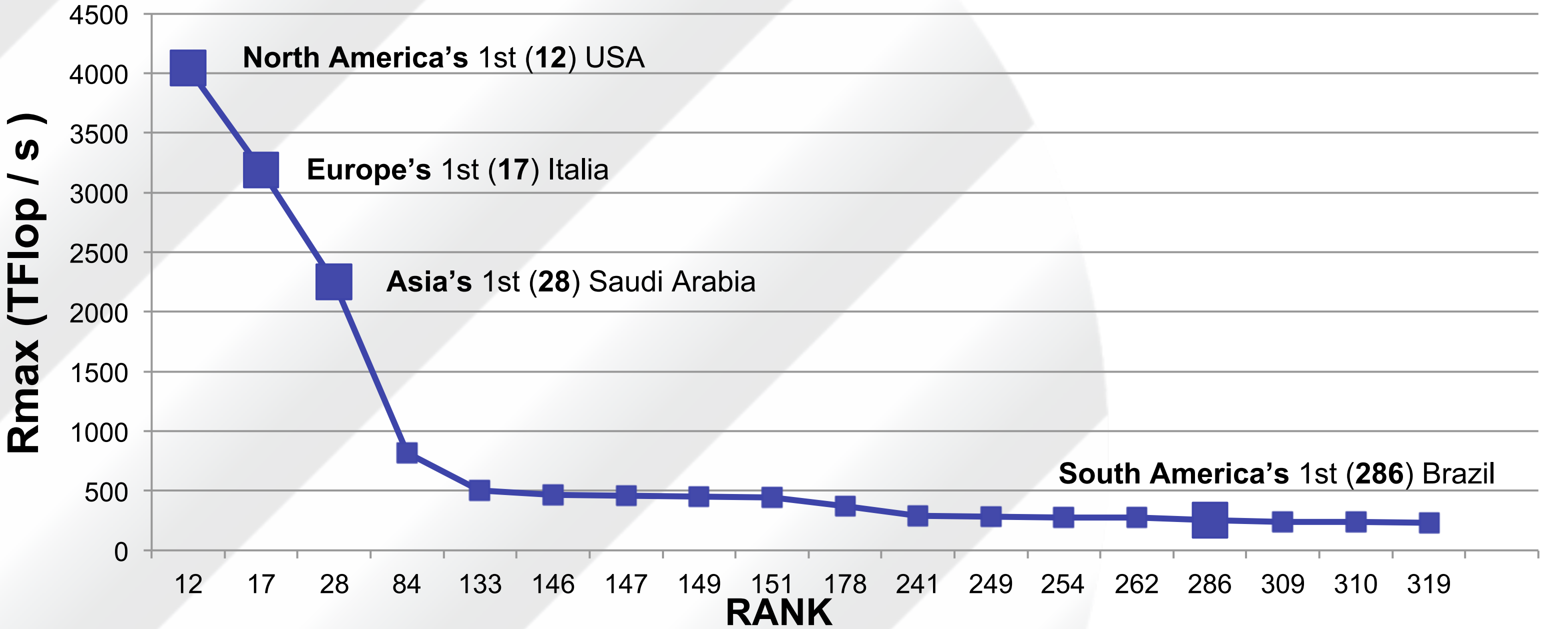
Figure 2. Single diapir evolution: (a) Geological time $=1My$, (b) Geological time $=5My$, (c) Geological time $=10My$, (d) Geological time $=25My$, (e) Geological time $=50My$ and (f) Geological time $=60My$.

Exploring the Stochastic Space

UNCERTAINTY QUANTIFICATION IN SEISMIC IMAGING

Motivation

- **Seismic imaging is one of the most computational demanding activities in Oil and Gas industry**
- **Involves massive data acquisition, computing, storage and visualization**
- **Oil companies are one of the main industries on TOP500 list (June 2015):**



- **Besides, according to U. Rüde, SIAM News, 2015:**

  "Extreme-scale systems will provide the computational power to move from qualitative simulation to predictive simulation, and from predictive simulation to optimization, parameter identification, and inverse problems; they will make stochastic simulations possible and allow us to better quantify uncertainties."

1 - Seismic Imaging in E&P Process

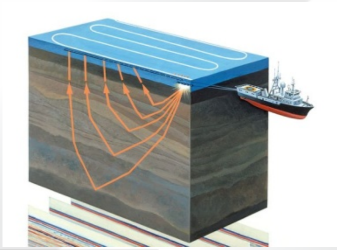
Data Acquisition → **Seismic Imaging** → **Geological interpretation** → **Reservoir Modeling** → **Financial Analysis**

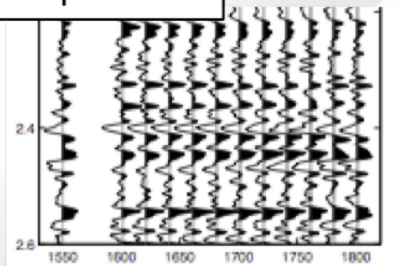
- Data Acquisition:
  - Seismograms
  - Well data
- Seismic Imaging:
  - Seismic Image
- Geological interpretation:
  - Geological model
- Reservoir Modeling:
  - Reservoir model
  - Flow simulation
  - Reserve estimations
- Financial Analysis:
  - Decision

Area of Interest

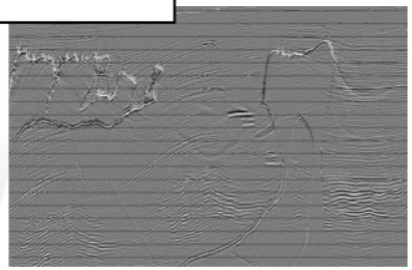
Seismic Imaging

Data Acquisition → **Tomography** → **Migration** → **Geological Interpretation**





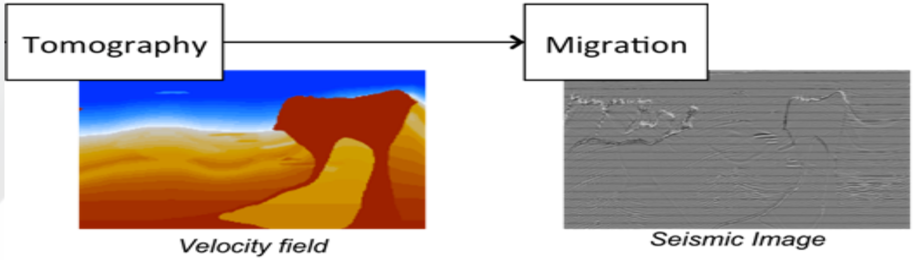
Velocity field



Seismic Image

1 - Seismic Imaging in E&P Process

Focus on **RTM** (Reverse Time Migration)



Input **Ouput**

2 forward models (acoustic or elastic wave propagation)

Forward model: Compute the time the wave takes to reach each point of the subsurface.

Reverse model: The receiver is now a source → **time reversing**

IMAGING CONCEPT: forward and reverse waves arrive at the same time → there is a "good chance" (probable) to be a **reflector**

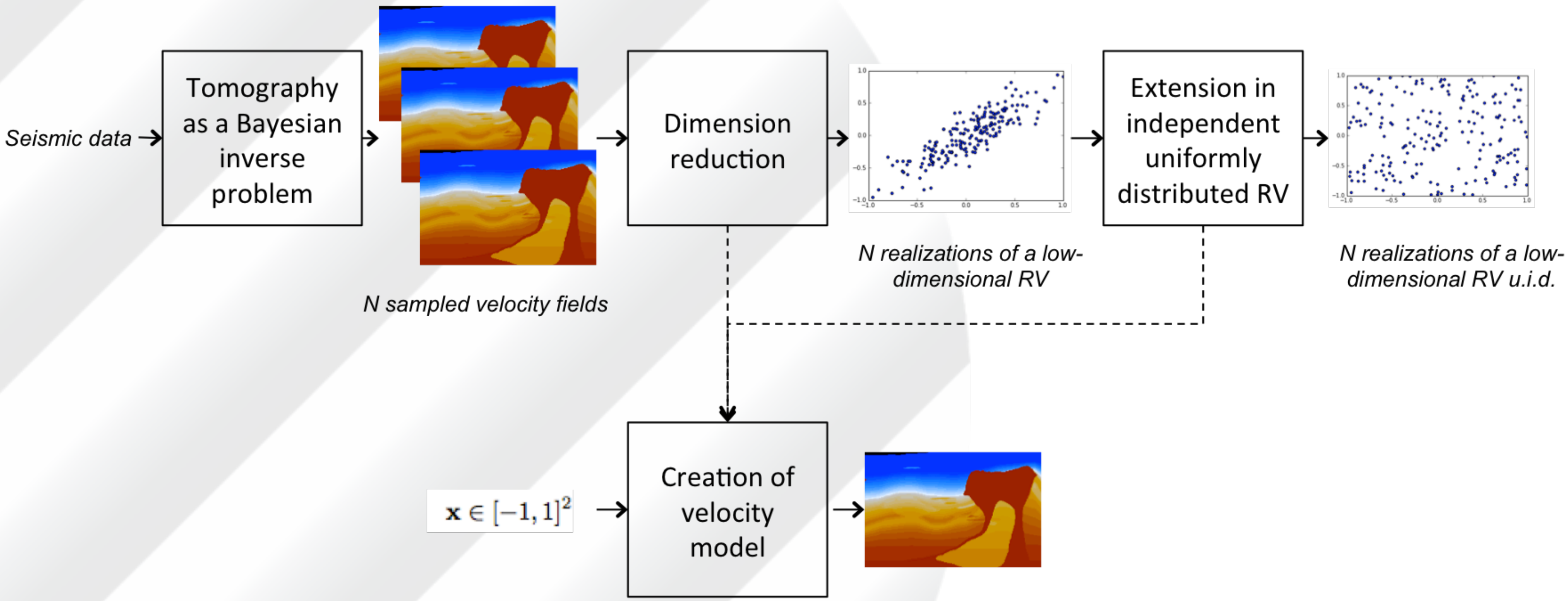
Tomography uses **approximations** of wave propagation: **Ray Tracing** or **Eikonal equation** (model errors)

Data can have some **measurements errors (seismograms noise data)**.

Uncertainties in the input data of the migration algorithm: propagation need to be understood and mitigated

2 - Uncertainty quantification in seismic imaging

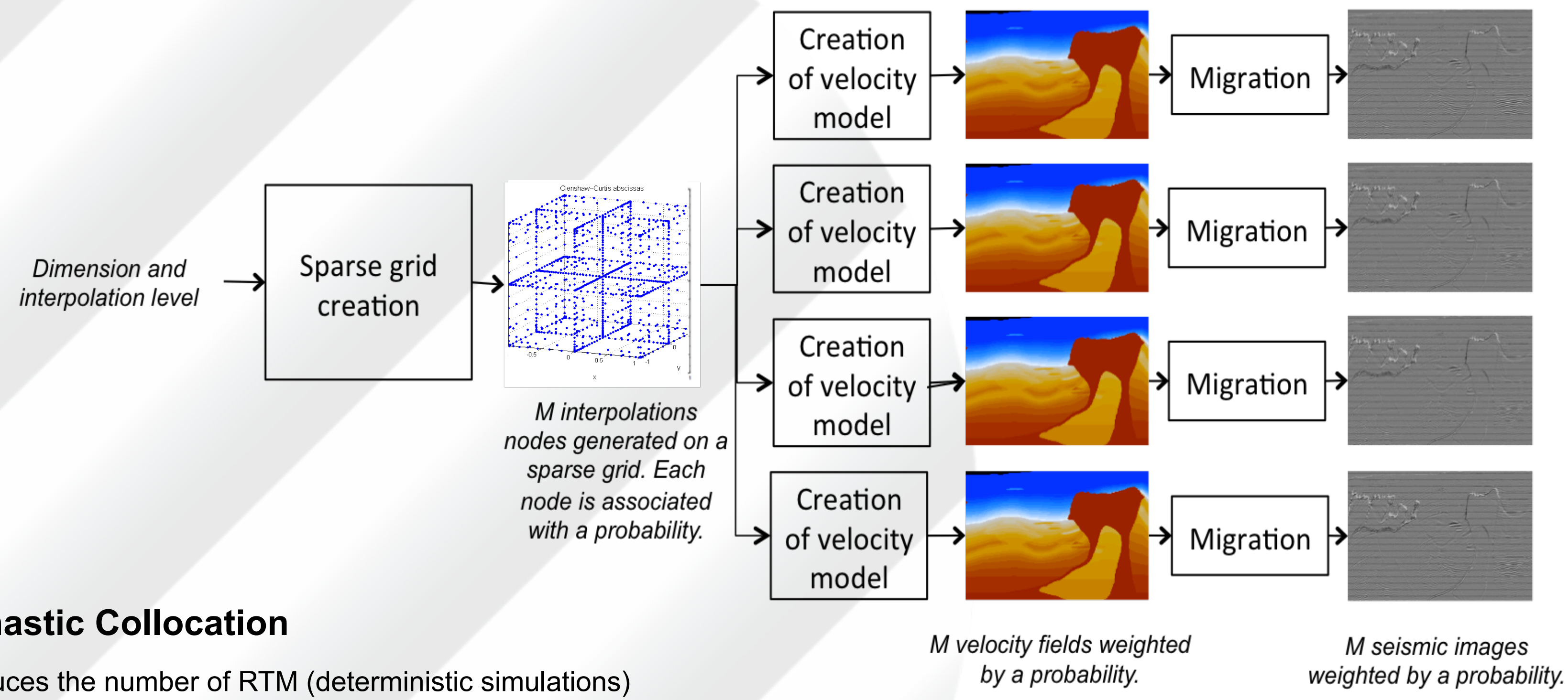
Step 1: Building an input data-driven model for imaging



RV: Random Vector (discrete field)

2 - Uncertainty quantification in seismic imaging

Step 2: propagation of uncertainties in RTM



Stochastic Collocation

- ❑ Reduces the number of RTM (deterministic simulations)
- ❑ Reduces interpolation error

Probabilistic Framework

Deterministic calculation: Optimized RTM Code

Uncertainty Quantification

Challenges in Uncertainty Quantification in Seismic Imaging

- Computing needs for conventional RTM (3D TTI, 1000 shots): one RTM runs in 5h on 10^2 Intel Xeon Phi, and thousands of RTMs are needed for UQ.
- **Main challenges to scale to real problem:**
  - Optimizing RTM Kernels
  - UQ: Stochastic collocation methods with dimension reduction vs MC methods.
  - Manage the computations: **Scientific Workflow Management System** needed to support the intensive (at least) two-level parallel computations ➔ Chiron SWfMS [3].
  - Draw insights from the results: how to visualize UQ on an seismic image?

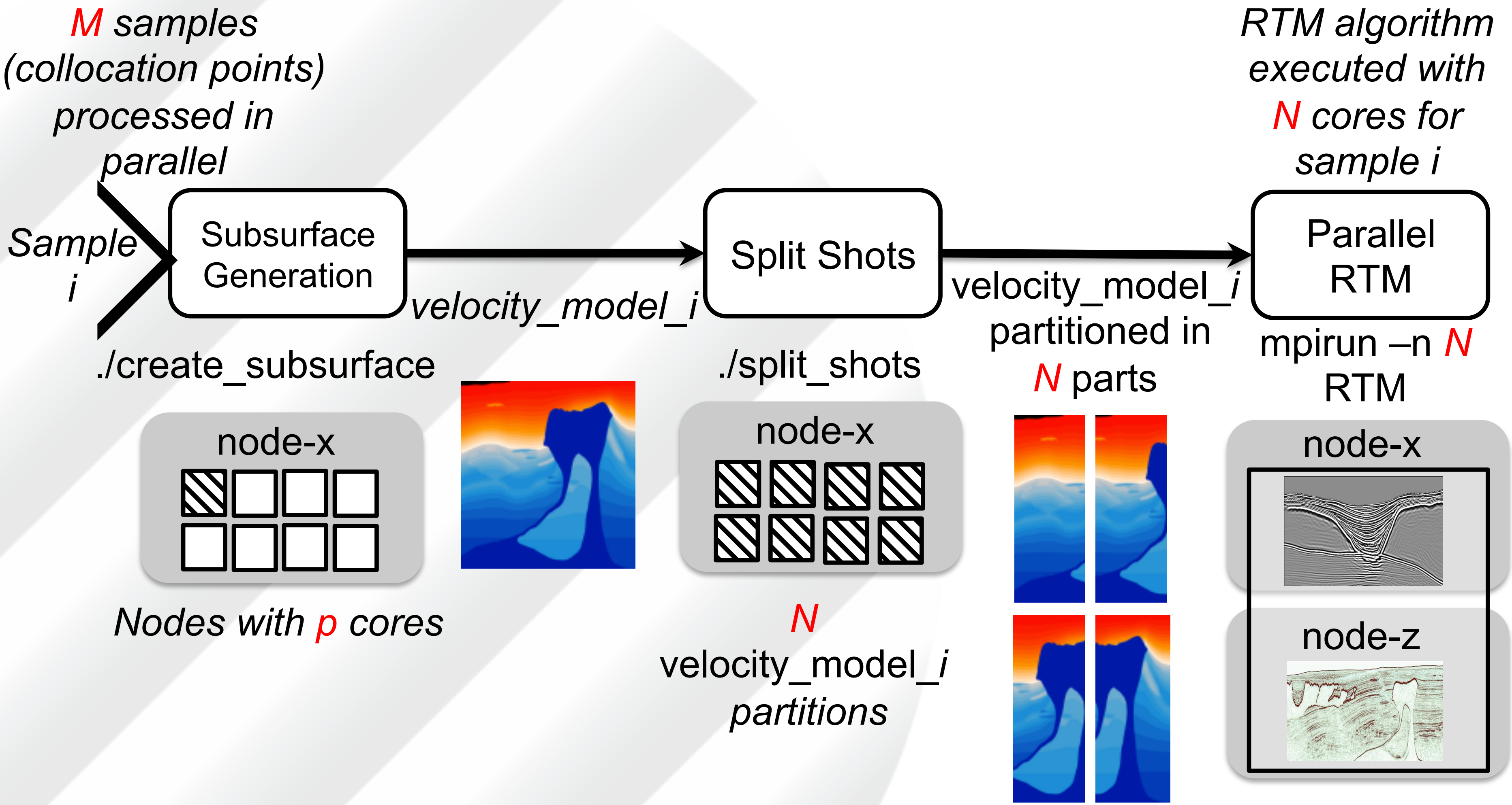
[3] Dias, J., et al. "Data-centric iteration in dynamic workflows." *Future Generation Computer Systems* (2014). (227 downloads since publication)

Chiron: a parallel workflow execution engine

- **Dataflow oriented engine by a workflow relational algebra [4]**
- **Non intrusive: workflow system supports several parallel numerical apps**
- **Strong Provenance support**
  - Online data analysis
  - Convergence tracking
  - Visualization of partial results
- **Dynamic interference on loop parameters**
- **Applications:**
  - Life sciences, Computational Fluid Dynamics (CFD), Uncertainty Quantification (UQ) in Particle Laden Flows and Seismic Imaging

[4] OGASAWARA, E., DIAS, J., SILVA, V., *et al.*, 2013, "Chiron: A Parallel Engine for Algebraic Scientific Workflows", *Concurrency and Computation*, v. 25, n. 16, pp. 2327–2341.

Chiron RTM Workflow Many-level Parallel Execution



Chiron’s distributed architecture

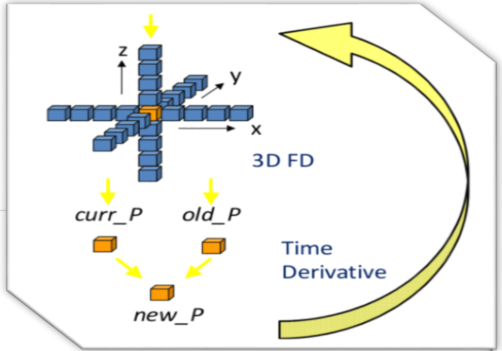
- **A decentralized approach (*d-Chiron*) for managing provenance data**
  - Fully distributed multi-master database
  - Horizontal data partitioning
- **Evaluation**
  - Synthetic workflow composed of three activities
    - *Average time of 30,240 activity executions (i.e. tasks): ~64s*
  - HPC environment: StRemi Cluster from Grid 5000 (Rennes region)
    - *88 CPUs AMD Opteron 6164 HE 1.7GHz: **1008 cores** (42 nodes)*
  - Performance evaluation
    - *Elapsed time → d-Chiron: 38 minutes, Chiron (centralized approach):* ~215 minutes
    - *Efficiency → d-Chiron: ~86.39% (high overhead: storing tasks in database)*

Some remarks:

- **Chiron's Profiler**
  - Development of a strategy to gather and query performance data while scientific workflows are executing
  - We achieved an efficiency of 93% using Endeavour Cluster (Intel, #92 TOP500, Jun 15) with 1,046 cores
    - *Chiron with a single provenance database*
    - *Synthetic workflow with three activities*
- **New Chiron's architecture**
  - Improvements to obtain better performances on more cores
- **Limitations in relation to the HPC environments**
  - RTM workflow execution using **large core counts**
  - **Storage capacity** can be **very limiting** for a large-scale data app like RTM
    - *In Endeavour cluster, we are limited to ~60GB*
    - *In SGI ICE at COPPE we produced ~300 GB of files using a small dataset*

Computational Optimizations for RTM kernels

RTM Stencil



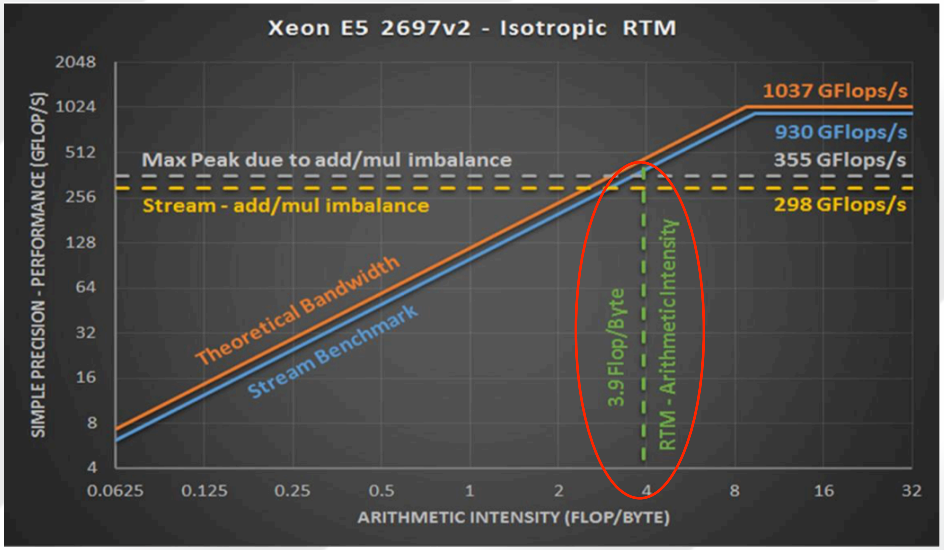

```
do k , Nz
   do j , Ny
      do i , Nx
                      new_P(i,j,k)   =    2 *    curr_P(i   ,j  ,k  )  -   old_P(i   ,j  ,k  )  +        C(I  ,j  ,k  )  * (  a0  *  curr_P(i   ,j  ,k  ) +
                                          a1 * (  curr_P(i-1,j  ,k  ) + curr_P(i+1,j  ,k  ) + curr_P(i  ,j-1,k  ) + curr_P(i  ,j+1,k  ) + curr_P(i  ,j  ,k-1) + curr_P(i  ,j  ,k+1)   )  +
                                          a2 * (            curr_P(i-2,j  ,k  ) + curr_P(i+2,j  ,k  ) + curr_P(i  ,j-2,k  ) + curr_P(i  ,j+2,k  ) + curr_P(i  ,j  ,k-2) + curr_P(i  ,j  ,k+2)   )  +
                                          a3 * (  curr_P(i-3,j  ,k  ) + curr_P(i+3,j  ,k  ) + curr_P(i  ,j-3,k  ) + curr_P(i  ,j+3,k  ) + curr_P(i  ,j  ,k-3) + curr_P(i  ,j  ,k+3)   )  +
                                          a4 * (  curr_P(i-4,j  ,k  ) + curr_P(i+4,j  ,k  ) + curr_P(i  ,j-4,k  ) + curr_P(i  ,j+4,k  ) + curr_P(i  ,j  ,k-4) + curr_P(i  ,j  ,k+4)   )  +
                                          a5 * (            curr_P(i-5,j  ,k  ) + curr_P(i+5,j  ,k  ) + curr_P(i  ,j-5,k  ) + curr_P(i  ,j+5,k  ) + curr_P(i  ,j  ,k-5) + curr_P(i  ,j  ,k+5)   )  +
                                          a6 * (  curr_P(i-6,j  ,k  ) + curr_P(i+6,j  ,k  ) + curr_P(i  ,j-6,k  ) + curr_P(i  ,j+6,k  ) + curr_P(i  ,j  ,k-6) + curr_P(i  ,j  ,k+6)   )  +
                                          a7 * (  curr_P(i-7,j  ,k  ) + curr_P(i+7,j  ,k  ) + curr_P(i  ,j-7,k  ) + curr_P(i  ,j+7,k  ) + curr_P(i  ,j  ,k-7) + curr_P(i  ,j  ,k+7)   )  +
                                          a8 * (  curr_P(i-8,j  ,k  ) + curr_P(i+8,j  ,k  ) + curr_P(i  ,j-8,k  ) + curr_P(i  ,j+8,k  ) + curr_P(i  ,j  ,k-8) + curr_P(i  ,j  ,k
+8)   )
      enddo
   enddo
enddo
```

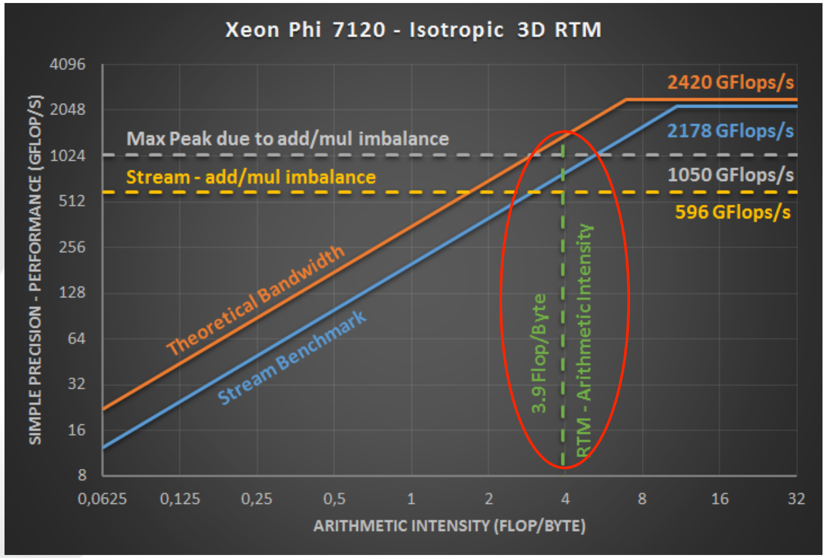
16th order → 3D → Acoustic → Isotropic → 61 Flops

NACAD

Computational optimizations for RTM kernels

Roofline Models for Xeon E5 and Xeon Phi 7210

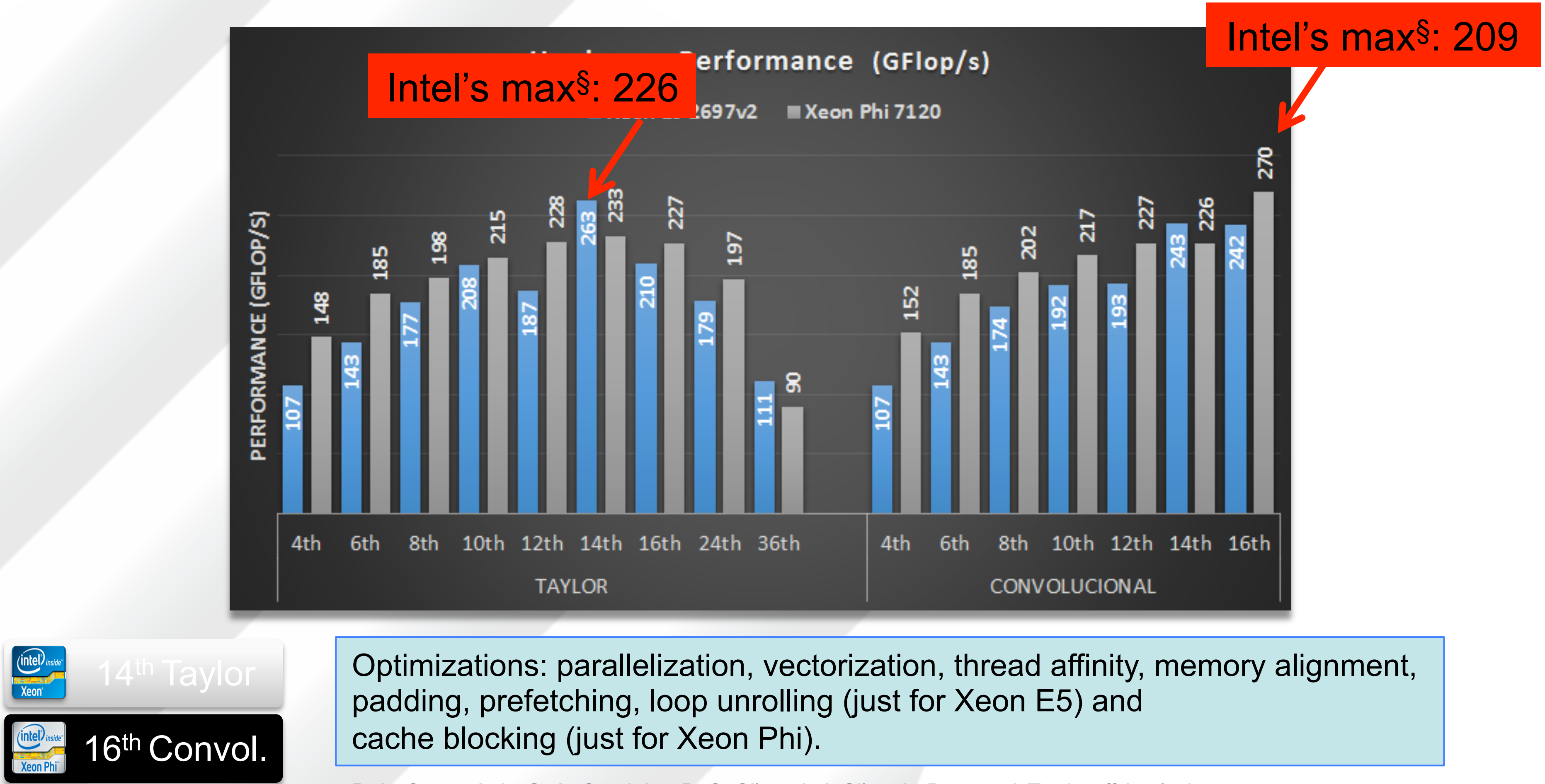




RTM algorithm classified as memory bound, low arithmetic intensity, memory also a bottleneck

Samuel Williams, Andrew Waterman, and David Patterson. 2009. Roofline: an insightful visual performance model for multicore architectures. Commun. ACM 52, 4 (April 2009), 65-76

Computational optimizations for RTM kernels



Optimizations: parallelization, vectorization, thread affinity, memory alignment, padding, prefetching, loop unrolling (just for Xeon E5) and cache blocking (just for Xeon Phi).

D. L. Costa, A. L. G. A. Coutinho, B. S. Silva, J. J. Silva, L. Borges, A Trade-off Analysis Between High-order Seismic RTM and Computational Performance Tuning,PANACM 2015

§C. Andreolli, P. Thierry, L. Borges, G. Skinner, C. Yount, Characterization and Optimization Methodology applied to Stencil Computations, Chapt 23, J. Reinders& & J. Jeffers, High Performance Parallelism Pearls, MK, 2015

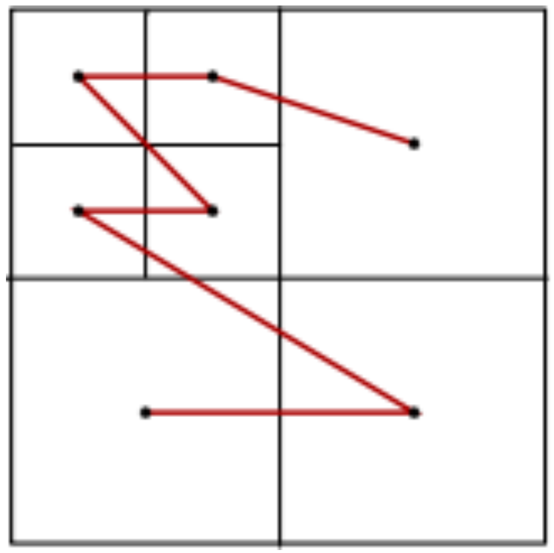
PARALLEL MESH GENERATION

Linear Octrees

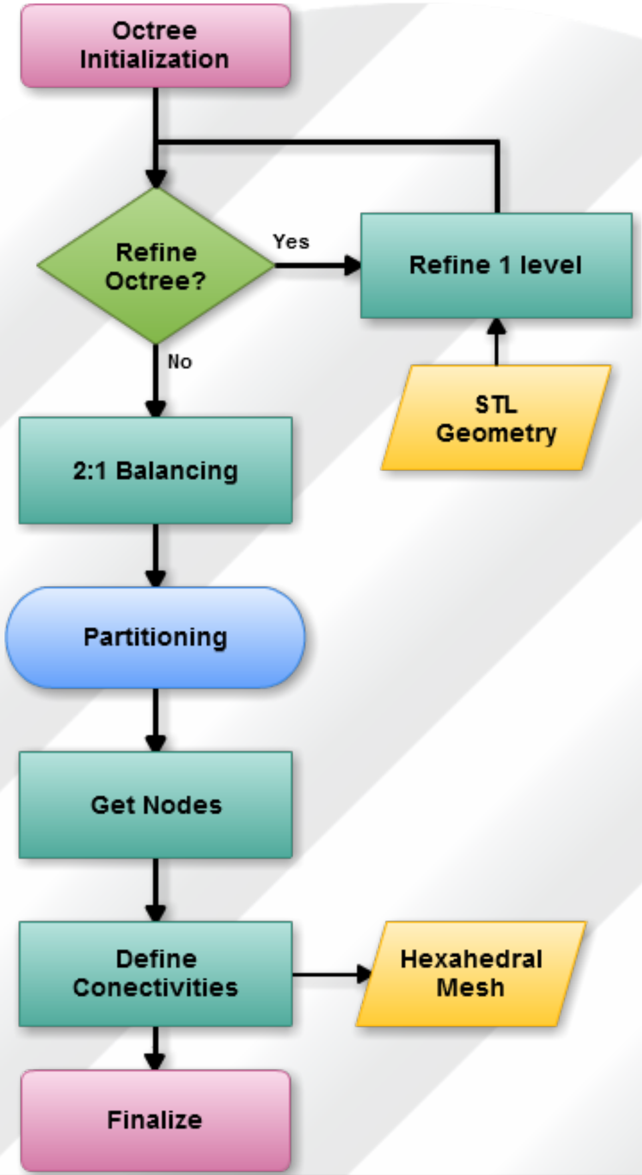
- **Octrees are hierarchical data structures that decompose a three-dimensional space in regular cubes, called octants.**

- Linear Octree
  - complete list of leaf nodes,
  - octants are encoded by a scalar key: Morton Code.

- Advantages:
  - Efficient memory use
  - Optimized tree traversal
  - Absence of pointers: more performance on sequential access and less communication overhead in parallel implementation



Non-conforming octree based mesh generation



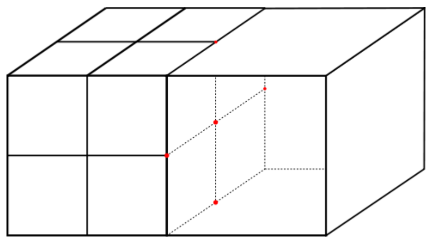
- ✓ **Refine** elements that intercept the immersed boundary.
- ✓ Surfaces are given by a triangulation, typically obtained from CAD package as STL files.
- ✓ Finding interceptions is based on bounding box trees.

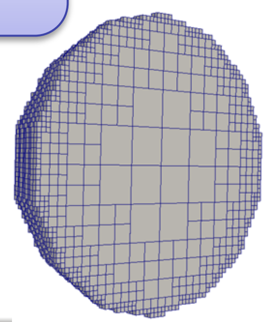
- ✓ **2:1 balancing** constrain ensure that no neighbor octants differentiated in more than one level.

- ✓ **Partitioning** is done just splitting the octree array in equal parts among the cores

- ✓ **Non-Conforming Mesh**

Hanging nodes are formed when there are two neighbor octants of different depth levels as their intersection is reduced to half-edges or a section of a face

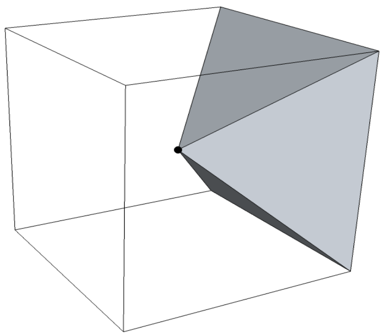




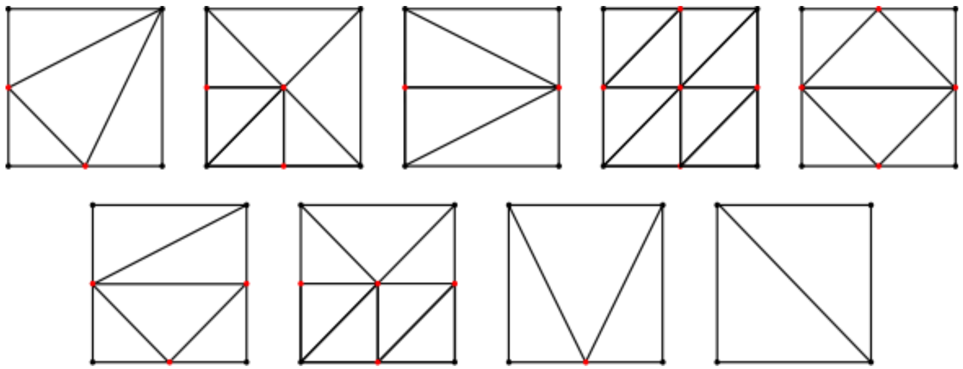
See more: Camata JJ, Coutinho ALGA. Parallel implementation and performance analysis of a linear octree finite elementmesh generation scheme. Concurrency and Computation: Practice and Experience 2013; 25(6):826–842, doi:10.1002/cpe.2869.

Conforming Technique

- ***Frey and George (2000)*** *propose a method where*
  - ✓ Irregular octants are decomposed in 6 pyramidal elements by inserting a central node
  - ✓ Nine templates are defined for face triangulation
  - ✓ All faces nodes are connected with the central node



Octant pyramidal decomposition



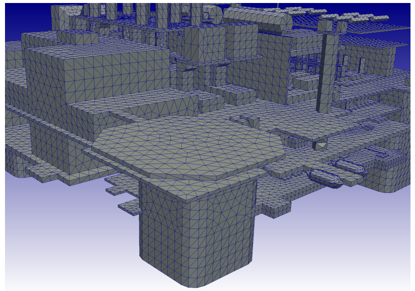
Pyramid base triangulation templates: red dots represent the existing hanging nodes in each face.

Conforming Technique

- ***Frey and George (2000)*** *scheme advantages:*
  - ✓ It does not require modifications in the octree construction
  - ✓ Embarrassing parallel (does not need neighboring information)
  - ✓ Availability of templates for all possible hanging nodes configuration

Because of its simplicity and the use of local mesh modifications, with potential low impact on parallel implementations this scheme is a good option to be incorporated in the our parallel octree mesher!

Performance Analysis on SDumont



- ***Weak scalability: offshore model***
  - Analysis performed from 64 to 16,384 cores (time in secs)

Cores	Octree levels	Mesh Size #elements	Total Time (BULL MPI)	Total Time (Intel MPI)
64	11	296,394,446	29.80	29.29
256	12	1,182,028,956	38.18	37.63
1024	13	4,725,678,648	50.13	49.61
4096	14	18,895,327,518	74.94	70.63
16384	15	75,519,891,076	133.74	--

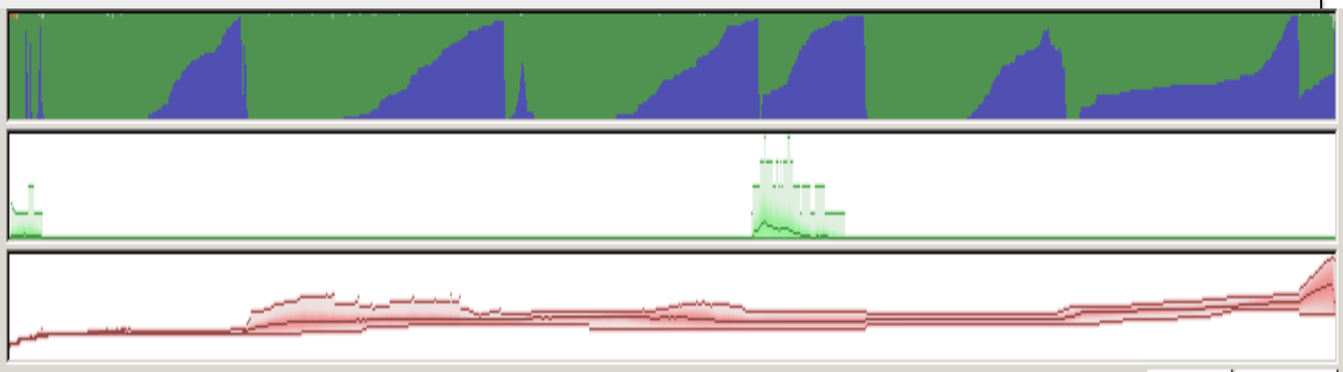
- Able to generate meshes up to 75 billion elements.
- Additional refinement levels increase runtime in about 20%.
- Last run shows highest increase in runtime.

Performance Analysis

Main thread activity

Floating-point operations

Memory usage



Profiles obtained by Allinea MAP
Execution on 1024 cores

- **Main thread activity**
  - 67% in CPU usage
  - 33 % in MPI communications
- **Floating-point Operations:**
  - Floating point operations are concentrated in computing geometrical interceptions with the input STL file.
- **Memory Usage:**
  - Continuous memory consumption growth due to the octree refinement
  - Sudden growth observed at the end could be explained by the meshing routine

See more: Ghisi IT, Camata JJ, Coutinho ALGA. Impact of tetrahedralization on parallel conforming octree mesh generation. IJNMF, 2014; 75(11):800–814, doi:10.1002/fld.3918.

CONCLUDING REMARKS AND DISCUSSION

- **We reviewed NACAD's activities. Challenges?**
- **Multiphysics**
  - Multiphysics problems have been addressed using 2 software platforms: EdgeCFD and libMesh
  - Simulation of complex systems combining multiple physical phenomena is one of the main motivations for extreme computing
  - Solvers, AMR/C challenging problems particularly at large core counters
- **UQ in Seismic Imaging**
  - We have tested and implemented a UQ framework for seismic imaging: Tomography+Migration, what about FWI?
  - Visualization of Uncertainty in Seismic Images beyond distance based metrics ? How? Entropy?
  - Getting insights from UQ: Data analytics?
  - Managing the complexity of sampling stochastic space
    - *Issues: data management, fault tolerance, data provenance, etc*
  - Stencil optimizations, half-precision, improving arithmetic intensity, I/O Optimization
- **Parallel Mesh Generation**
  - Scale-up a parallel octree mesh generation scheme
  - Build a new solver for it? Using immersed boundary techniques?

NACAD's Team

- Faculty:
  - HPC:A. Coutinho (PI), R. Elias
  - UQ: 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, D. Costa, J. Dias, I. Ghisi, M. Goncalves Jr, J. Gonzalez, G. Guerra, N. Guevara Jr, F. Horta, T. Lavril, E. Lins, A. Mendonça, T. Miras, E. Ogasawara, D. Oliveira, A. Rossa, F. Seabra, C. Silva, J. Silva, V. Silva, R. Souza, D. Vasconcelos, S. Zio
- Computer Resources: NACAD/COPPE/UFRJ, Intel, LNCC

Visit us at www.nacad.ufrj.br