A Fictitious domain method

SMARTWING-DYNAMORPH

INSTITUT de MATHEMATIQUES de TOULOUSE

Michel Fournié

Sébastien Court

Friday, March 30th, 2012

Local developments

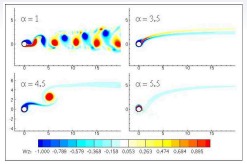
CFD
GETFEM++

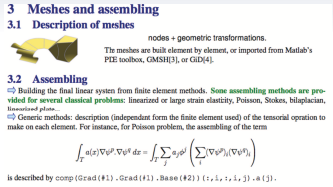
HPC
Parallelization

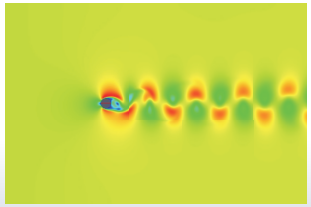
Local developments

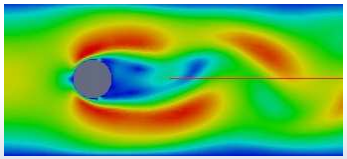
CFD
GETFEM++

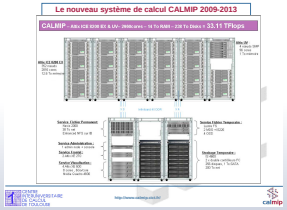
HPC
Parallelization




3 Meshes and assembling
3.1 Description of meshes
nodes + geometric transformations.
The meshes are built element by element, or imported from Matlab's PDE toolbox, GMSH[3], or GiD[4].
3.2 Assembling
⇨ Building the final linear system from finite element methods. Some assembling methods are provided for several classical problems: linearized or large strain elasticity, Poisson, Stokes, bilaplacian, linearized plate...
⇨ Generic methods: description (independent form the finite element used) of the tensorial opration to make on each element. For instance, for Poisson problem, the assembling of the term
$$\int_T a(x)\nabla\psi^P.\nabla\psi^Q\,dx = \int_T \sum_j a_j\phi^j\left(\sum_i(\nabla\psi^P)_i(\nabla\psi^Q)_i\right)$$
is described by comp(Grad(#1).Grad(#1).Base(#2))(:,i,:,i,j).a(j).



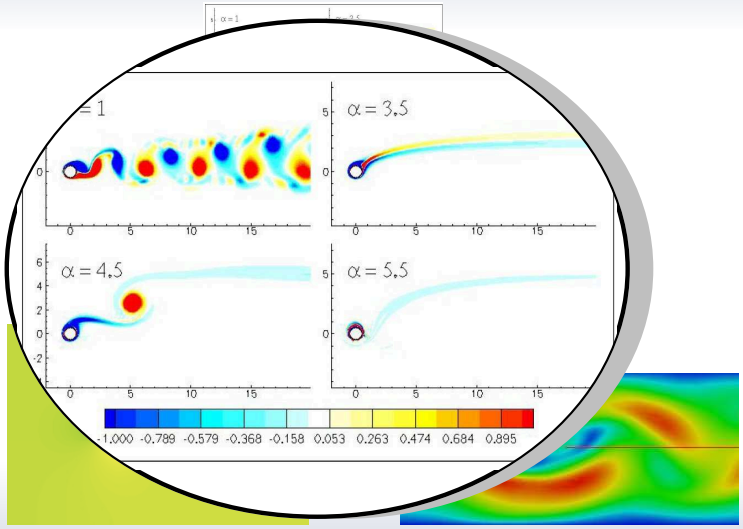


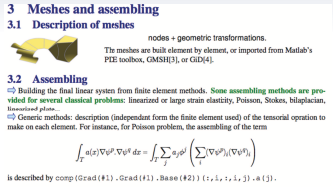


Local developments

CFD
GETFEM++

HPC
Parallelization




3 Meshes and assembling

3.1 Description of meshes

nodes + geometric transformations.

The meshes are built element by element, or imported from Matlab's PDE toolbox, GMSH[3], or GiD[4].

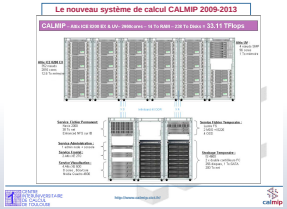
3.2 Assembling

⇨ Building the final linear system from finite element methods. Some assembling methods are provided for several classical problems: linearized or large strain elasticity, Poisson, Stokes, bilaplacian, linearized plate...

⇨ Generic methods: description (independent form the finite element used) of the tensorial opration to make on each element. For instance, for Poisson problem, the assembling of the term

$$\int_T a(x)\nabla\psi^p.\nabla\psi^q\,dx = \int_T \sum_j a_j \phi^j \left(\sum_i (\nabla\psi^p)_i (\nabla\psi^q)_i \right)$$

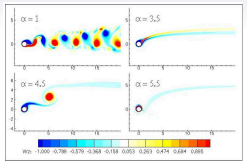
is described by `comp(Grad(#1).Grad(#1).Base(#2))(:,i,:,i,j).a(j)`.

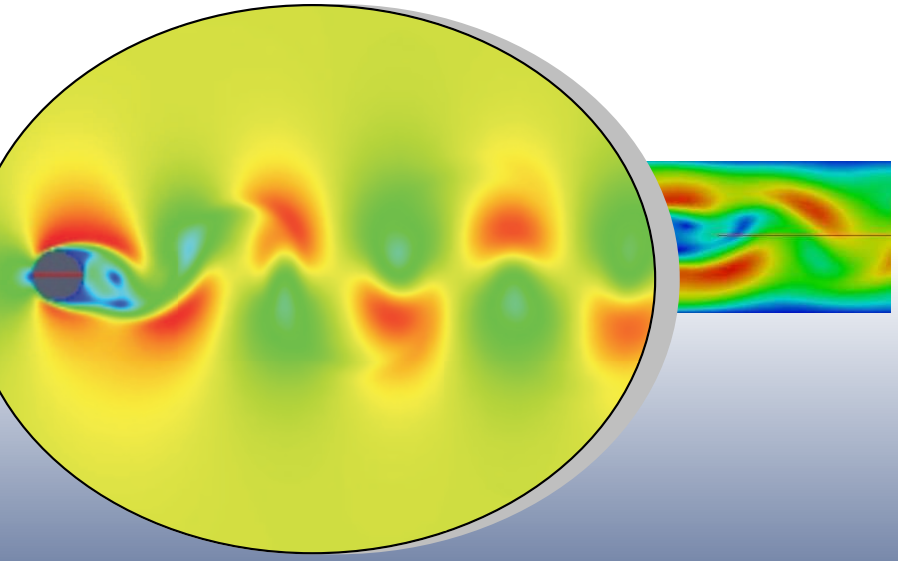


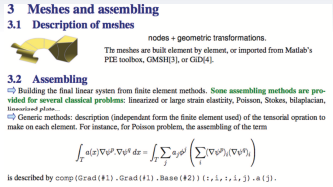
Local developments

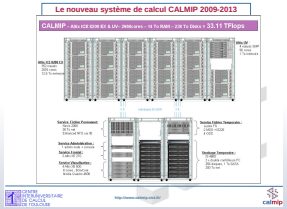
CFD
GETFEM++

HPC
Parallelization





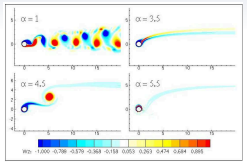

3 Meshes and assembling
3.1 Description of meshes
nodes + geometric transformations.
The meshes are built element by element, or imported from Matlab's PDE toolbox, GMSH[3], or GiD[4].
3.2 Assembling
⇨ Building the final linear system from finite element methods. Some assembling methods are provided for several classical problems: linearized or large strain elasticity, Poisson, Stokes, bilaplacian, linearized plate...
⇨ Generic methods: description (independent form the finite element used) of the tensorial opration to make on each element. For instance, for Poisson problem, the assembling of the term
$$\int_T a(x)\nabla\psi^P.\nabla\psi^Q\,dx = \int_T \sum_j a_j\phi^j\left(\sum_i(\nabla\psi^P)_i(\nabla\psi^Q)_i\right)$$
is described by comp(Grad(#1).Grad(#1).Base(#2))(:,i,:,i,j).a(j).

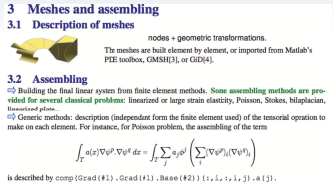


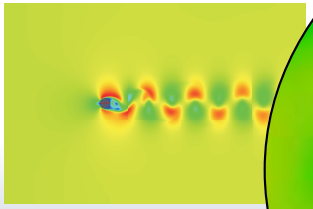
Local developments

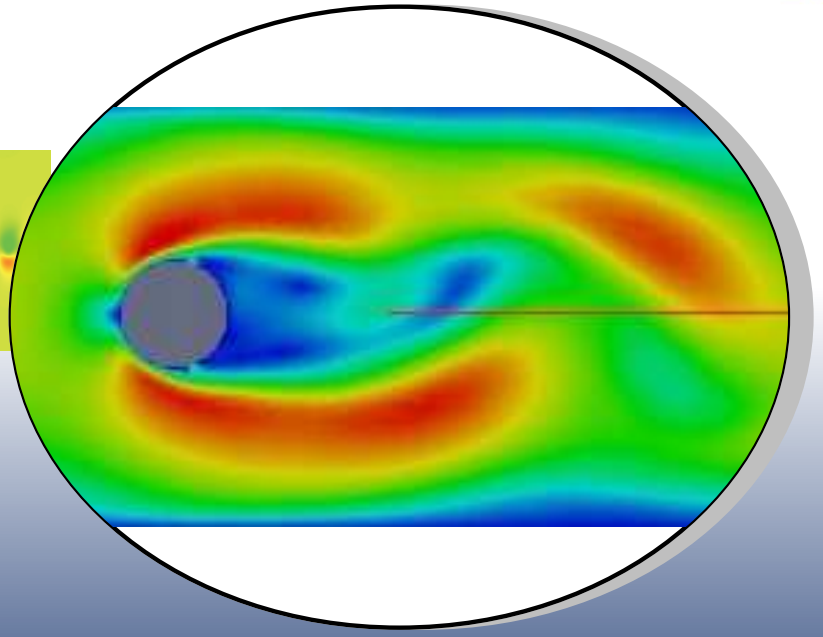
CFD
GETFEM++

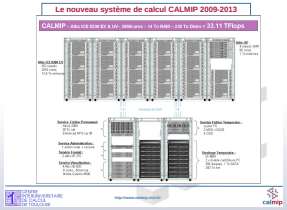
HPC
Parallelization




3 Meshes and assembling
3.1 Description of meshes
nodes + geometric transformations.
The meshes are built element by element, or imported from Matlab's PDE toolbox, GMSH[3], or GiD[4].
3.2 Assembling
⇨ Building the final linear system from finite element methods. Some assembling methods are provided for several classical problems: linearized or large strain elastizity, Poisson, Stokes, bilaplacian, linearized plate...
⇨ Generic methods: description (independent form the finite element used) of the tensorial opration to make on each element. For instance, for Poisson problem, the assembling of the term
$$\int_T a(x)\nabla\psi^P.\nabla\psi^S\,dx = \int_T \sum_j a_j \phi^j \left(\sum_i (\nabla\psi^P)_i (\nabla\psi^S)_i\right)$$
is described by comp(Grad(#1).Grad(#1).Base(#2))(:,i,:,i,j).a(j).



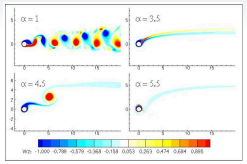


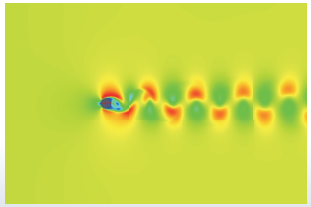


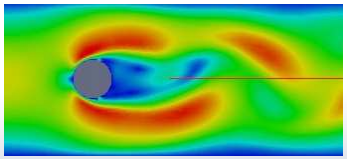
Local developments

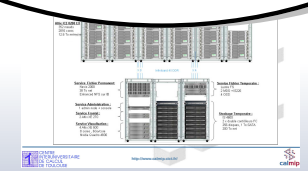
CFD
GETFEM++

HPC





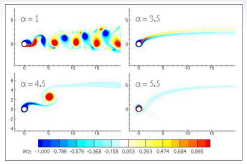


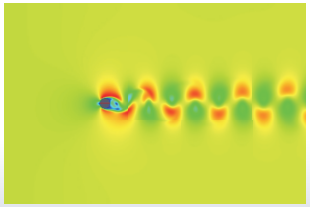


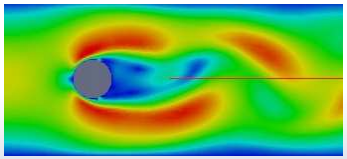
Local developments

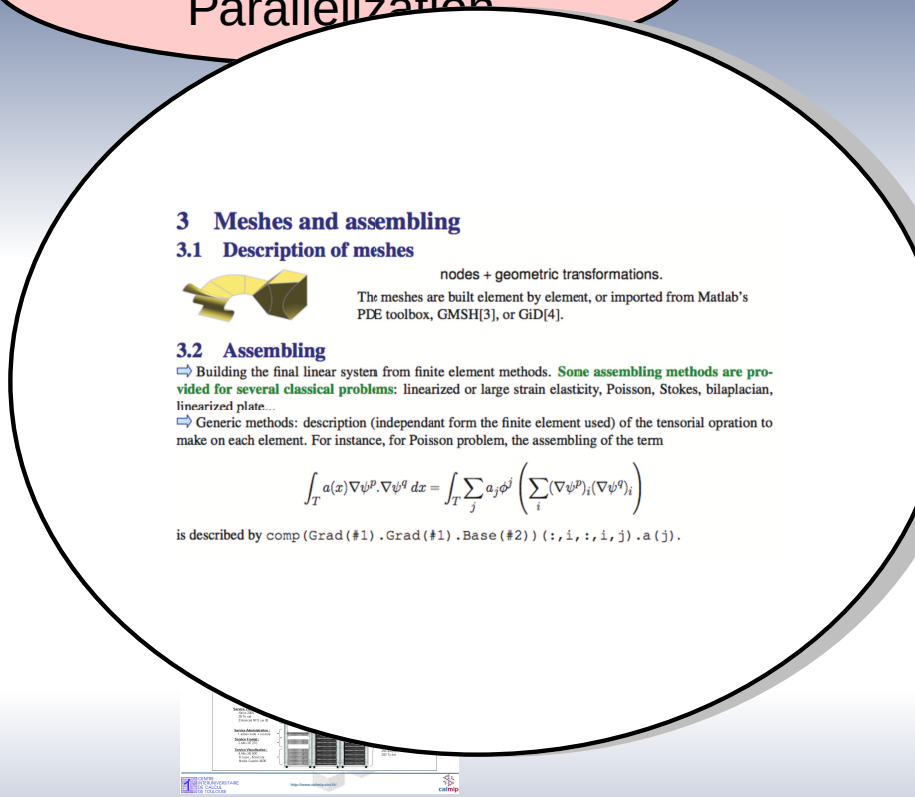
CFD
GETFEM++

HPC
Parallelization









3 Meshes and assembling

3.1 Description of meshes

nodes + geometric transformations.

The meshes are built element by element, or imported from Matlab's PDE toolbox, GMSH[3], or GiD[4].

3.2 Assembling

⇨ Building the final linear system from finite element methods. **Some assembling methods are provided for several classical problems**: linearized or large strain elasticity, Poisson, Stokes, bilaplacian, linearized plate...

⇨ Generic methods: description (independant form the finite element used) of the tensorial opration to make on each element. For instance, for Poisson problem, the assembling of the term

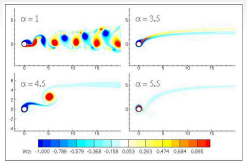
$$\int_T a(x)\nabla\psi^p.\nabla\psi^q\,dx = \int_T \sum_j a_j\phi^j \left(\sum_i (\nabla\psi^p)_i(\nabla\psi^q)_i\right)$$

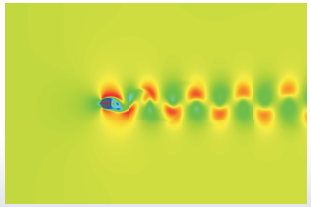
is described by `comp(Grad(#1).Grad(#1).Base(#2))(:,i,:,i,j).a(j).`

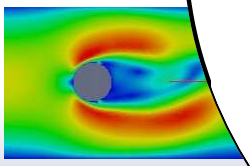
Local developments

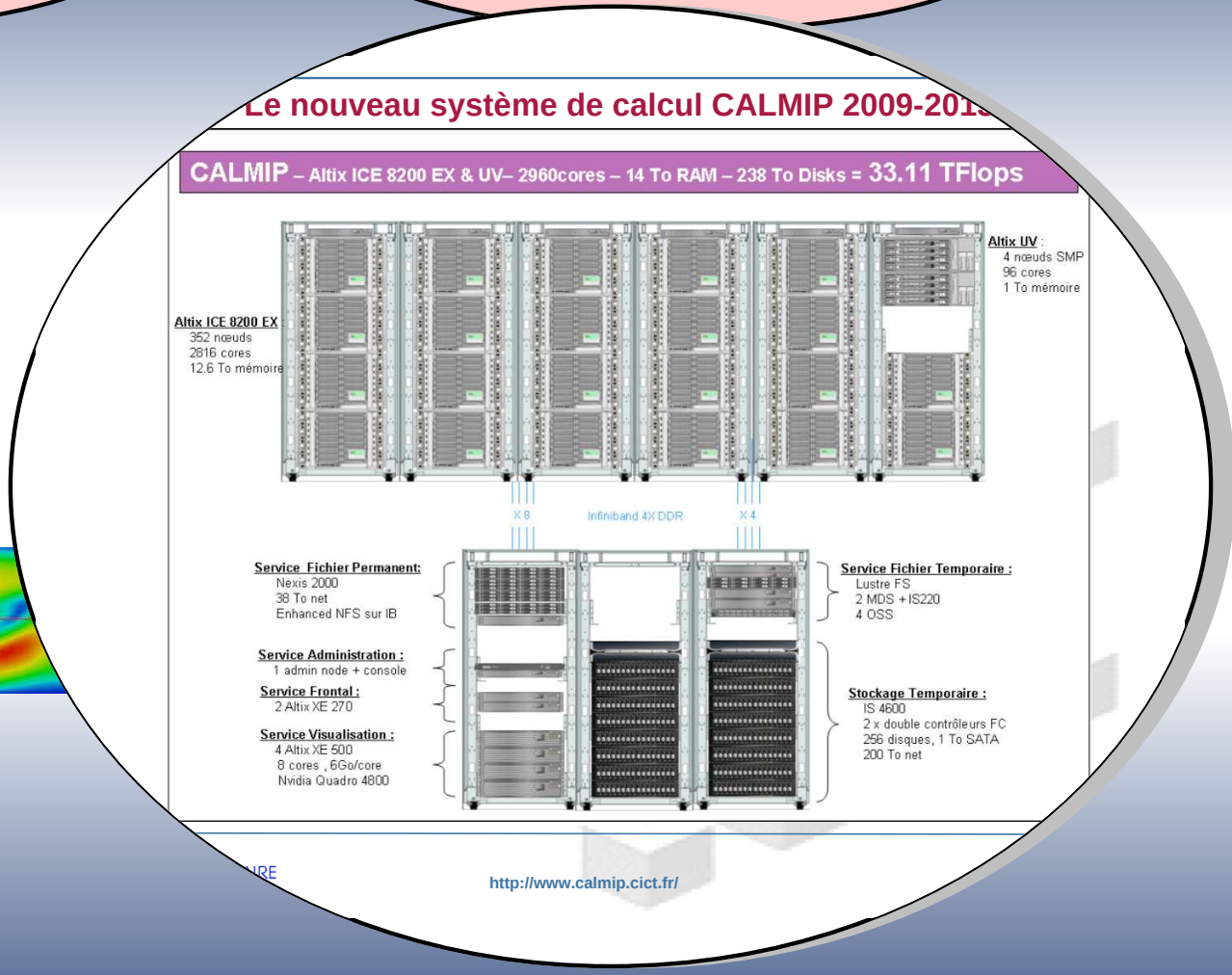
CFD
GETFEM++

HPC
Parallelization









Three aspects in the project

theoretical experimental **numerical**

The difficulties for numerical part

- complexity of the implementation for coupling model in 2D and 3D
- evolving boundary (interface)
- efficiency (robustness)
- accuracy
- CPU time

Three aspects in the project

theoretical experimental **numerical**

The difficulties for numerical part $\Longrightarrow$ Research in Fluid-Structure

- complexity of the implementation for coupling model in 2D and 3D
- evolving boundary (interface)
- efficiency (robustness)
- accuracy
- CPU time

Three aspects in the project

theoretical | experimental | **numerical**

The difficulties for numerical part

Fictitious domain based on Xfem method using Getfem++

- complexity of the implementation for coupling model in 2D and 3D
- evolving boundary (interface) $\Longrightarrow$ Level-set
- efficiency (robustness) $\Longrightarrow$ Local assembling
- accuracy $\Longrightarrow$ Optimal error
- CPU time $\Longrightarrow$ Parallel implementation (CALMIP)

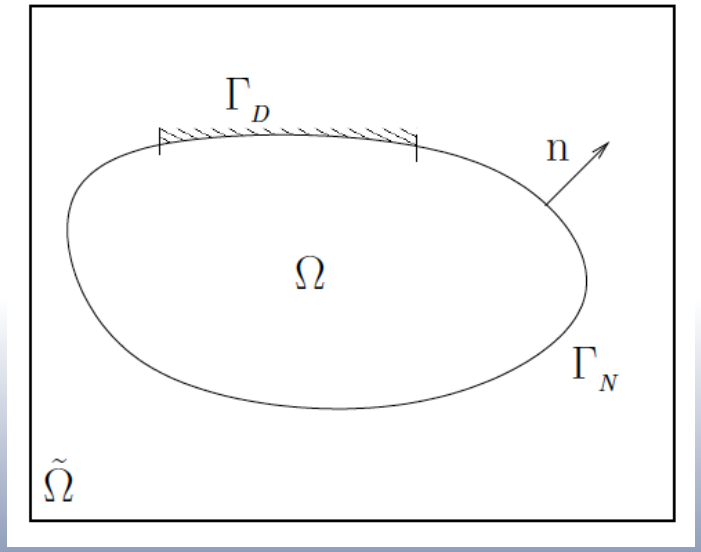
The model problem

- $\Omega \in \mathbb{R}^d$ (d=2 or d=3) the computational domain
- $\tilde{\Omega} \in \mathbb{R}^d$ rectangular or parallelepiped domain (the fictitious domain), $\Omega \subset \tilde{\Omega}$
- $\Gamma = \partial\Omega = \Gamma_D \cup \Gamma_N$ (Γ_D of nonzero measure, Γ_N can be $\emptyset$)

Fondamental problem

$$\begin{cases} \textit{Find } u : \Omega \longrightarrow \mathbb{R} \textit{ such that} \\ -\Delta u = f & \text{in } \Omega \\ u = 0 & \text{on } \Gamma_D \\ \partial_n u = g & \text{on } \Gamma_N \end{cases}$$

where $f \in L^2(\Omega)$, $g \in L^2(\Gamma_N)$ are given
n outward unit vector to Γ



Weak formulation

- Weak formulation

$$\begin{cases} \textit{Find } u \in V_0 \textit{ such that} \\ a(u,v) = l(v),\ \forall v \in V_0 \end{cases}$$

where

$$V = H^1(\Omega),\ V_0 = \{v \in V : v = 0 \text{ on } \Gamma_D\}$$

$$a(u,v) = \int_\Omega \nabla u.\nabla v d\Omega \ \textit{ and } \ l(v) = \int_\Omega f v d\Omega + \int_{\Gamma_N} g v d\Gamma$$

- Equivalent mixed formulation

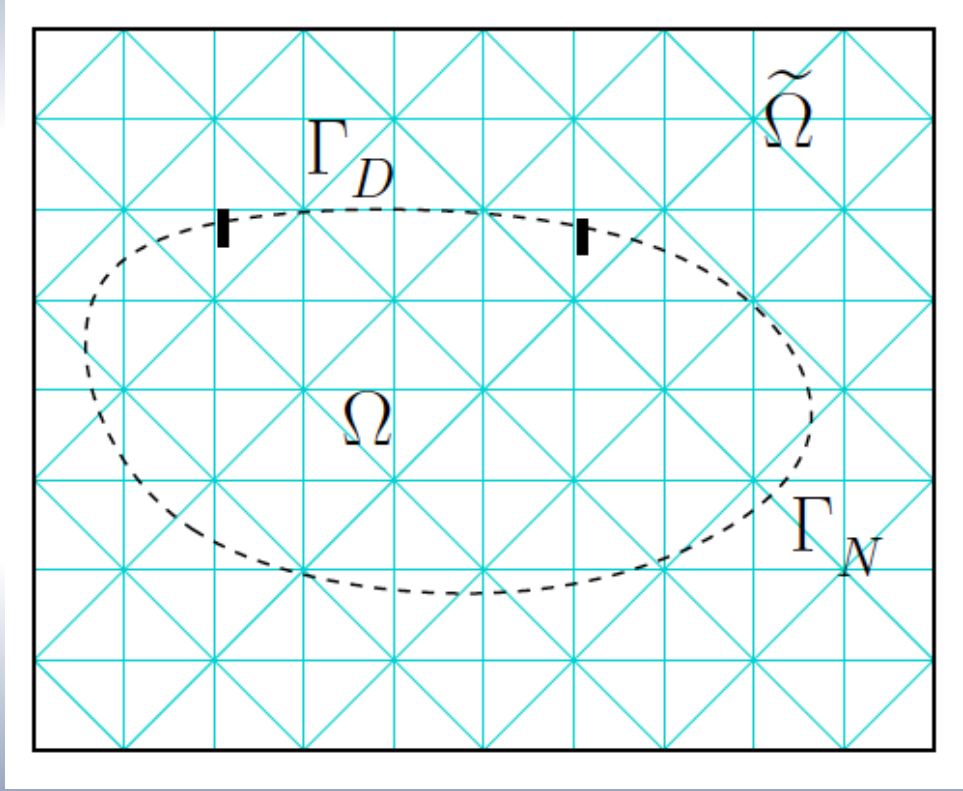
$$\begin{cases} \textit{Find } u \in V \textit{ and } \lambda \in W \textit{ such that} \\ a(u,v) + \int_{\Gamma_D} \lambda v d\Gamma = l(v),\ \forall v \in V \\ \int_{\Gamma_D} \mu u d\Gamma = 0,\ \forall \mu \in W \end{cases}$$

where $X = \left\{ w \in L^2(\Gamma_D) : \exists v \in V \textit{ such that } w = v_{|\Gamma_D} \right\}$ and $W = X'$

- Lagrange multiplier's interpretation : $\lambda = -\partial_n u$ on Γ_D

The mesh

- We define a regular mesh including the computational domain
- The boundary is characterized by Level-set



Fictitious domain approach inspired by Xfem

- On the fictitious domain $\tilde{\Omega}$,
  we consider two finite element spaces $\tilde{V}^h \subset H^1(\tilde{\Omega})$ and $\tilde{W}^h \subset L^2(\tilde{\Omega})$
- Then we define

$$\boxed{V^h := \tilde{V}^h_{|\Omega} \quad \text{and} \quad W^h := \tilde{W}^h_{|\Gamma_D}}$$

New approximation problem

$$\boxed{\begin{cases} \textit{Find } u^h \in V^h \textit{ and } \lambda^h \in W^h \textit{ such that} \\ a(u^h, v^h) + \displaystyle\int_{\Gamma_D} \lambda^h v^h d\Gamma = l(v^h), \ \forall v^h \in V^h \\ \displaystyle\int_{\Gamma_D} \lambda^h u^h d\Gamma = 0, \ \forall \mu^h \in W^h \end{cases}}$$

Convergence analysis [Haslinger & Renard (SIAM 2009)]

- Under technical conditions [1] the solution exists and is unique
- For $V^h = \left\{ v^h \in \mathcal{C}(\overline{\tilde{\Omega}}) : v^h_{|T} \in P_k(T)\ \forall T \in \mathcal{T}^h \right\}$ $(k \geq 1)$

$$\inf_{\mu^h \in W^h} \|\lambda - \mu^h\|_W \leq h^{\beta},\ \beta \geq \frac{1}{2}$$

- If $\Gamma = \Gamma_D$ and $u \in H^{1+\frac{d}{2}+\varepsilon}(\Omega) \cap H^1_0(\Omega)$, $\varepsilon > 0$

$$\|u - u^h\|_V \leq C h^{\frac{1}{2}},\ h \longrightarrow 0$$

Accuracy limitation in $\sqrt{h}$ for any k

1. $1_{|\Gamma_D} \in W^h$ *and* $\bar{\mu}^h \in W^h : \int_{\Gamma_D} \bar{\mu}^h v^h d\Gamma = 0,\ \forall v^h \in V^h \Longrightarrow \bar{\mu}^h = 0$

A stabilized problem [Barbosa & Hugues (CMAME 1991)]

- Initial problem (Find a saddle point of the Lagrangian on $V \times W$)

$$\mathcal{L}(v,\mu) = \frac{1}{2}a(v,v) + \int_{\Gamma_D} \mu v d\Gamma_D - l(v)$$

- Stabilized problem

$$\mathcal{L}_h(v^h,\mu^h) = \mathcal{L}(v^h,\mu^h) - \frac{\gamma}{2}\int_{\Gamma_D} \left(\mu^h + \partial_n v^h\right)^2 d\Gamma,\ v^h \in V^h,\ \mu^h \in W^h$$

where $\gamma := h\,\gamma_0$

A stabilized formulation

- Stabilized discret problem

$$\begin{cases} \textit{Find } u^h \in V^h \textit{ and } \lambda^h \in W^h \textit{ such that} \\ a(u^h, v^h) + \int_{\Gamma_D} \lambda^h v^h d\Gamma - \gamma \int_{\Gamma_D} \left(\lambda^h + \partial_n u^h\right) \partial_n v^h d\Gamma = l(v^h),\ v^h \in V^h \\ \int_{\Gamma_D} \mu^h u^h d\Gamma - \gamma \int_{\Gamma_D} \left(\lambda^h + \partial_n u^h\right) \mu^h d\Gamma = 0,\ \forall \mu^h \in W^h \end{cases}$$

- We define $\mathcal{B}_h : (V^h \times W^h)^2 \longrightarrow \mathbb{R}$ by

$$\mathcal{B}_h(u^h, \lambda^h; v^h, \mu^h) := a(u^h, v^h) + \int_{\Gamma_D} \lambda^h v^h d\Gamma + \int_{\Gamma_D} \mu^h u^h d\Gamma$$

$$-\gamma \int_{\Gamma_D} \left(\lambda^h + \partial_n u^h\right) \left(\mu^h + \partial_n v^h\right) d\Gamma$$

then the equivalent discret problem

$$\begin{cases} \textit{Find } u^h \in V^h \textit{ and } \lambda^h \in W^h \textit{ such that} \\ \mathcal{B}_h(u^h, \lambda^h; v^h, \mu^h) = l(v^h),\ \forall (v^h, \mu^h) \in V^h \times W^h \end{cases}$$

Convergence analysis [Haslinger & Renard (SIAM 2009)]

- Under the same assumptions and γ_0 sufficiently small an Inf-Sup condition is satisfied that ensures existence and uniqueness of the solution (using the norm $|||(z^h,\eta^h)|||^2 := \|z^h\|^2 + h^{-1}\|z^h\|^2_{0,\Gamma_D} + h\|\eta^h\|^2_{0,\Gamma_D}$)
- For Finite Element Method

$$\tilde{V}^h = \left\{ v^h \in \mathcal{C}(\tilde{\Omega}) : v^h_{|_T} \in P_{k_u}(T)\ \forall T \in \mathcal{T}^h \right\}, k_u \geq 1$$

$$\tilde{W}^h = \left\{ \mu^h \in L^2(\tilde{\Omega}) : \mu^h_{|_T} \in P_{k_\lambda}(T)\ \forall T \in \mathcal{T}^h \right\}, k_\lambda \geq 0$$

Under an assumption [2] on the intersection of the mesh with Ω, for a solution $u \in H^{k+1}(\Omega)$ and $\lambda \in H^{k-\frac{1}{2}}(\Gamma_D)$ where $k = min\{k_u, k_\lambda + 1\}$

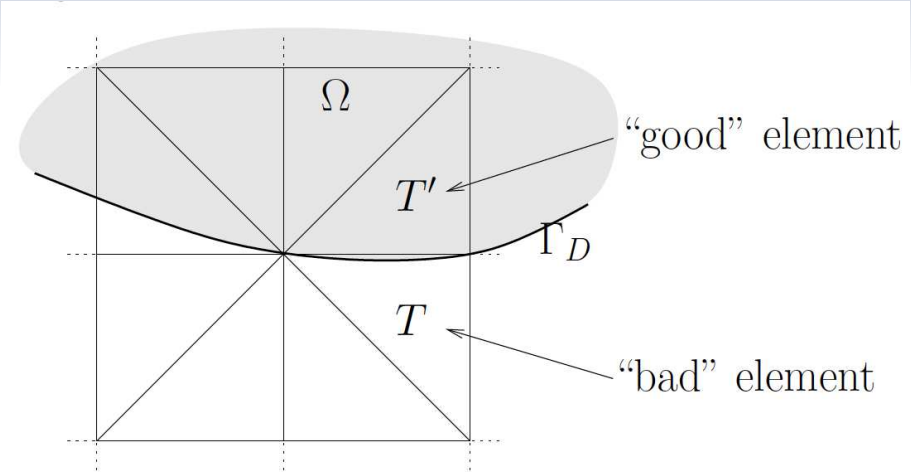
$$|||(u-u^h, \lambda-\lambda^h)||| \leq C h^k \|u\|_{k+1,\Omega}$$

No limitation
optimal order
under
mesh assumption

2. assumption required $h^{\frac{1}{2}}\|\partial_n v^h\|_{0,\Gamma_D} \leq C\|\nabla v^h\|_{0,\Omega},\ \ \forall v^h \in V^h,\ \forall h > 0$

The intersection of the mesh with Ω

- We suppose there exists a radius $\hat{\rho} > 0$ independent of h such that $\forall T \in \mathcal{T}^h$, $T \cap \Omega \neq \emptyset$ the reference element $\hat{T}$ contains a ball $B(\hat{y}_T, \hat{\rho})$ which satisfies $B(\hat{y}_T, \hat{\rho}) \subset \tau_T^{-1}(T \cap \Omega)$ where τ is a regular affine transformation in $\mathbb{R}^d$
- In the figure T is a "bad" element because its intersection with Ω is small.



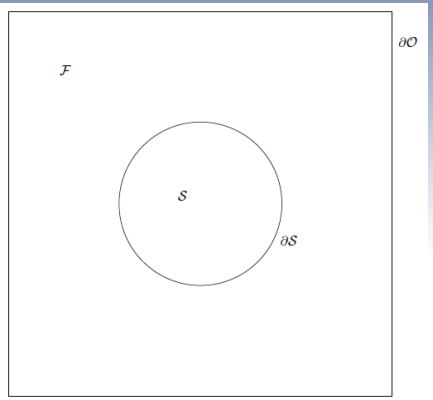
smaller is the thickness of the intersection, poorer approximation of the normal derivative on $T \cap \partial\Omega$ is obtained using $v^h_{|T}$

- We choice a neighbor element T' and evaluate the normal derivative from a natural extension of v^h from T' on T then no assumption required

Stokes problem

- Stokes problem (possible extension to Navier-Stokes)

$$\begin{cases} -\nu\Delta u + \nabla p &= f \quad \text{in } \mathcal{F} \\ div(u) &= 0 \quad \text{in } \mathcal{F} \\ u &= 0 \quad \text{on } \partial\mathcal{O} \\ u &= g \quad \text{on } \partial\mathcal{S} \end{cases}$$



- Augmented Lagrangian

$$\mathcal{L}(u,p,\lambda) = \mathcal{L}_0(u,p,\lambda) - \frac{\gamma}{2}\int_{\partial\mathcal{S}} |\lambda - \sigma(u,p)n|^2 d\Gamma$$

where

$$\mathcal{L}_0(u,p,\lambda) = \nu\int_{\mathcal{F}} |D(u)|^2 d\mathcal{F} - \int_{\mathcal{F}} p\,div(u) d\mathcal{F} - \int_{\mathcal{F}} f\cdot u d\Gamma - \int_{\partial\mathcal{S}} \lambda\cdot(u-g) d\Gamma$$

$\sigma(u,p)n = 2\nu D(u)n - pn$ and $D(u) = \frac{1}{2}\left(\nabla u + \nabla u^T\right)$

Fictitious domain

- The stabilized formulation of the problem is

$$\begin{cases} \text{Find } (u,p,\lambda) \in \mathbf{V} \times L^2_0(\mathcal{F}) \times \mathbf{H}^{1/2}(\partial\mathcal{S}) \text{ such that} \\ \mathcal{A}((u,p,\lambda);v) = \mathcal{L}(v) \quad \forall v \in \mathbf{V}_0, \\ \mathcal{B}((u,p,\lambda);q) = 0 \quad \forall q \in L^2_0(\mathcal{F}), \\ \mathcal{C}((u,p,\lambda);\mu) = \mathcal{G}(\mu), \quad \forall \mu \in \mathbf{H}^{-1/2}(\partial\mathcal{S}), \end{cases}$$

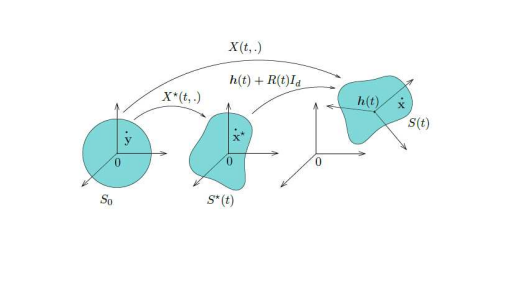
where

$$\begin{aligned} \mathcal{A}((u,p,\lambda);v) &= 2\nu \int_{\mathcal{F}} D(u) : D(v) d\mathcal{F} - \int_{\mathcal{F}} p\,div(v) d\mathcal{F} - \int_{\partial\mathcal{S}} \lambda \cdot v d\Gamma \\ &\quad -4\nu^2\gamma \int_{\partial\mathcal{S}} (D(u)n)\cdot(D(v)n)\, d\Gamma + 2\nu\gamma \int_{\partial\mathcal{S}} p\,(D(v)n\cdot n)\, d\Gamma + 2\nu\gamma \int_{\partial\mathcal{S}} \lambda\cdot(D(v)n)\, d\Gamma, \\ \mathcal{B}((u,p,\lambda);q) &= -\int_{\mathcal{F}} q\,div(u) d\mathcal{F} + 2\nu\gamma \int_{\partial\mathcal{S}} q\,(D(u)n\cdot n)\, d\Gamma - \gamma \int_{\partial\mathcal{S}} pq d\Gamma - \gamma \int_{\partial\mathcal{S}} q\lambda\cdot n d\Gamma, \\ \mathcal{C}((u,p,\lambda);\mu) &= -\int_{\partial\mathcal{S}} \mu\cdot u d\Gamma + 2\nu\gamma \int_{\partial\mathcal{S}} \mu\cdot(D(u)n) d\Gamma - \gamma \int_{\partial\mathcal{S}} p(\mu\cdot n) d\Gamma - \gamma \int_{\partial\mathcal{S}} \lambda\cdot\mu d\Gamma. \end{aligned}$$

- the extension from Laplacian to Stokes and Navier-Stokes is in progress

Fluid-Structure interaction

- Moving solid occupying a time-depending domain $\mathcal{S}(t)$
- The displacement of a rigid solid is given by



$$\begin{aligned} X(y,t) &= h(t)+\mathbf{R}(t)y, \quad y\in\mathcal{S}(0), \\ \mathcal{S}(t) &= h(t)+\mathbf{R}(t)\mathcal{S}(0), \end{aligned}$$

where $h(t)$ is the gravity center and $\mathbf{R}(t)$ the rotation $\begin{pmatrix} \cos(\theta(t)) & -\sin(\theta(t)) \\ \sin(\theta(t)) & \cos(\theta(t)) \end{pmatrix}$

- Find u, p, $h(t)$ and its angular velocity $\theta'(t)=\omega(t)$ (a scalar in 2D)
- Coupling at the interface $\partial\mathcal{S}$ using the Dirichlet condition

$$u(x,t) = h'(t)+\theta'(t)(x-h(t)), \quad x\in\partial\mathcal{S}(t),$$

with

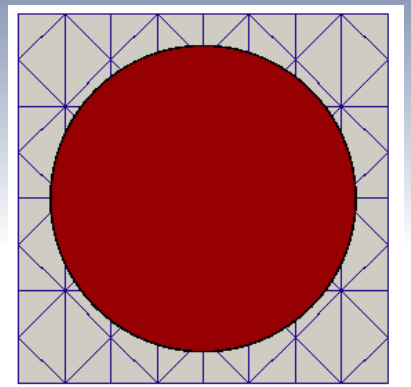
$$\begin{aligned} Mh''(t) &= -\int_{\partial\mathcal{S}(t)} \sigma(u,p)n\,d\Gamma \\ I\theta''(t) &= -\int_{\partial\mathcal{S}(t)} (x-h(t))\wedge\sigma(u,p)n\,d\Gamma \end{aligned}$$

accurate computation of $\sigma(u,p)n$ is crutial

Numerical results

- Regular mesh

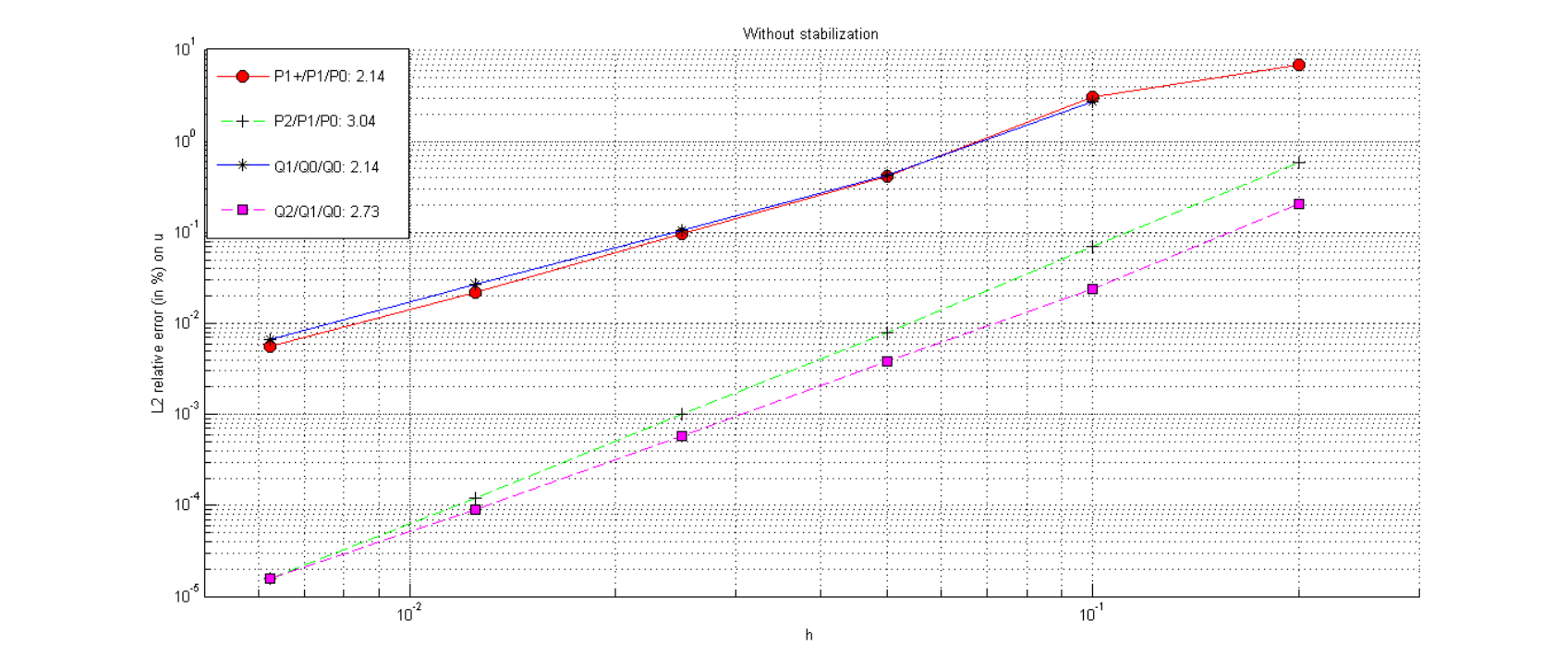
  Finite Element Methods
  P1+/P1/P1
  P2/P1/P1
  Q1/Q0/Q0
  Q2/Q1/Q0



- $\Omega = [0,1] \times [0,1]$ and $\partial\mathcal{S}$ the circle $(x-0.5)^2 + (y-0.5)^2 = R^2$ with $R = 0.21$
- The exact solutions for Stokes problem

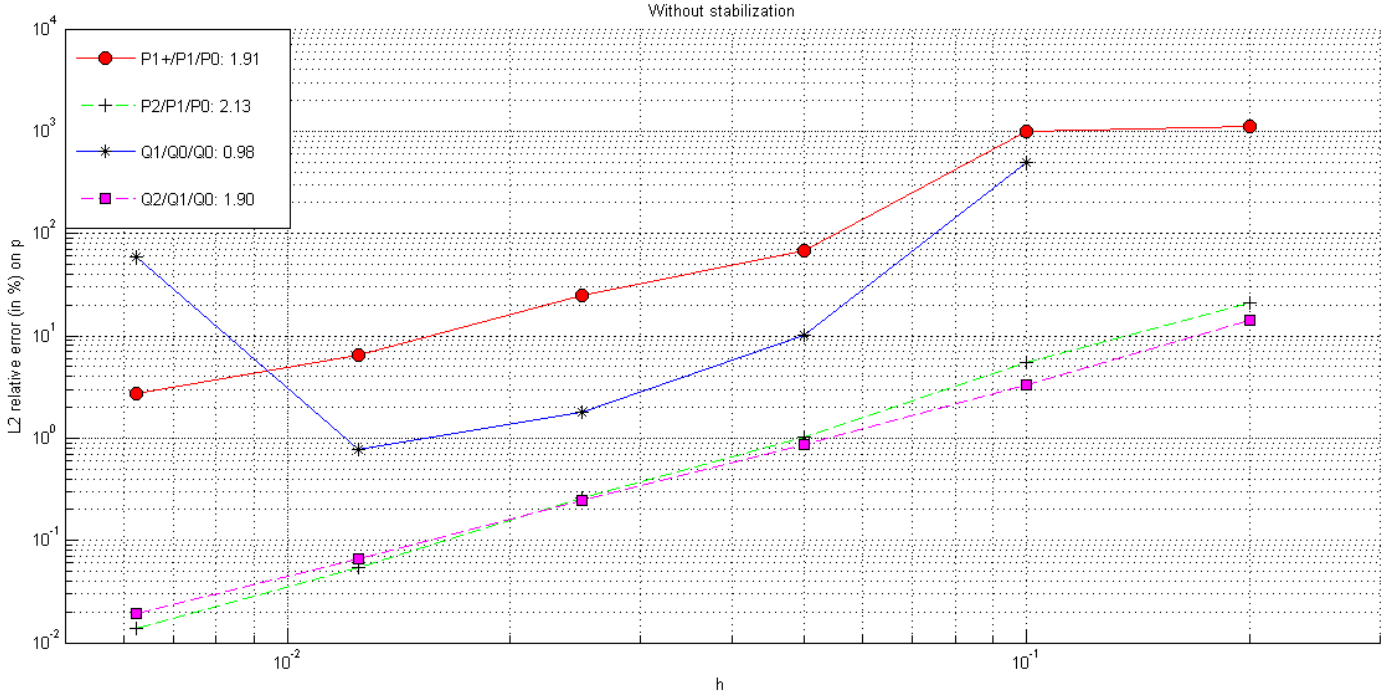
$$
\begin{aligned}
u_{ex}(x,y) &= \begin{pmatrix} \cos(\pi x)\sin(\pi y) \\ -\sin(\pi x)\cos(\pi y) \end{pmatrix} \\
p_{ex}(x,y) &= (y-0.5)\cos(2\pi x) + (x-0.5)\sin(2\pi y)
\end{aligned}
$$

Rates of convergence without stabilization *u*



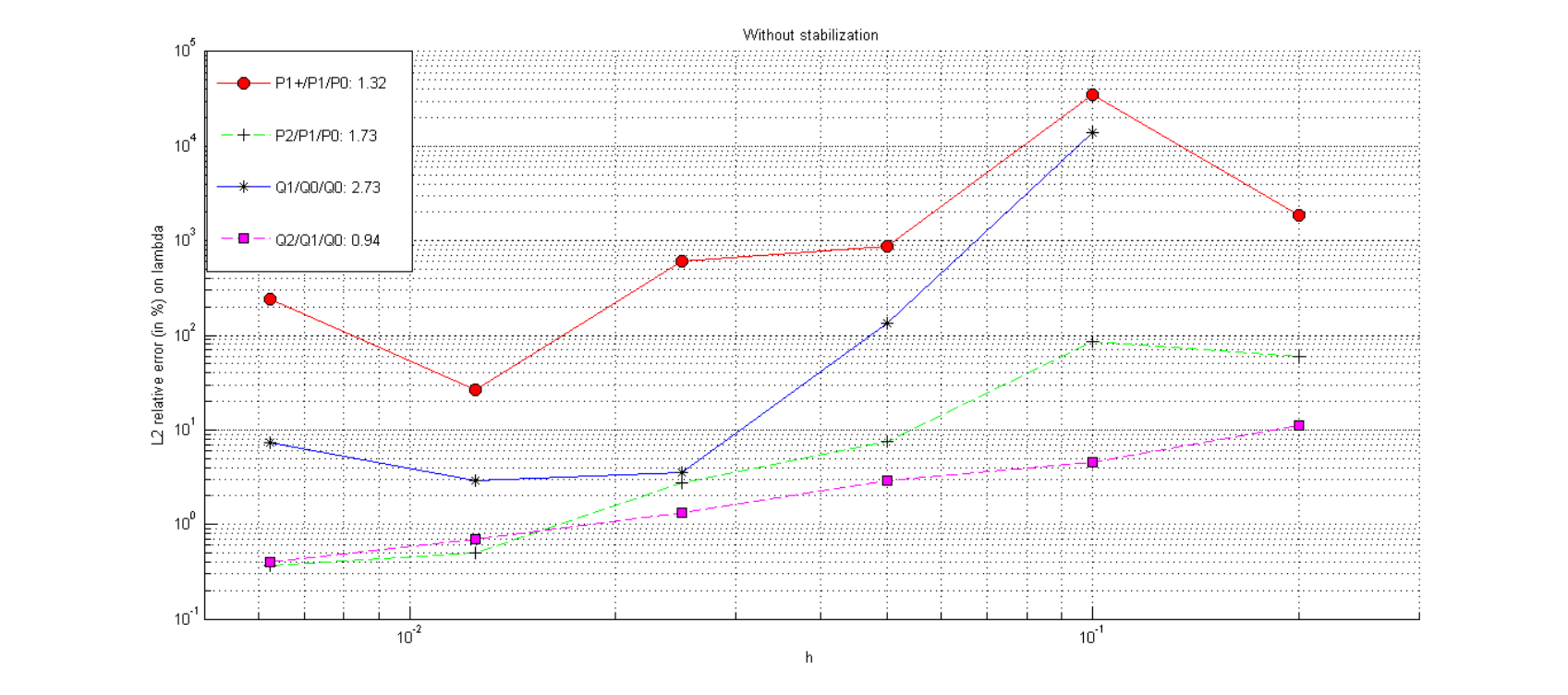
$$\|u - u^h\|_{\mathbf{L}^2(\mathcal{F})}$$

Rates of convergence without stabilization p



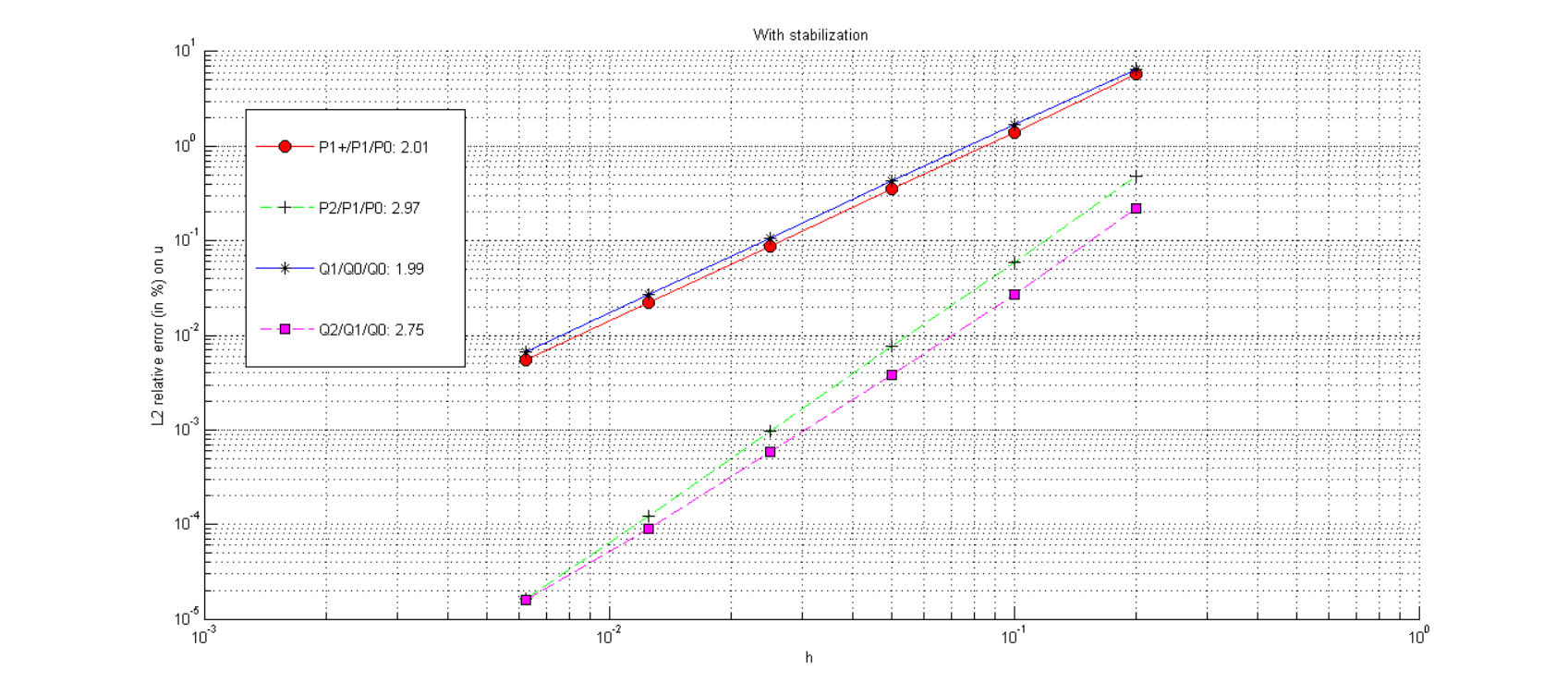
$$\|p - p^h\|_{L^2(\mathcal{F})}$$

Rates of convergence without stabilization λ



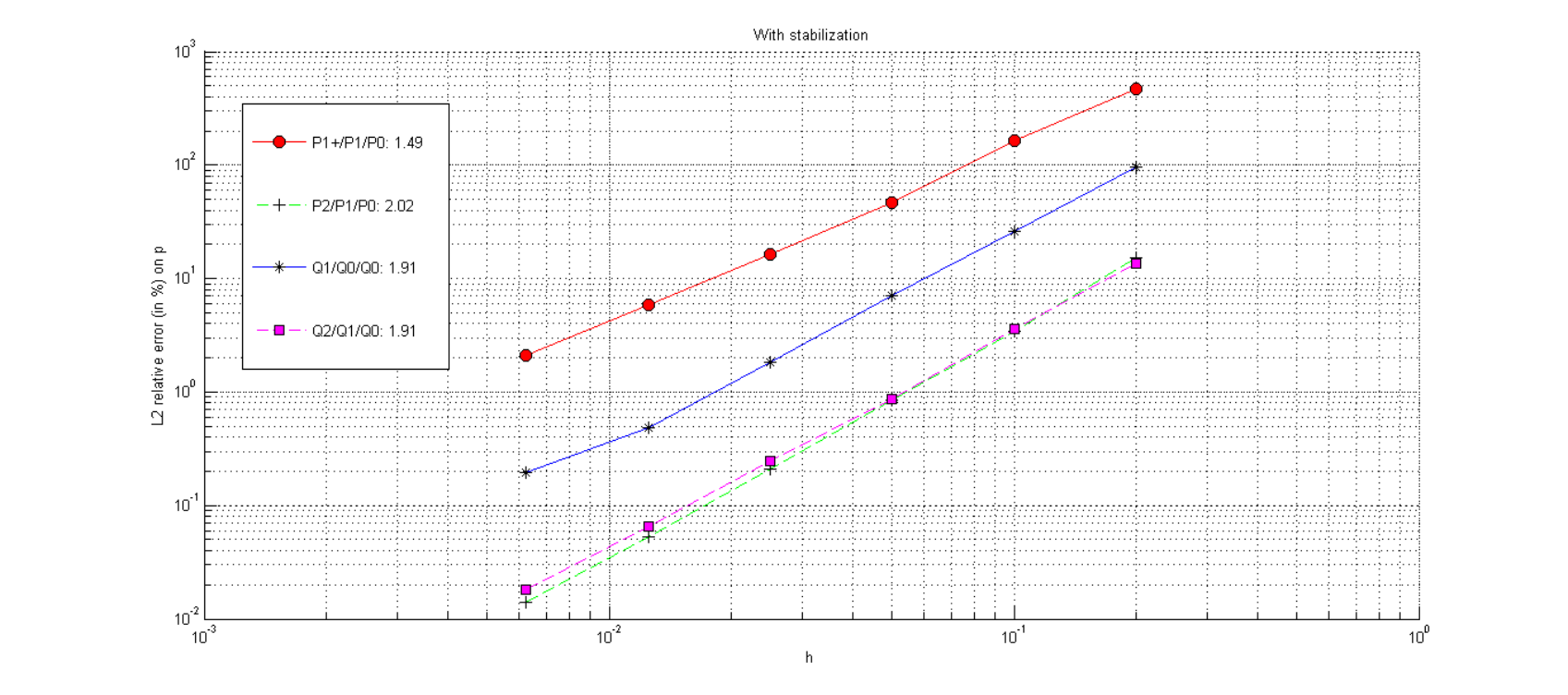
$$\|\lambda - \lambda^h\|_{\mathbf{L}^2(\partial\mathcal{S})}$$

Rates of convergence with stabilization u



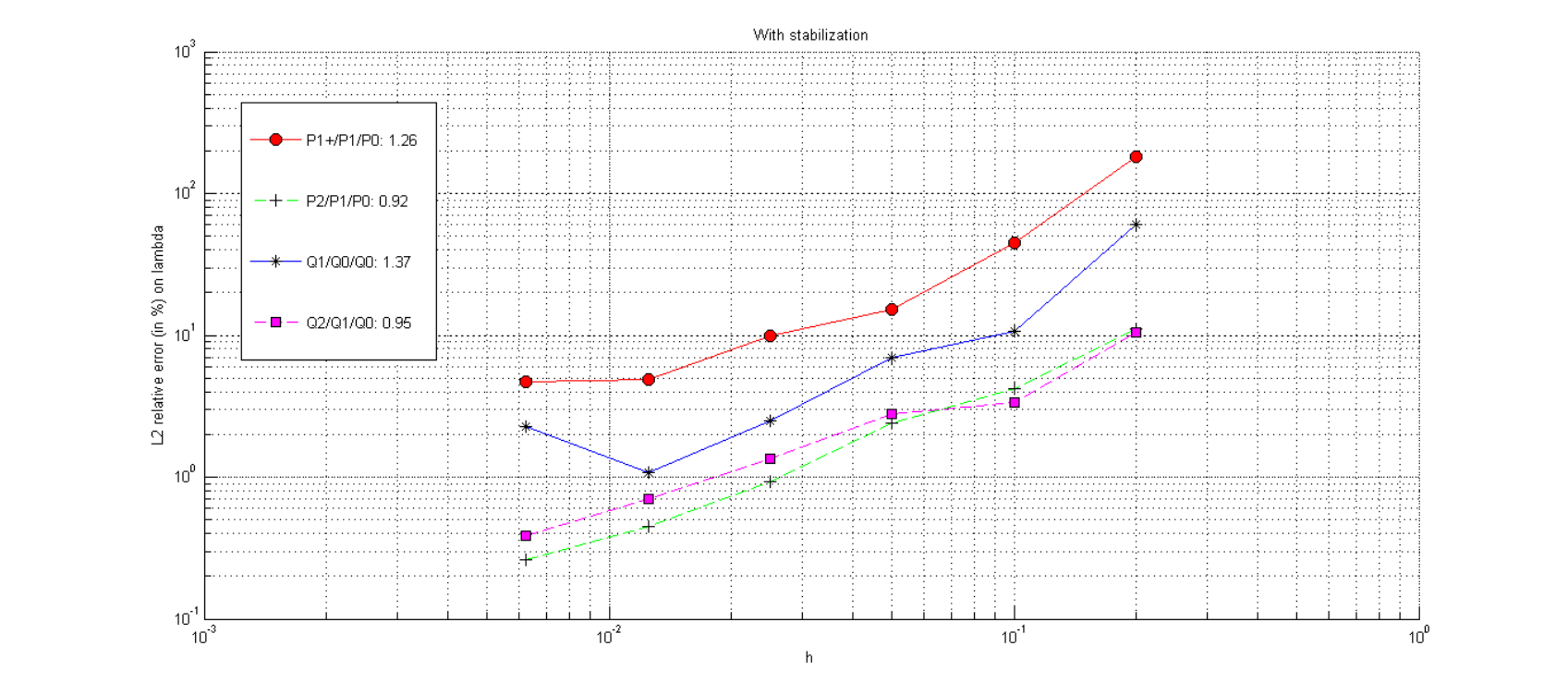
$$\|u - u^h\|_{\mathbf{L}^2(\mathcal{F})}$$

Rates of convergence with stabilization p



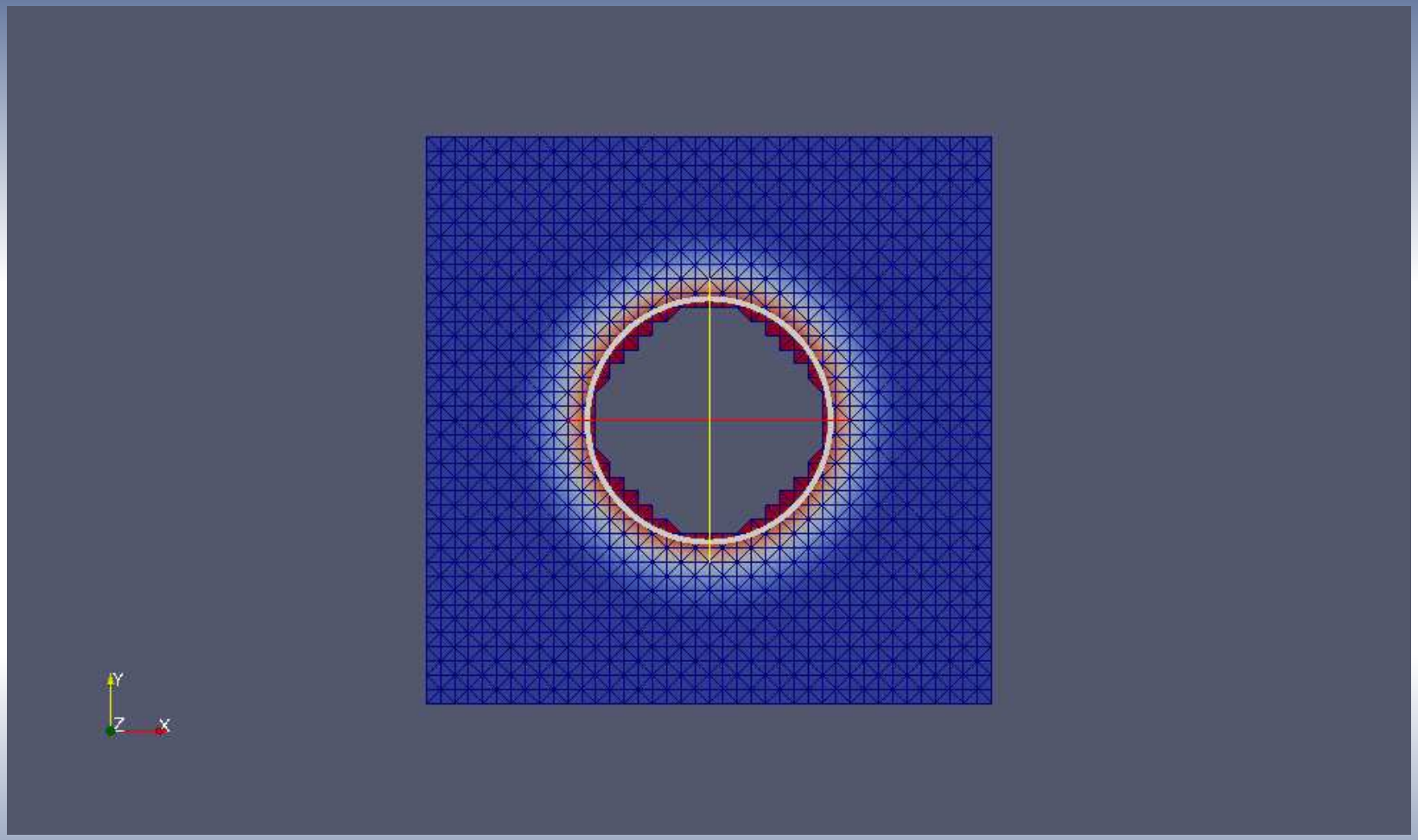
$$\|p - p^h\|_{\mathbf{L}^2(\mathcal{F})}$$

Rates of convergence with stabilization λ



$$\|\lambda - \lambda^h\|_{\mathbf{L}^2(\partial\mathcal{S})}$$

Numerical tests for Fluid-Structure



- Getfem++ is a generic C++ Finite Element library that allows us to consider an independent implementation from de dimension of the problem 2D or 3D (at least for classical model)
  $\Longrightarrow$ validations of the generic programation "must" be done
  $\Longrightarrow$ tests "must" be done to study the memory required

- The development of the Getfem++ library for CFD, on HPC is in perpetual improvement (parallelization aspect is developed on Calmip)
  $\Longrightarrow$ parallel tests "must" be done to verify the full parallelization

- Efficiency of the fictitious method : to approximate evolving boundary (fluid-structure) the assembling "must" be actualized locally near the interface
  $\Longrightarrow$ post-doc (time programation) is reserved in the project

State of our work

- Implementation and validation of the stationary Stokes problem — YES
  - fictitious domain without stabilization — YES
  - fictitious domain with stabilization — YES
  - fictitious domain with stabilization and choice of optimal triangle — YES
- Implementation of the time dependent Stokes problem — YES
  - fictitious domain without stabilization — IN PROGRESS
  - fictitious domain with stabilization — IN PROGRESS
  - fictitious domain with stabilization and choice of optimal triangle — IN PROGRESS
- Fluid-Structure (using Stokes model) — IN PROGRESS
- Navier-Stokes — NO
- Fluid-Structure (using Navier-Stokes model) — NO
- The coupling (structure model) — NO