# Domain decomposition techniques for hyperbolic equations on unstructured grids

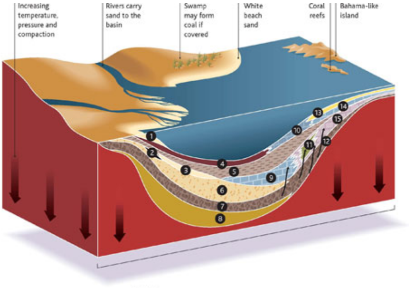
Antonio Cervone, Nur A. Fadel, Luca Formaggia
antonio.cervone@polimi.it

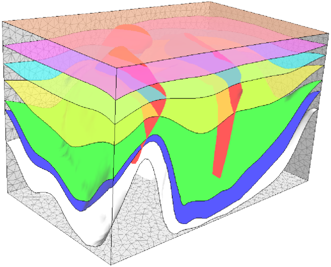
MOX, Politecnico di Milano

June 5th, 2012

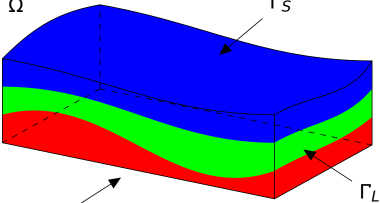
www.lifev.org

# Motivation - sedimentary basins



# Mathematical model




$$\begin{cases} \nabla \cdot (\mu(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^T)) - \nabla p = -\rho \boldsymbol{g} & \text{in } \Omega \times (0, T] \\ \nabla \cdot \boldsymbol{u} = 0 & \text{in } \Omega \times (0, T] \\ \dfrac{\partial}{\partial t}\{\rho, \mu\} + \boldsymbol{u} \cdot \nabla \{\rho, \mu\} = 0 & \text{in } \Omega \times (0, T] \\ \rho = \rho_0, \quad \mu = \mu_0 & \text{in } \Omega \times \{0\} \\ \boldsymbol{u} = \bar{\boldsymbol{u}} & \text{on } \Gamma \end{cases}$$

# Stokes system discretization - 1

bilinear forms

$$a(\boldsymbol{u},\boldsymbol{v}) = \int_\Omega \mu \nabla \boldsymbol{u} : \nabla \boldsymbol{v} \qquad \forall \boldsymbol{u},\boldsymbol{v} \in \boldsymbol{H}^1$$

$$b(p,\boldsymbol{v}) = \int_\Omega p \nabla \cdot \boldsymbol{v} \qquad \forall p \in L^2_0, \quad \forall \boldsymbol{v} \in \boldsymbol{H}^1$$

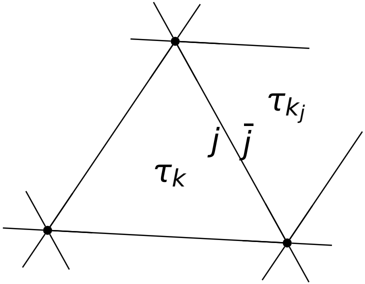
$$f(\boldsymbol{v}) = -\int_\Omega \rho \boldsymbol{g} \cdot \boldsymbol{v} \qquad \forall \boldsymbol{v} \in \boldsymbol{H}^1$$

weak formulation

$$\begin{cases} a(\boldsymbol{u},\boldsymbol{v}) + b(p,\boldsymbol{v}) = f(\boldsymbol{v}) & \forall \boldsymbol{v} \in \boldsymbol{X} \\ b(q,\boldsymbol{u}) = 0 & \forall q \in S \end{cases}$$

# Stokes system discretization - 2

- tetrahedral grid $\mathcal{T}_h(\Omega)$, $n_e$ elements and $n_p$ points
- element $\tau_k$, $\bigcup_k \tau_k = \mathcal{T}_h$
- $X = \{v_h \in H^1 : v_h|_{\tau_k} \in \mathbb{P}^1_b\}$
- $S = \{q_h \in L^2_0 : v_h|_{\tau_k} \in \mathbb{P}^1\}$



# Characteristic function

$$\lambda_i(\boldsymbol{x}) = \begin{cases} 1 & \text{if } \boldsymbol{x} \in \Omega_i \\ 0 & \text{if } \boldsymbol{x} \notin \Omega_i \end{cases}$$

$i =$ , , ($s$ components)

$$\rho = \sum_{i=1}^{s} \lambda_i \rho_i, \quad \mu = \sum_{i=1}^{s} \lambda_i \mu_i$$

evolution equation for $\boldsymbol{\lambda}$

$$\frac{\partial \boldsymbol{\lambda}}{\partial t} + \boldsymbol{u}^n \cdot \nabla \boldsymbol{\lambda} = 0$$

# Characteristic function discretization - 1

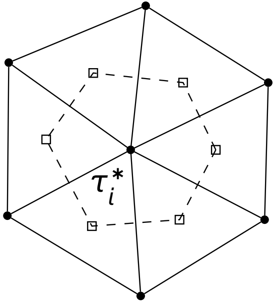
- multi-fluid support
- robust
- efficient
- automatic topology changes

# Characteristic function discretization - 2

- finite volume explicit method

$$\boldsymbol{\lambda}_h^{n+1} = \boldsymbol{\lambda}_h^n - \Delta t^n \boldsymbol{u}^n \cdot \nabla \boldsymbol{\lambda}_h^n$$

- dual mesh $\mathcal{T}_h^*(\Omega)$ with $n_p$ elements
- element $\tau_i^*$, $\bigcup_i \tau_i = \mathcal{T}_h^*$
- $\lambda_h \in V_0^*$, $V_0^* = \{\varphi_h \in L^2 : \varphi_h|_{\tau_i^*} \in \mathbb{P}^0\}$
- $\phi_h \in V_1$, $V_1 = \{\varphi_h \in C^1 : \varphi_h|_{\tau_k} \in \mathbb{P}^1\}$
- $\phi_h^n = \boldsymbol{I}_h^1 \lambda_h^n \quad (\phi_i^n \equiv \lambda_i^n)$



# Flux approximation

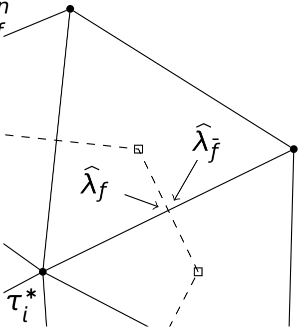
finite volume approximation on the dual grid

$$\lambda_s^{n+1} = (1 + D^n)\,\lambda_s^n - \sum_f \nu_f^n \Phi(\widehat{\lambda}_{s,f}^n, \widehat{\lambda}_{s,\bar{f}}^n)$$

$$D^n = \frac{\Delta t^n}{|\tau|} \oint_{\partial\tau} \boldsymbol{u} \cdot \boldsymbol{n} = \sum_f \nu_f^n$$

$$\nu_f^n = \frac{\Delta t^n}{|\tau|} \oint_f \boldsymbol{u} \cdot \boldsymbol{n}$$

$$\widehat{\lambda}_{s,f}^n = \lambda_s^n + \delta\lambda_{s,f}^n$$



# Interface values - 1

$\delta\lambda^n_{s,f}$ comes from the constrined minimization problem

$$\begin{cases} \min\limits_{\delta\lambda^n_{s,f}} \frac{1}{2} \sum\limits_s \left( \lambda^n_s - \phi^n_{s,f} + \delta\lambda^n_{s,f} \right)^2 \\ \sum\limits_s \delta\lambda^n_{s,f} = 0 \\ -\lambda^n_s < \delta\lambda^n_{s,f} < min \left( \frac{1+D^n-\nu^n_f|f|}{\nu^n_f|f|}, 1 - \lambda^n_s \right) \end{cases}$$

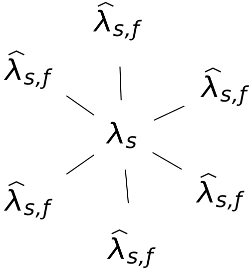
$\delta\lambda^n_{s,f}$ as a best fit approx of LS [1]

[1] Villa A., Formaggia L. Implicit tracking for multi-fluid simulations. JCP 229 (2010) 5788–5802

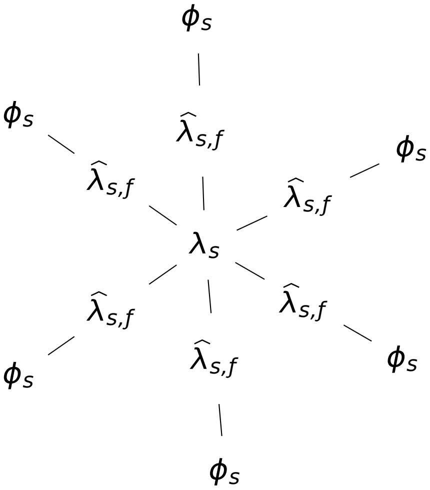
# Interface values - 2

$$\lambda_s$$

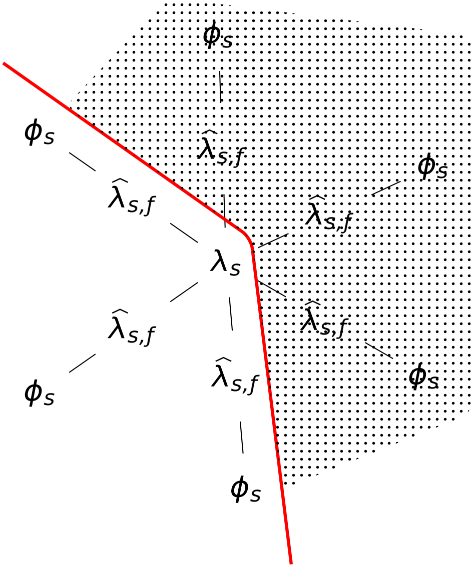
# Interface values - 2



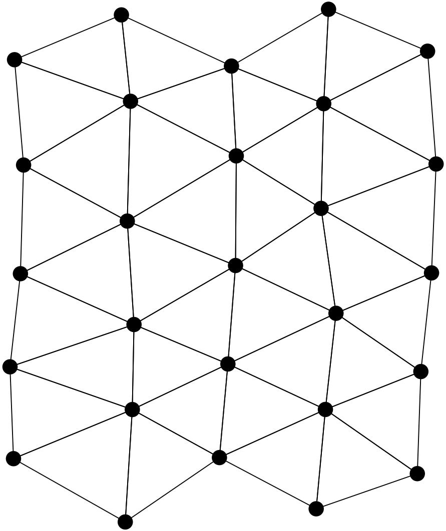
# Interface values - 2



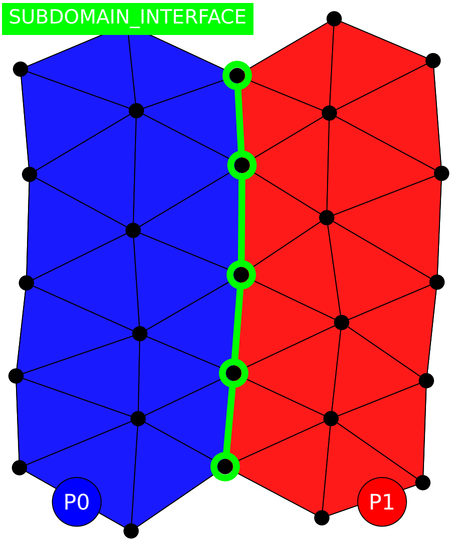
# Interface values - 2



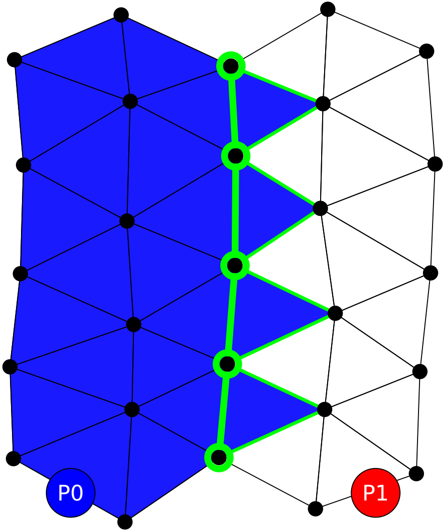
# Domain decomposition with hyperbolic eqns



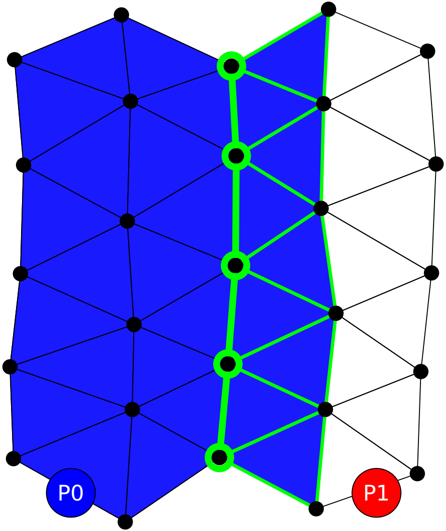
# Domain decomposition with hyperbolic eqns



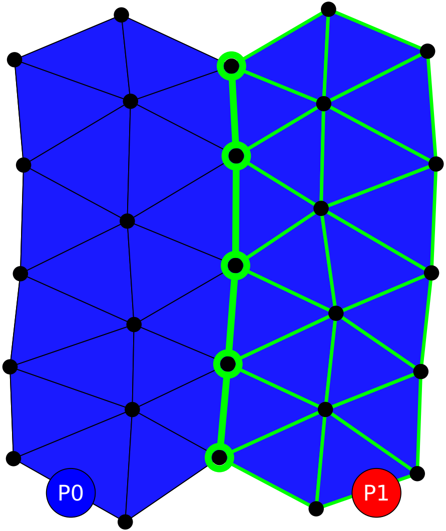
# Domain decomposition with hyperbolic eqns



# Domain decomposition with hyperbolic eqns



# Domain decomposition with hyperbolic eqns



# LifeV

- finite element library
- originally developed for life sciencies
- advanced FSI solvers
- heart modeling
- 1D models
- multiscale
- parallel
- based on `Trilinos`

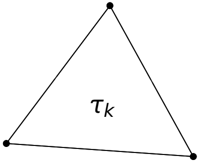
# LifeV maps

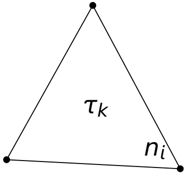
the `MapEpetra` object

- stores 2 `Epetra_Map`s
  - `Unique`: objects used to assembly
  - `Repeated`: objects used to share information
- handles import/exports between different maps
- `operator=()`
- `operator+=()`
- `operator|=()` for block support
- used to build all algebraic objects

# Overlapping maps - 1

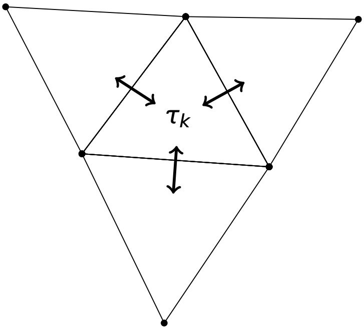
- based on connectivity
- each geometric entity knows its neighborhood

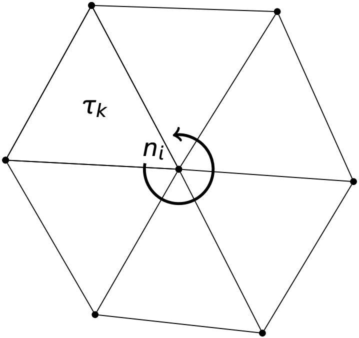


# Overlapping maps - 1

- based on connectivity
- each geometric entity knows its neighborhood



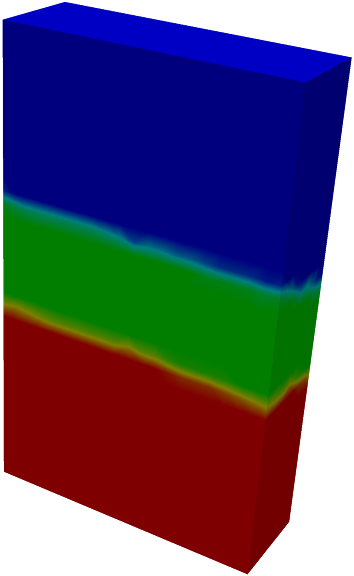



# Overlapping maps - 2

Algorithm

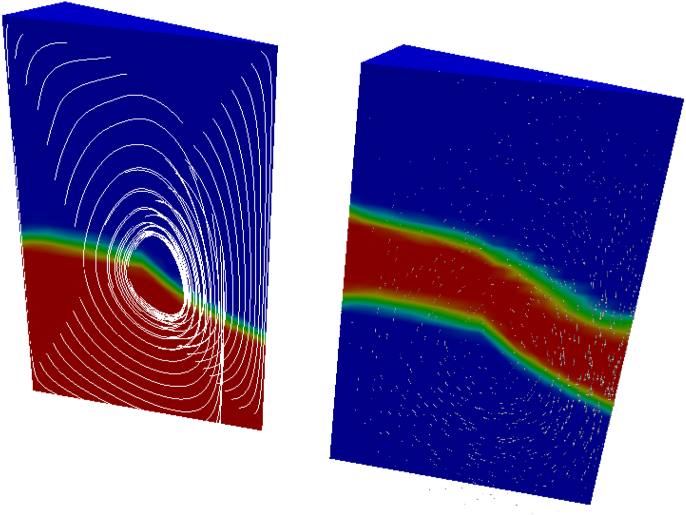
- build neighborhood information on the full mesh
- partition the mesh
- delete full mesh
- identify subdomain interface entities
  (check on neighbors)
- init searching set with subdomain interface entities
- `for:` level of overlap
  - add all neighbors not on current partition
  - replace searching set with added entities

# Simulation setup

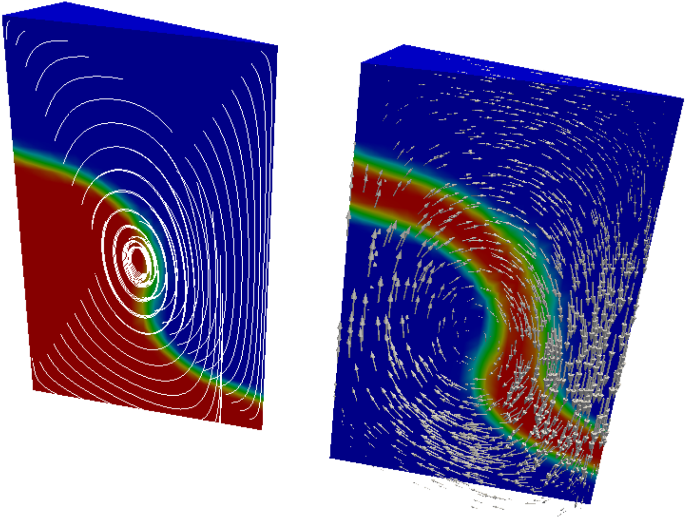


- $n_e \sim 200K$
- $n_p \sim 40K$
- dof $\sim 750K$
- (blue) $\mu = 3.0$, $\rho = 3.5$
- (green) $\mu = 2.0$, $\rho = 0.2$
- (red) $\mu = 0.1$, $\rho = 1$

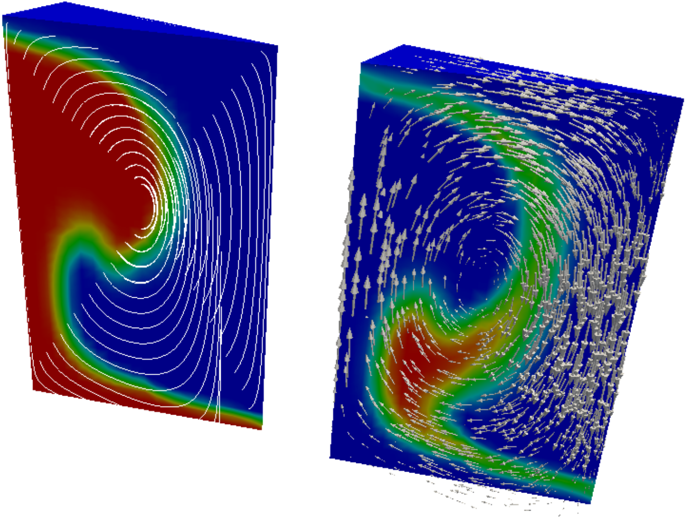
# Time evolution



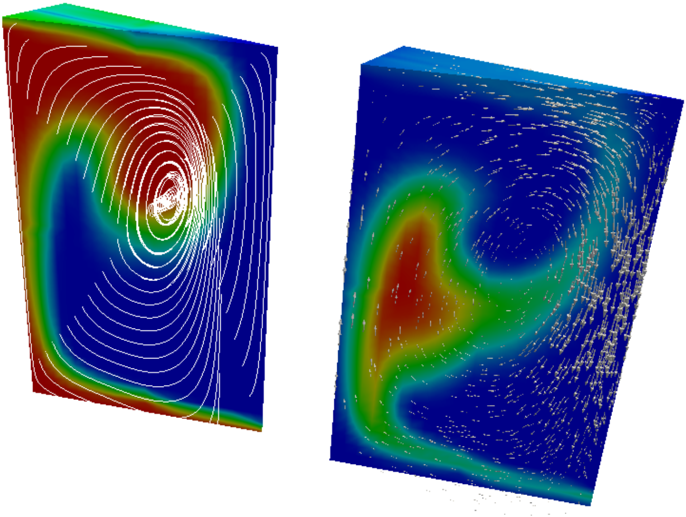
# Time evolution



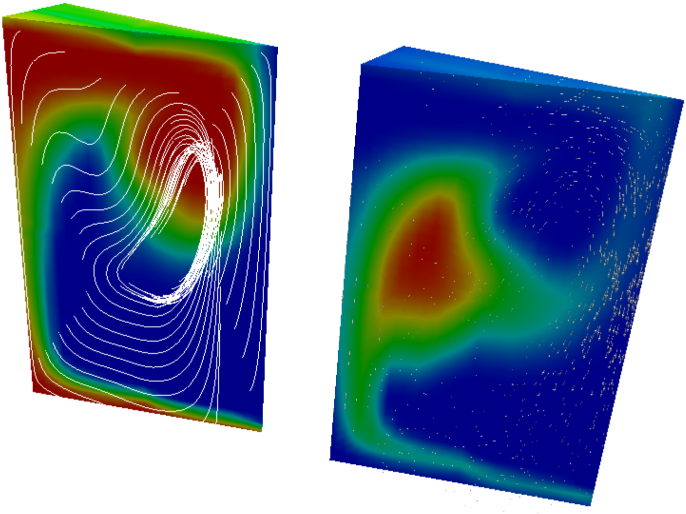
# Time evolution



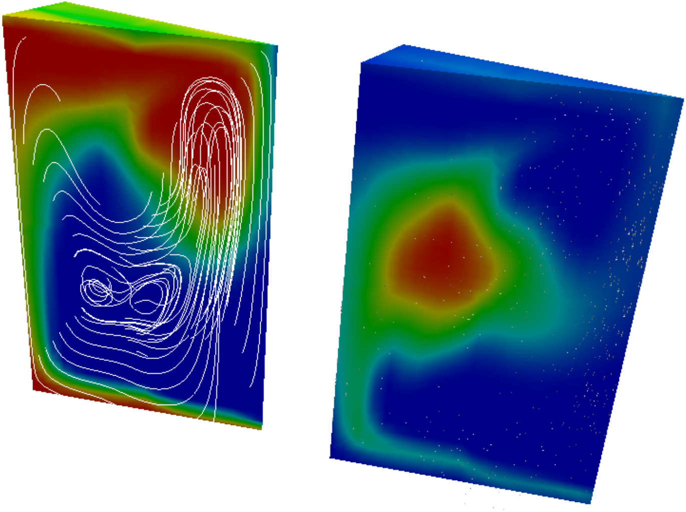
# Time evolution



# Time evolution



# Time evolution

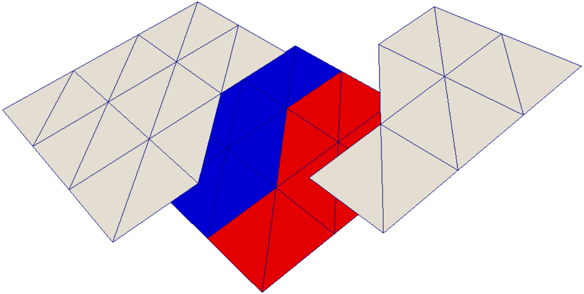


# Performance analysis

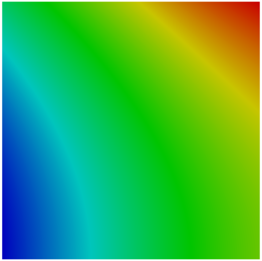
- $\mu_1/\mu_2 \sim 1 \rightarrow CN = 7 \cdot 10^5$
- $\mu_1/\mu_2 \sim 10 \rightarrow CN = 2.5 \cdot 10^6$
- $\mu_1/\mu_2 \sim 100 \rightarrow CN = 2.5 \cdot 10^7$
- preconditioning computation
  - reuse?
  - Vanka smoothers
  - HYPRE
- mesh memory footpint

# Assembly

- phisically replicated mesh
- resort to overlapping maps to avoid comm?
- computation intensive vs memory/comm intensive
- create graph for matrix init
- virtual subdomain interface nodes



# ADR test



- $4.5M$ elements
- $2.2M$ points
- $4K$ interface elements
- $4K$ interface points

# Performance

| PROC/NODE | STANDARD | REPEATED | GAIN |
|---|---|---|---|
| 2/2 | 11.52 | 11.1 | 3.6% |
| 4/4 | 6.13 | 5.89 | 3.9% |
| 8/8 | 3.11 | 3.01 | 3.2% |
| 16/2 | 1.71 | 1.64 | 4.1% |
| 16/4 | 1.63 | 1.62 | 0.0% |
| 16/16 | 1.80 | 1.71 | 5.0% |
| 64/8 | 0.47 | 0.42 | 10.6% |
| 64/64 | 0.21 | 0.18 | 14.3% |

thanks for the attention... go watch the poster!

# LifeV::Geophysics@EuroTUG2012


LIFEV

www.lifev.org

A. Cervone†, N. Fadel†, A. Fumagalli†, G. Iori† and M. Kern*

† MOX Laboratory, Department of Mathematics, Politecnico di Milano, Milan

* INRIA Rocquencourt, Paris

Contacts: alessio.fumagalli@mail.polimi.it and lifev-user@googlegroups.com

INVENTEURS DU MONDE NUMÉRIQUE