THE DOMAIN DECOMPOSITION :
A solution method

Colloque en l'honneur d'Alain DERVIEUX
April 10 th 2015, Sophia-Antipolis

Marina Vidrascu
Project team REO
Inria Paris-Rocquencourt & Laboratoire J-L Lions
marina.vidrascu@inria.fr
& Collaborators!!!!

What is Domain Decomposition ?

Data decomposition

- ★ Parallel computing: distribute data among processors
- ★ Do not change the solution algorithm

Homogenous domain decomposition

- ★ New solution method for large scale linear systems
- ★ Well adapted for coarse grain parallel methods
- ★ Principle : solve independently on each domain and glue solutions at interface

Heterogenous domain decomposition

- ★ Separate the physical domain in regions
- ★ Use a different model in each region
- ★ Glue solutions at interface

What is Domain Decomposition ?

Data decomposition

★ Parallel computing: distribute data among processors
★ Do not change the solution algorithm

Homogenous domain decomposition

★ New solution method for large scale linear systems
★ Well adapted for coarse grain parallel methods
★ Principle : solve independently on each domain and glue solutions at interface

Heterogenous domain decomposition

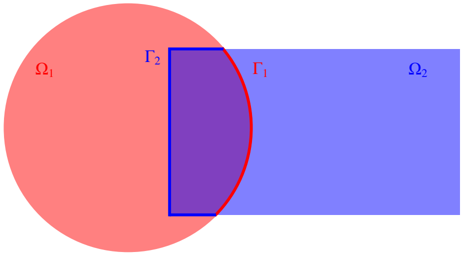
★ Separate the physical domain in regions
★ Use a different model in each region
★ Glue solutions at interface

FSI

Un peu d'histoire

Schwarz alterné

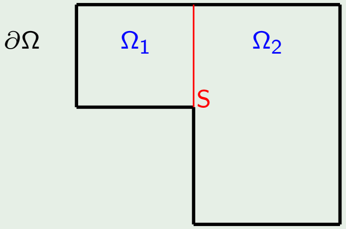
- décomposition avec recouvrement
- calcul **analytique** de fonctions harmoniques



Sous-structuration

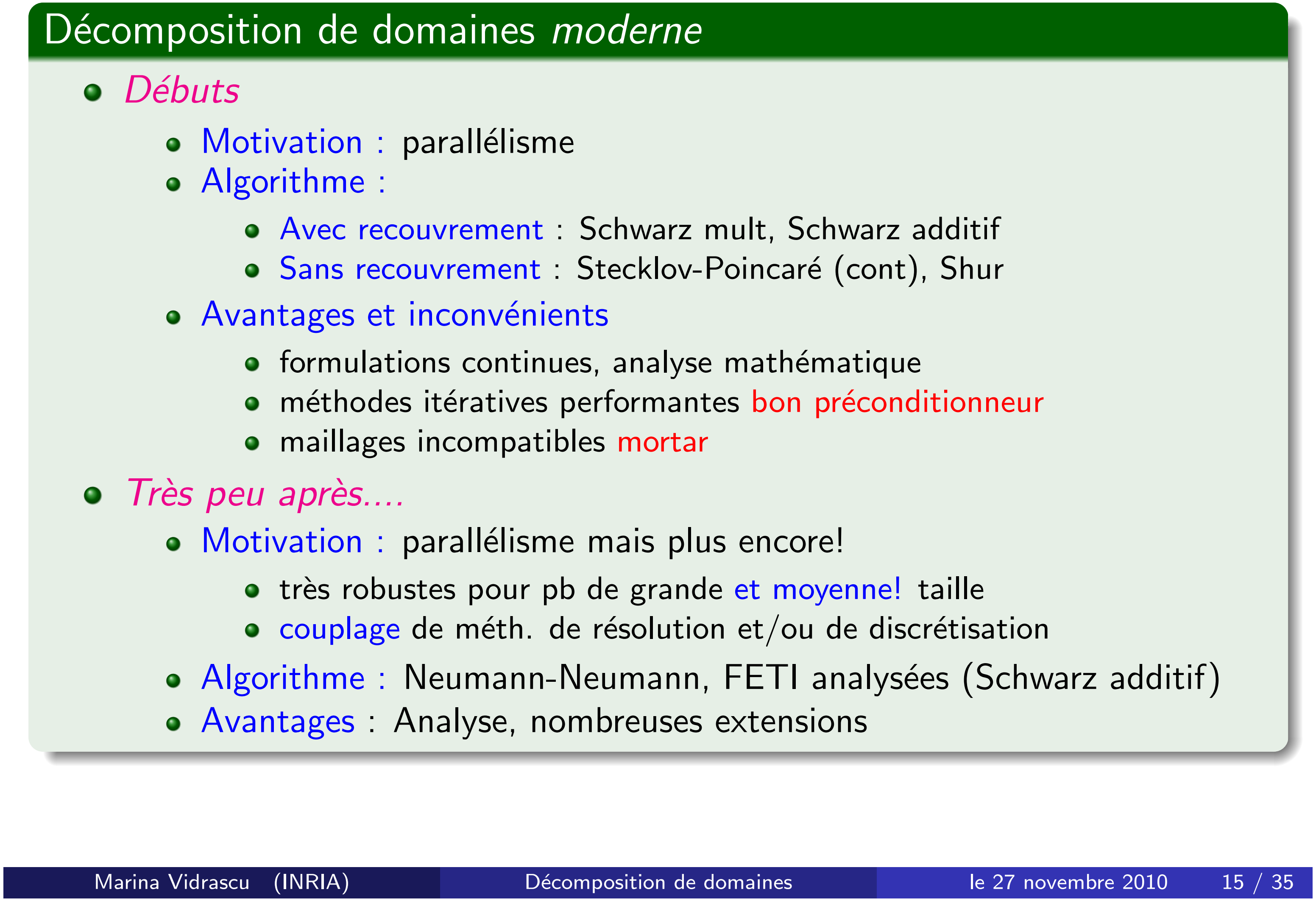
Numérotation : premier sous-domaine, deuxième , la frontière S

$$\begin{bmatrix} A_{11} & 0 & A_{1S} \\ 0 & A_{22} & A_{2S} \\ A_{1S}^T & A_{2S}^T & A_{SS} \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_S \end{bmatrix} = \begin{bmatrix} B_1 \\ B_2 \\ B_S \end{bmatrix}$$



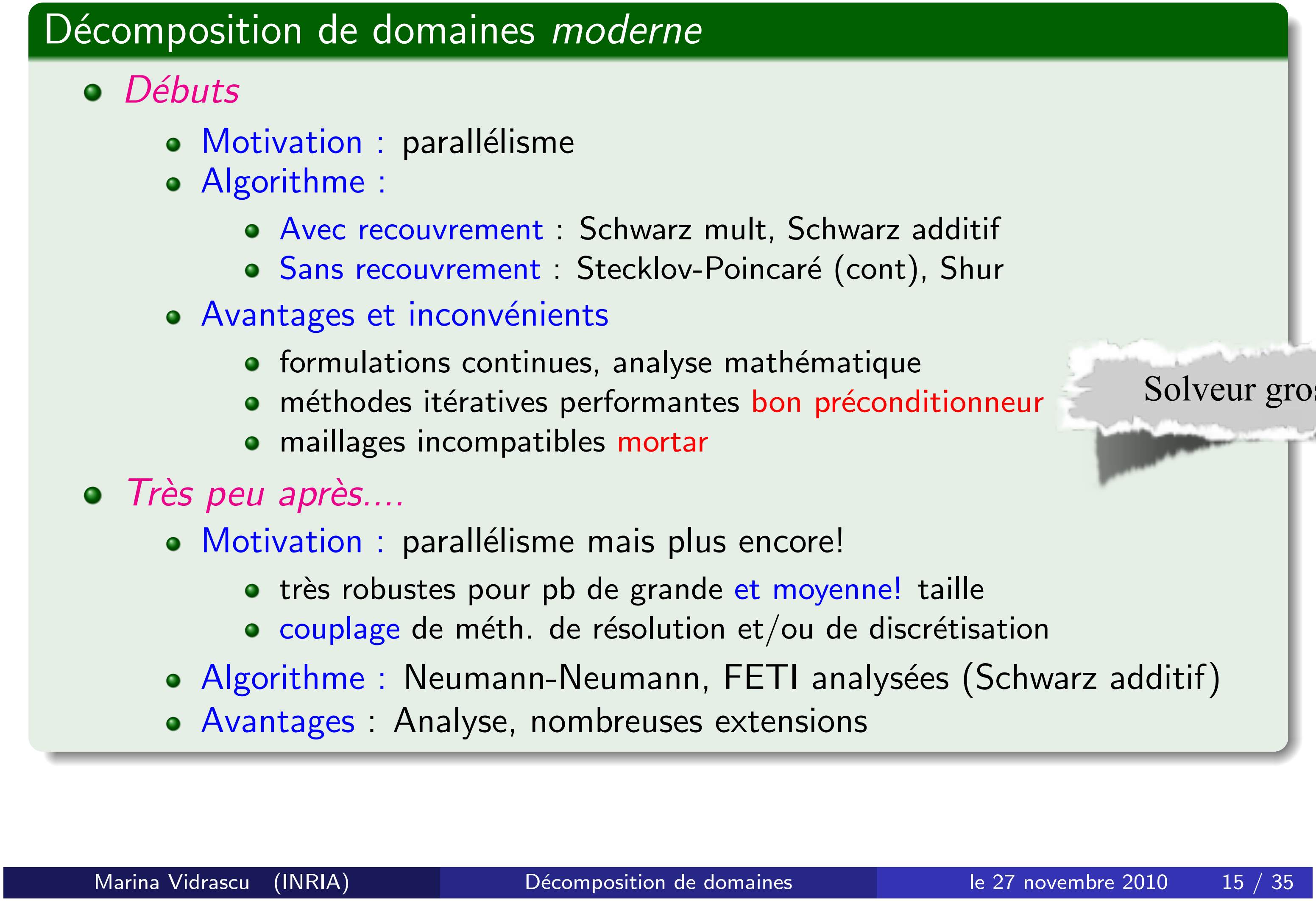
$$X_i = A_{ii}^{-1}\,(B_i - A_{iS}\,X_S)$$

$$\underbrace{(A_{SS} - \sum_{i=1}^{2} A_{iS}^T A_{ii}^{-1} A_{iS})}_{S\text{(Shur complement)}} X_S = B_S - \sum_{i=1}^{2} A_{iS}^T A_{ii}^{-1} B_i$$



Décomposition de domaines *moderne*

- *Débuts*
  - Motivation : parallélisme
  - Algorithme :
    - Avec recouvrement : Schwarz mult, Schwarz additif
    - Sans recouvrement : Stecklov-Poincaré (cont), Shur
  - Avantages et inconvénients
    - formulations continues, analyse mathématique
    - méthodes itératives performantes bon préconditionneur
    - maillages incompatibles mortar
- *Très peu après....*
  - Motivation : parallélisme mais plus encore!
    - très robustes pour pb de grande et moyenne! taille
    - couplage de méth. de résolution et/ou de discrétisation
  - Algorithme : Neumann-Neumann, FETI analysées (Schwarz additif)
  - Avantages : Analyse, nombreuses extensions

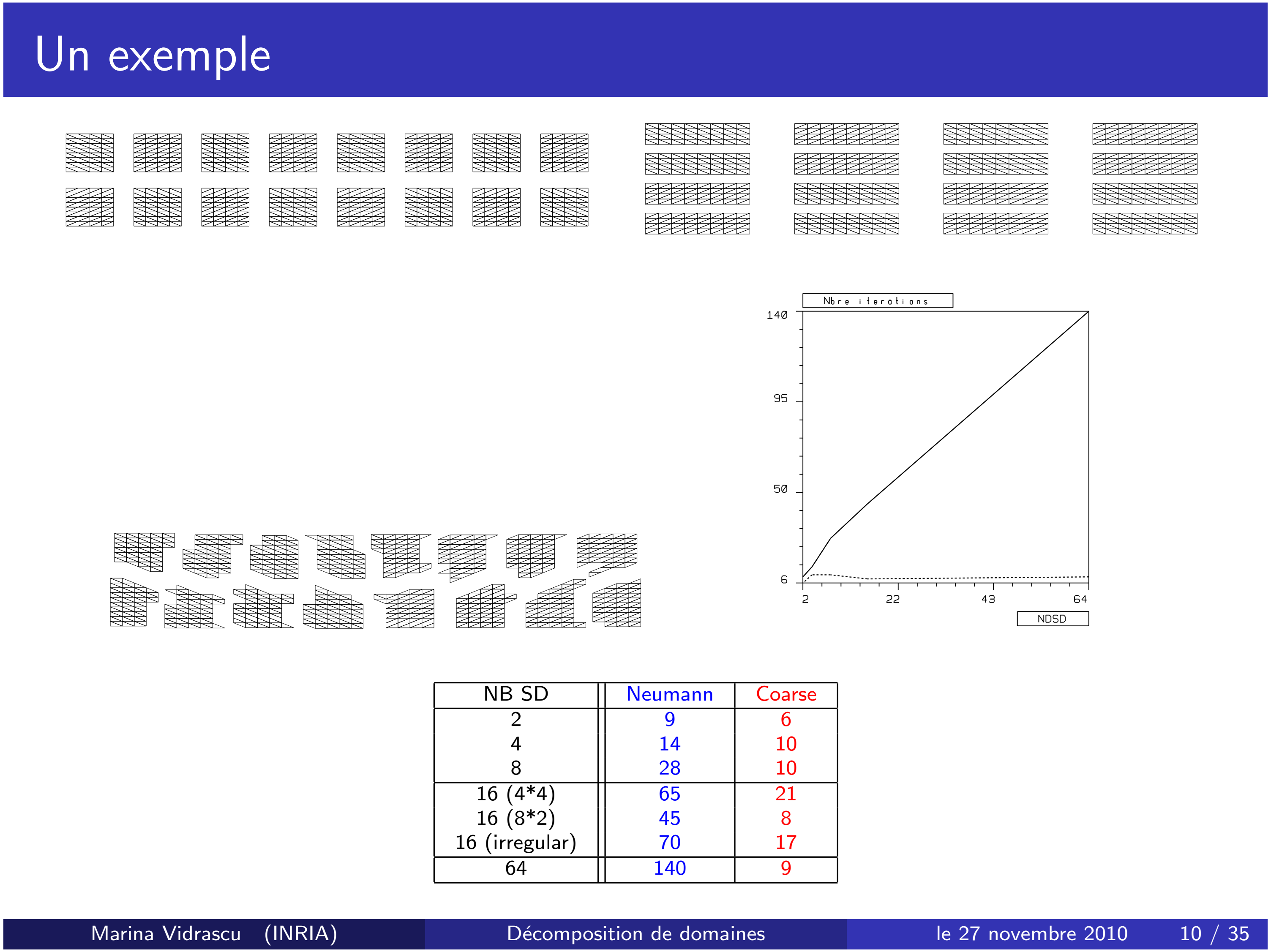


Décomposition de domaines *moderne*

- *Débuts*
  - Motivation : parallélisme
  - Algorithme :
    - Avec recouvrement : Schwarz mult, Schwarz additif
    - Sans recouvrement : Stecklov-Poincaré (cont), Shur
  - Avantages et inconvénients
    - formulations continues, analyse mathématique
    - méthodes itératives performantes bon préconditionneur
    - maillages incompatibles mortar
- *Très peu après....*
  - Motivation : parallélisme mais plus encore!
    - très robustes pour pb de grande et moyenne! taille
    - couplage de méth. de résolution et/ou de discrétisation
  - Algorithme : Neumann-Neumann, FETI analysées (Schwarz additif)
  - Avantages : Analyse, nombreuses extensions

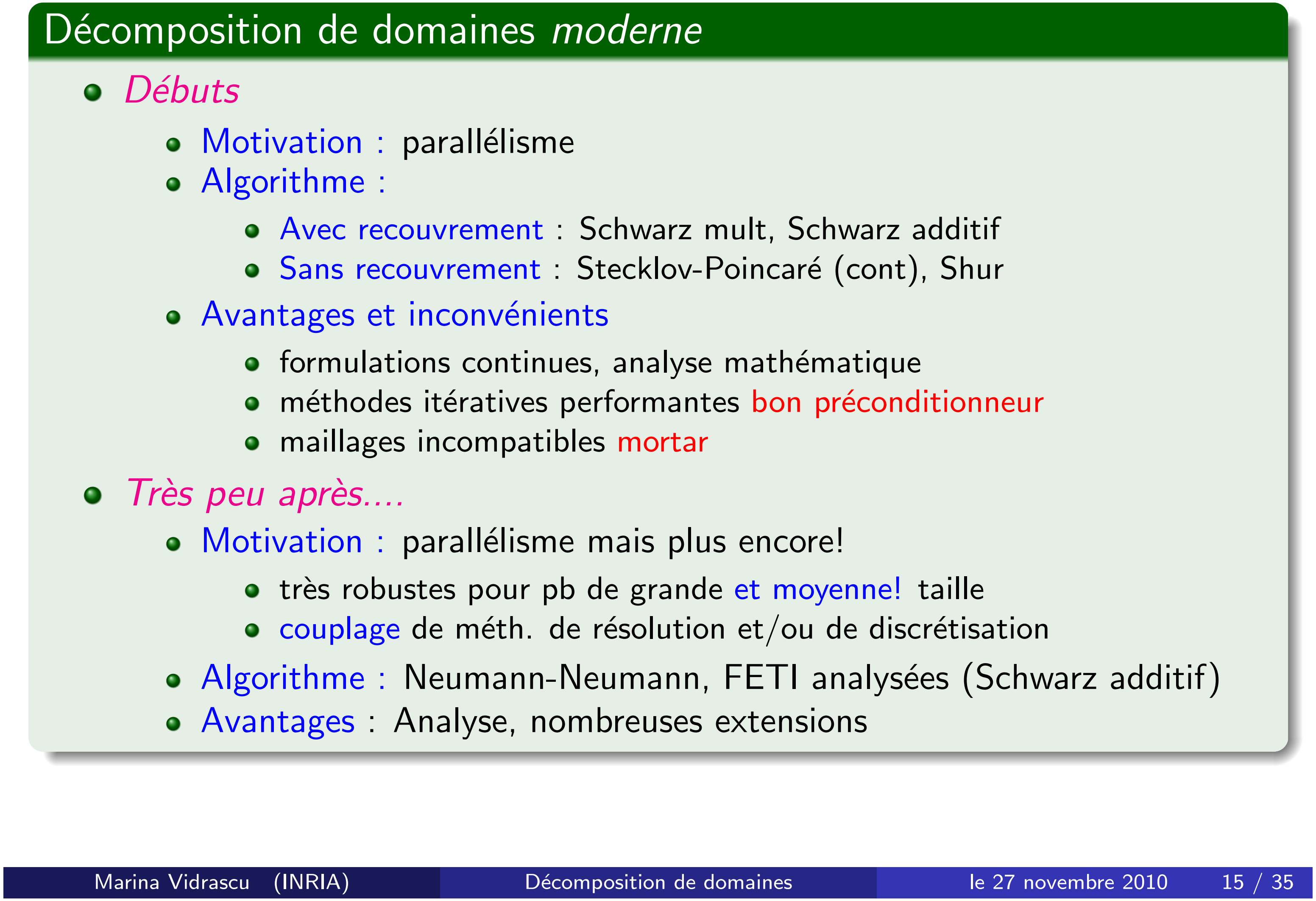
Marina Vidrascu (INRIA) Décomposition de domaines le 27 novembre 2010 15 / 35

Solveur grossier



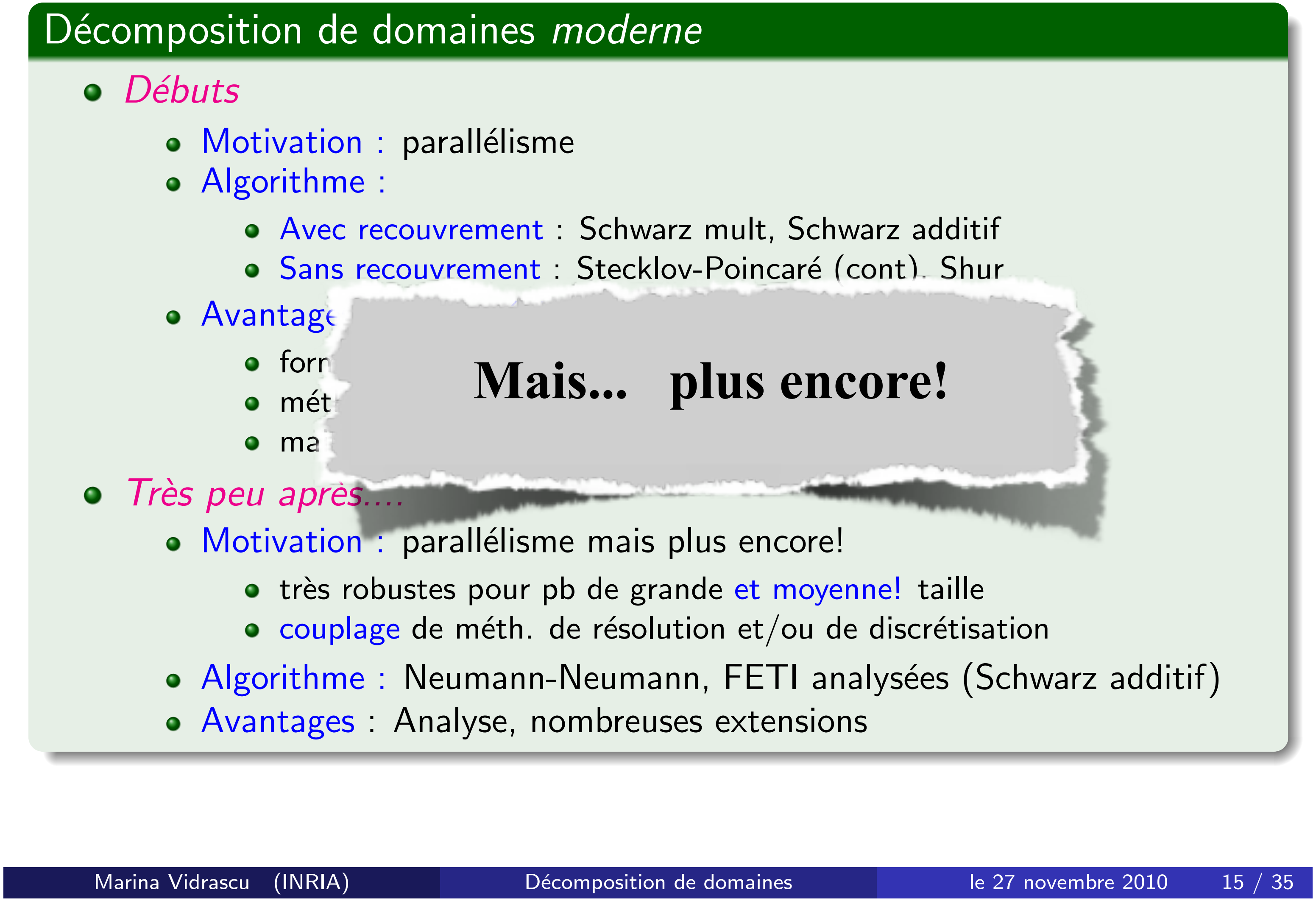
Un exemple

NB SD	Neumann	Coarse
2	9	6
4	14	10
8	28	10
16 (4*4)	65	21
16 (8*2)	45	8
16 (irregular)	70	17
64	140	9



Décomposition de domaines *moderne*

- *Débuts*
  - Motivation : parallélisme
  - Algorithme :
    - Avec recouvrement : Schwarz mult, Schwarz additif
    - Sans recouvrement : Stecklov-Poincaré (cont), Shur
  - Avantages et inconvénients
    - formulations continues, analyse mathématique
    - méthodes itératives performantes bon préconditionneur
    - maillages incompatibles mortar
- *Très peu après....*
  - Motivation : parallélisme mais plus encore!
    - très robustes pour pb de grande et moyenne! taille
    - couplage de méth. de résolution et/ou de discrétisation
  - Algorithme : Neumann-Neumann, FETI analysées (Schwarz additif)
  - Avantages : Analyse, nombreuses extensions



Décomposition de domaines *moderne*

- *Débuts*
  - Motivation : parallélisme
  - Algorithme :
    - Avec recouvrement : Schwarz mult, Schwarz additif
    - Sans recouvrement : Stecklov-Poincaré (cont), Shur
  - Avantage
    - forr
    - mét
    - ma

Mais... plus encore!

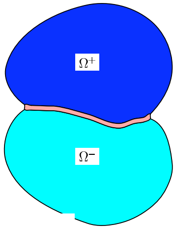
- *Très peu après....*
  - Motivation : parallélisme mais plus encore!
    - très robustes pour pb de grande et moyenne! taille
    - couplage de méth. de résolution et/ou de discrétisation
  - Algorithme : Neumann-Neumann, FETI analysées (Schwarz additif)
  - Avantages : Analyse, nombreuses extensions

DD a tool for practical solution of asymptotic problems

Multi-scale elasticity problems

★A *sandwich* structure with a thin layer. High ratio in material properties

- $E_l << E_{3d}$ **glue**
- $E_l >> E_{3d}$ **reinforcement sheets**



★ With heterogeneities

Modeling

direct solution of the problem is not efficient

★poor conditioning
★large size (the element size same order as the thickness of the layer)

asymptotic study : **eliminate** the thin layer and replace it with ad hoc transition conditions on the interface

Soft layer

(G.Geymonat, F. Krasucki, D. Marini et M. V, 1996)

Goland and Reissner conditions

$$\left\{\begin{array}{lcl} \sigma^+ n^+ & = & -\sigma^- n^- \\ \sigma^+ n^+ & = & -\frac{K^s}{h}\,[u] \end{array}\right.$$

Fourier Robin conditions

$$\left\{\begin{array}{lcl} \sigma^+ n^+ 2\frac{K^s}{h} u^+ & = & \sigma^- n^- + 2\frac{K^s}{h} u^+ \\ \sigma^+ n^+ & = & -\sigma^- n^- \end{array}\right.$$

Soft layer

(G.Geymonat, F. Krasucki, D. Marini et M. V, 1996)

Goland and Reissner conditions

$$\left\{\begin{array}{lcl} \sigma^+ n^+ & = & -\sigma^- n^- \\ \sigma^+ n^+ & = & -\frac{K^s}{h}[u] \end{array}\right.$$

Fourier Robin conditions

$$\left\{\begin{array}{lcl} \sigma^+ n^+ 2\frac{K^s}{h} u^+ & = & \sigma^- n^- + 2\frac{K^s}{h} u^+ \\ \sigma^+ n^+ & = & -\sigma^- n^- \end{array}\right.$$

Reinforcement sheet

(A.L Bessoud, F. Krasucki, M. Serpilli, 2008) (M. Serpilli, MV)

Add a **membrane** energy

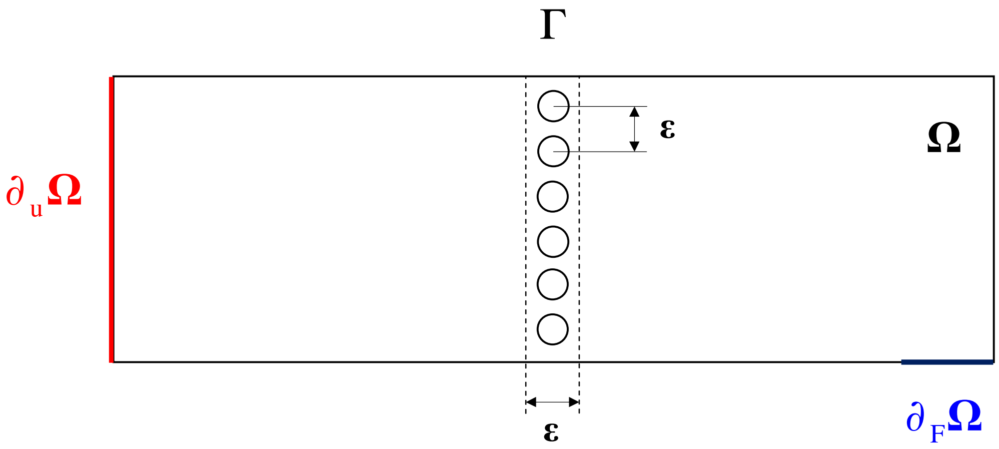
$$A_m(\bar{u}, \bar{v}) + A_{3D}(\bar{U}, \bar{V}) = F(V)$$

Interface problem

$$\mathcal{A}(\lambda) = S_1(\lambda) + S_2(\lambda) + A_m(\lambda) = 0$$

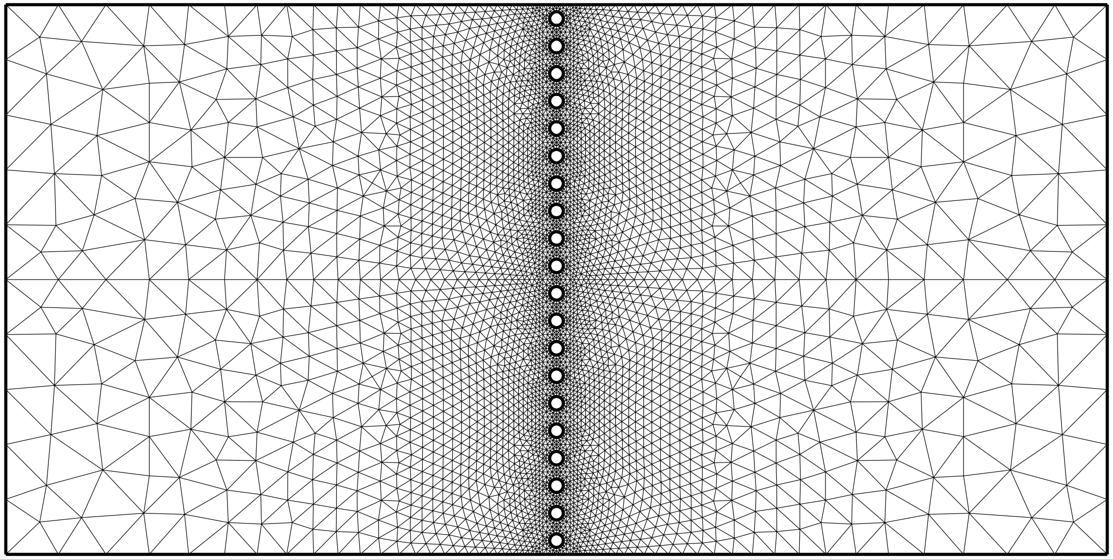
Heterogeneities: a model problem

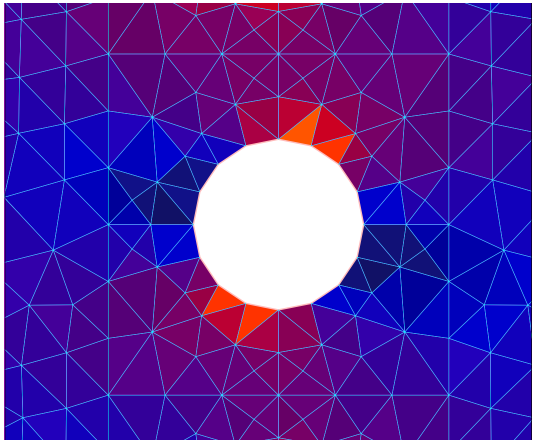
(G.Geymonat, S. Hendili, F. Krasucki, et M. V, 2012)



$$\begin{cases} -\boldsymbol{div}\boldsymbol{\sigma}^\varepsilon = \boldsymbol{g} & \text{in } \Omega \\ \boldsymbol{\sigma}^\varepsilon = \boldsymbol{A}\gamma(\boldsymbol{u}^\varepsilon) & \text{in } \Omega\backslash\mathcal{I}^\varepsilon \\ \boldsymbol{\sigma}^\varepsilon = \boldsymbol{A}^I\gamma(\boldsymbol{u}^\varepsilon) & \text{in } \mathcal{I}^\varepsilon \\ \boldsymbol{\sigma}^\varepsilon\boldsymbol{n} = \boldsymbol{F} & \text{on } \partial\Omega_F \\ \boldsymbol{u}^\varepsilon = \boldsymbol{0} & \text{on } \partial\Omega_u \end{cases}$$

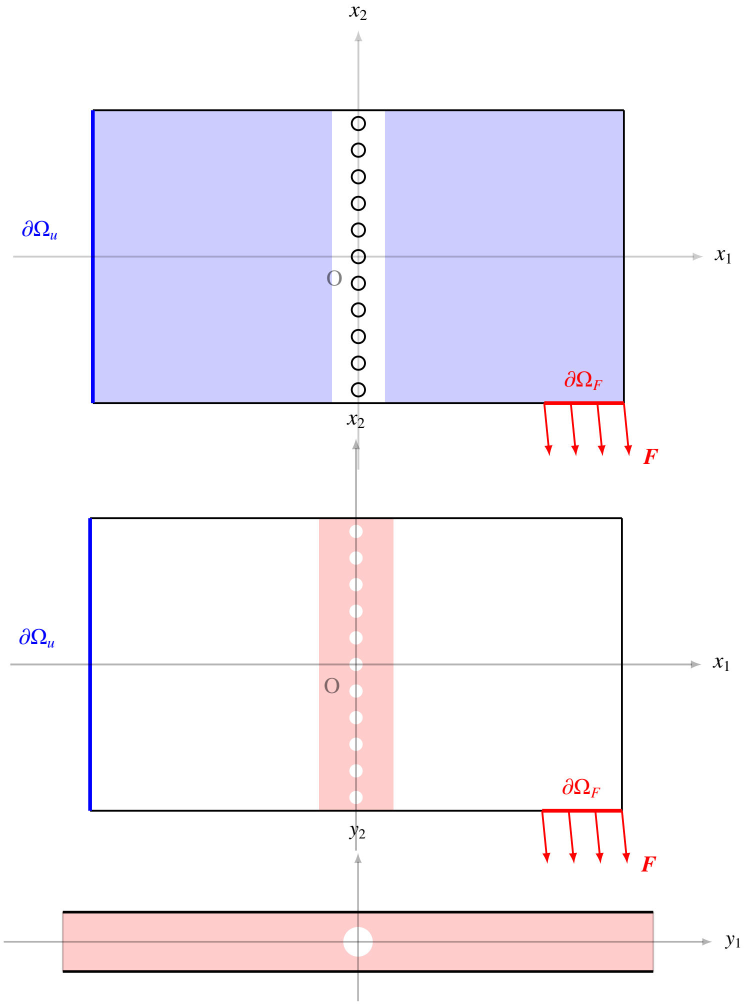
Difficulties to solve the problem





- Large number of heterogeneities
- Computational cost **increase** with the number of heterogeneities
- Difficult to obtain a correct mesh

A two scale problem



Outer expansion

$$\boldsymbol{u}^{\epsilon}(x_1,x_2) = \sum_{i=0}^{\infty} \epsilon^i \boldsymbol{u}^i(x_1,x_2)$$

$$\boldsymbol{\sigma}^{\epsilon}(x_1,x_2) = \sum_{i=0}^{\infty} \epsilon^i \boldsymbol{\sigma}^i(x_1,x_2)$$

Inner expansion

$$\boldsymbol{u}^{\epsilon}(x_1,x_2) = \sum_{i=0}^{\infty} \epsilon^i \boldsymbol{v}^i(x_2,y_1,y_2)$$

$$\boldsymbol{\sigma}^{\epsilon}(x_1,x_2) = \sum_{i=0}^{\infty} \epsilon^i \boldsymbol{\tau}^i(x_2,y_1,y_2)$$

with $y_1 = \dfrac{x_1}{\epsilon}$, $y_2 = \dfrac{x_2}{\epsilon}$

Matched asymptotic method : Results

Zero order approximation

- outer approximation

$$\left\{\begin{array}{llll} -\boldsymbol{div}\boldsymbol{\sigma}^0 & = & 0 & \text{in } \Omega\backslash\Gamma \\ \boldsymbol{\sigma}^0 & = & \boldsymbol{A}\gamma(\boldsymbol{u}^0) & \text{in } \Omega\backslash\Gamma \\ \boldsymbol{\sigma}^0\boldsymbol{n} & = & F & \text{on } \partial_F\Omega \\ \boldsymbol{u}^0 & = & \boldsymbol{u}^d & \text{on } \partial_u\Omega \\ \left[\boldsymbol{\sigma}^0\right]\boldsymbol{e}_1 & = & \left[\boldsymbol{u}^0\right] = 0 & \text{on } \Gamma \end{array}\right.$$

- inner approximation

independent of y

$$\boldsymbol{v}^0\left(\boldsymbol{x},\boldsymbol{y}\right) = \boldsymbol{v}^0\left(\boldsymbol{x}\right) = \boldsymbol{u}^0\left(0,\hat{\boldsymbol{x}}\right)$$

Remark : Zero order problem is **independent** of the heterogeneities

Matched asymptotic method : Results(cont)

First order approximation

- outer approximation :

$$\left\{\begin{array}{lcll} -\boldsymbol{div}\boldsymbol{\sigma}^{\mathbf{1}} & = & \mathbf{0} & \text{in } \Omega\backslash\Gamma \\ \boldsymbol{\sigma}^{\mathbf{1}} & = & \boldsymbol{A}\gamma(\boldsymbol{u}^1) & \text{in } \Omega\backslash\Gamma \\ \boldsymbol{\sigma}^{\mathbf{1}}\boldsymbol{n} & = & \mathbf{0} & \text{on } \partial\Omega_F \\ \boldsymbol{u}^1 & = & \mathbf{0} & \text{on } \partial\Omega_u \\ \left[\boldsymbol{u}^1\right](\hat{\boldsymbol{x}}) & = & \mathcal{G}_d\left(\boldsymbol{u}^0(0,\hat{\boldsymbol{x}})\ ;\ \left[\boldsymbol{V}^{ij}\right]^{\infty}\right) & \\ \left[\boldsymbol{\sigma}^1\boldsymbol{e}_1\right](\hat{\boldsymbol{x}}) & = & \mathcal{G}_{nS}\left(\boldsymbol{u}^0(0,\hat{\boldsymbol{x}})\ ;\ \int_{Y^\star}\boldsymbol{T}^{ij}(\boldsymbol{y})\,d\boldsymbol{y}\right) & \end{array}\right.$$

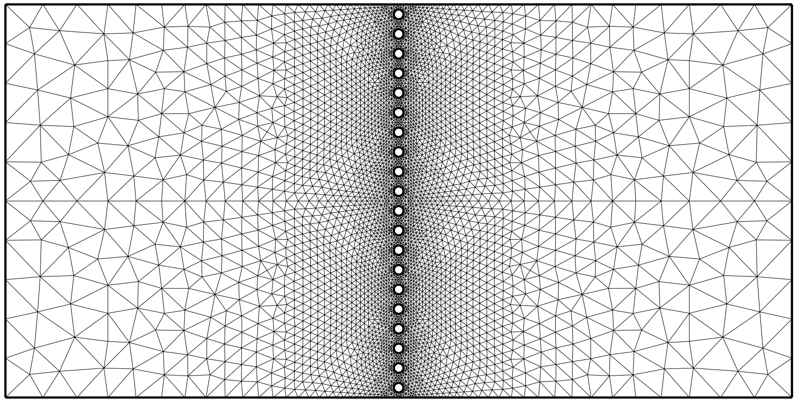
- $\mathcal{G}_d, \mathcal{G}_{nS}$ depend on $\boldsymbol{u}^0$ and its first derivative

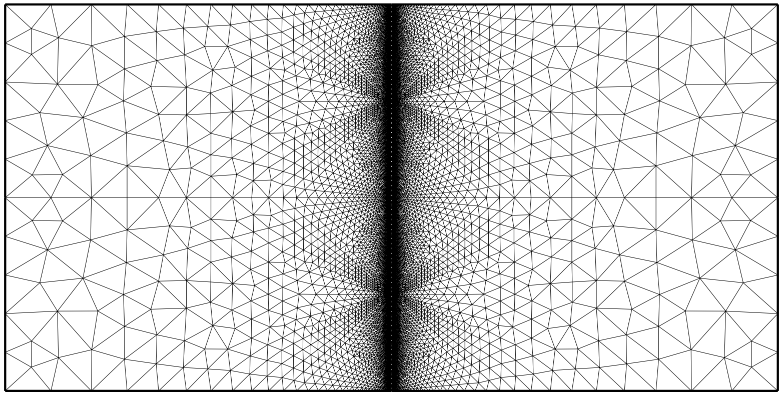
☑ This is a **non standard** problem which will be solved by a ***domain decomposition type*** algorithm

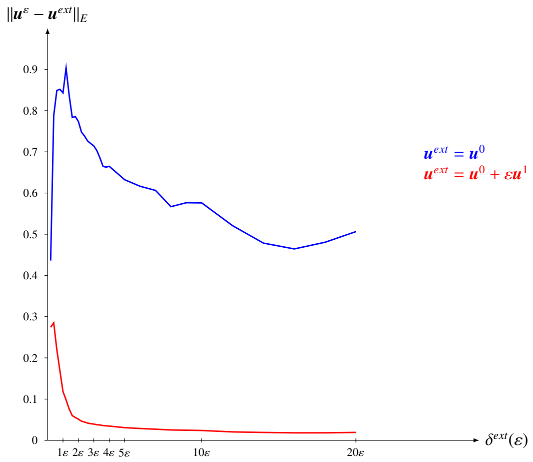
Results at a glance

$\epsilon = \frac{1}{20}$ 9836 P_2 *elements* 20125 *nodes*

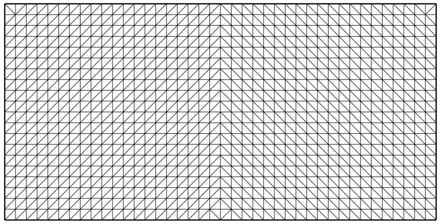
$\epsilon = \frac{1}{80}$ 41332 P_2 *elements* 85313 *nodes*



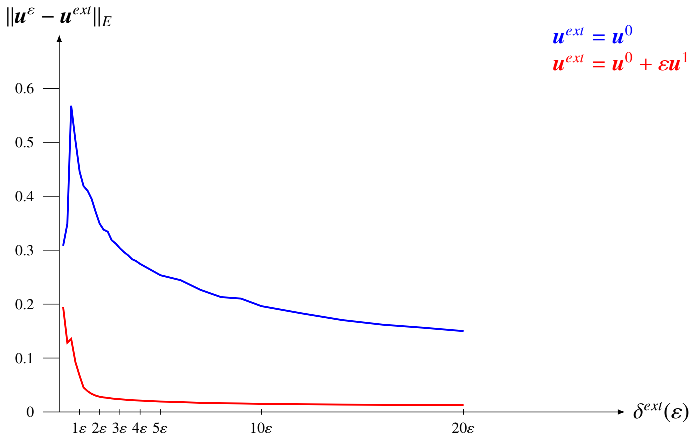




$\|u^\varepsilon - u^{ext}\|_E$
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
$u^{ext} = u^0$
$u^{ext} = u^0 + \varepsilon u^1$
1ε 2ε 3ε 4ε 5ε
10ε
20ε
$\delta^{ext}(\varepsilon)$



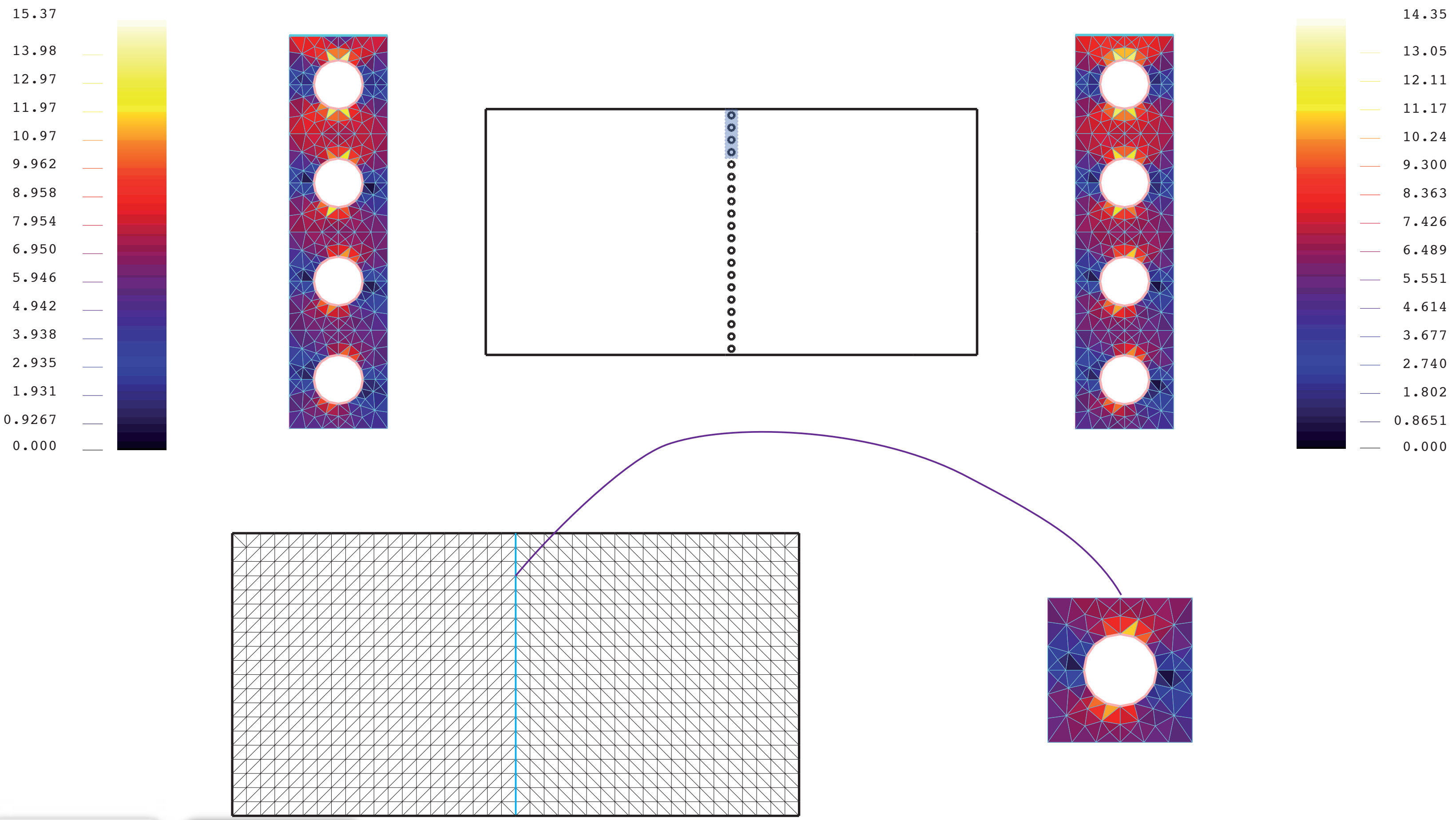
1600 elements P_2 3321 nodes



Inner problem : comparison of stresses

Solution de référence

Solution du modèle asymptotique



Heterogeneous Domain Decomposition

FLUID STRUCTURE INTERACTION

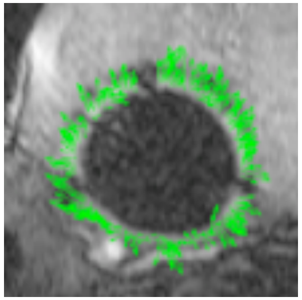
Collaboration with: Miguel Fernández, Mikel Landajuela

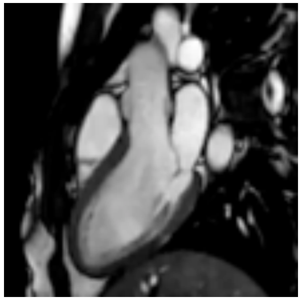
Incompressible Fluid-Structure Interaction

- Framework: coupling of
  - Fluid: incompressible (viscous,...)
  - Structure: elastic (non-linear,...)
- Widespread multi-physic problem:
  - Aeroelasticity (bridge, parachute, etc.), naval hydrodynamics,...
  - Mechanics of bio-fluid flow: blood, cerebrospinal fluid, air,...

Incompressible Fluid-Structure Interaction

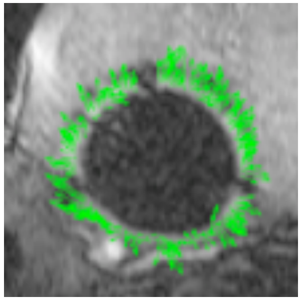
- Framework: coupling of
  - Fluid: incompressible (viscous,...)
  - Structure: elastic (non-linear,...)
- Widespread multi-physic problem:
  - Aeroelasticity (bridge, parachute, etc.), naval hydrodynamics,...
  - Mechanics of bio-fluid flow: blood, cerebrospinal fluid, air,...
- Targeted application: arterial and ventricular blood flow simulation

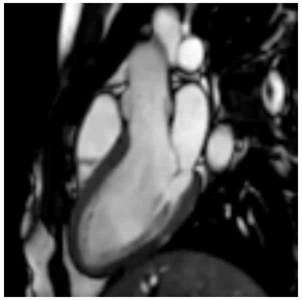




Incompressible Fluid-Structure Interaction

- Framework: coupling of
  - Fluid: incompressible (viscous,...)
  - Structure: elastic (non-linear,...)
- Widespread multi-physic problem:
  - Aeroelasticity (bridge, parachute, etc.), naval hydrodynamics,...
  - Mechanics of bio-fluid flow: blood, cerebrospinal fluid, air,...
- Targeted application: arterial and ventricular blood flow simulation





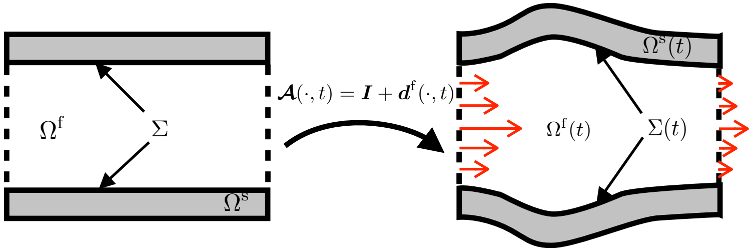
Motivations:

- Improve diagnosis (via data assimilation), therapy planing, medical devices
- Major issues in modeling, scientific computing and numerical analysis

Standard 3D Model of Blood Flow in Arteries

- Structure: non-linear elastodynamics

$$\begin{cases} \rho^{\mathrm{s}}\partial_t \dot{\boldsymbol{d}} - \mathbf{div}\left(\boldsymbol{\Pi}(\boldsymbol{d})\right) = \boldsymbol{0} & \text{in} \quad \Omega^{\mathrm{s}} \\ \dot{\boldsymbol{d}} = \partial_t \boldsymbol{d} & \text{in} \quad \Omega^{\mathrm{s}} \end{cases}$$



reference configuration

current configuration

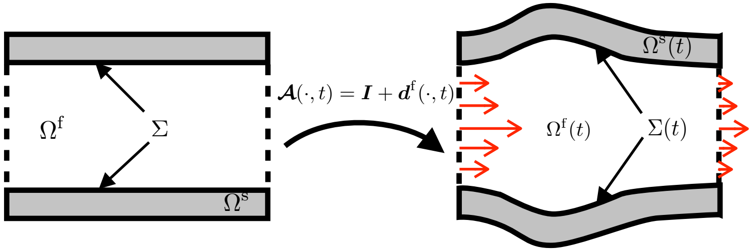
- Fluid: Navier-Stokes *(ALE formalism)*

$$\begin{cases} \rho^{\mathrm{f}}\partial_t \boldsymbol{u}|_{\boldsymbol{\mathcal{A}}} + \rho^{\mathrm{f}}(\boldsymbol{u}-\boldsymbol{w})\cdot\boldsymbol{\nabla}\boldsymbol{u} - \mathbf{div}\,\boldsymbol{\sigma}(\boldsymbol{u},p) = \boldsymbol{0} & \text{in} \quad \Omega^{\mathrm{f}}(t) \\ \operatorname{div}\boldsymbol{u} = 0 & \text{in} \quad \Omega^{\mathrm{f}}(t) \end{cases}$$

Standard 3D Model of Blood Flow in Arteries

- Structure: non-linear elastodynamics

$$\begin{cases} \rho^{\mathrm{s}} \partial_t \dot{\boldsymbol{d}} - \mathbf{div}\left(\boldsymbol{\Pi}(\boldsymbol{d})\right) = \boldsymbol{0} & \text{in} \quad \Omega^{\mathrm{s}} \\ \dot{\boldsymbol{d}} = \partial_t \boldsymbol{d} & \text{in} \quad \Omega^{\mathrm{s}} \end{cases}$$



reference configuration

current configuration

- Fluid: Navier-Stokes *(ALE formalism)*

$$\begin{cases} \rho^{\mathrm{f}} \partial_t \boldsymbol{u}|_{\boldsymbol{\mathcal{A}}} + \rho^{\mathrm{f}} (\boldsymbol{u} - \boldsymbol{w}) \cdot \boldsymbol{\nabla} \boldsymbol{u} - \mathbf{div}\, \boldsymbol{\sigma}(\boldsymbol{u}, p) = \boldsymbol{0} & \text{in} \quad \Omega^{\mathrm{f}}(t) \\ \operatorname{div} \boldsymbol{u} = 0 & \text{in} \quad \Omega^{\mathrm{f}}(t) \end{cases}$$

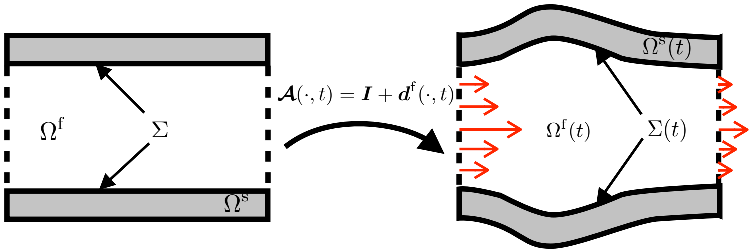
- Coupling conditions:

$$\begin{cases} \boldsymbol{d}^{\mathrm{f}} = \operatorname{Ext}\left(\boldsymbol{d}|_{\Sigma}\right), \quad \boldsymbol{w} = \partial_t \boldsymbol{d}^{\mathrm{f}} & \text{in} \quad \Omega^{\mathrm{f}} & \longleftarrow \text{geometric compatibility} \\ \boldsymbol{u} = \partial_t \boldsymbol{d} & \text{on} \quad \Sigma & \longleftarrow \text{kinematic continuity} \\ \boldsymbol{\Pi}(\boldsymbol{d}) \boldsymbol{n}^{\mathrm{s}} = -J \boldsymbol{\sigma}(\boldsymbol{u}, p) \boldsymbol{F}^{-\mathrm{T}} \boldsymbol{n} & \text{on} \quad \Sigma & \longleftarrow \text{kinetic continuity} \end{cases}$$

Standard 3D Model of Blood Flow in Arteries

- Structure: non-linear elastodynamics

$$\begin{cases} \rho^{\mathrm{s}}\partial_t\dot{\boldsymbol{d}} - \mathbf{div}\,\big(\boldsymbol{\Pi}(\boldsymbol{d})\big) = \mathbf{0} & \text{in} \quad \Omega^{\mathrm{s}} \\ \dot{\boldsymbol{d}} = \partial_t \boldsymbol{d} & \text{in} \quad \Omega^{\mathrm{s}} \end{cases}$$



reference configuration

current configuration

- Fluid: Navier-Stokes *(ALE formalism)*

$$\begin{cases} \rho^{\mathrm{f}}\partial_t\boldsymbol{u}|_{\boldsymbol{\mathcal{A}}} + \rho^{\mathrm{f}}(\boldsymbol{u}-\boldsymbol{w})\cdot\boldsymbol{\nabla}\boldsymbol{u} - \mathbf{div}\,\boldsymbol{\sigma}(\boldsymbol{u},p) = \mathbf{0} & \text{in} \quad \Omega^{\mathrm{f}}(t) \\ \operatorname{div}\boldsymbol{u} = 0 & \text{in} \quad \Omega^{\mathrm{f}}(t) \end{cases}$$

- Coupling conditions:

$$\begin{cases} \boldsymbol{d}^{\mathrm{f}} = \operatorname{Ext}\,(\boldsymbol{d}|_{\Sigma}), \quad \boldsymbol{w} = \partial_t\boldsymbol{d}^{\mathrm{f}} & \text{in} \quad \Omega^{\mathrm{f}} \\ \boldsymbol{u} = \partial_t\boldsymbol{d} & \text{on} \quad \Sigma \\ \boldsymbol{\Pi}(\boldsymbol{d})\boldsymbol{n}^{\mathrm{s}} = -J\boldsymbol{\sigma}(\boldsymbol{u},p)\boldsymbol{F}^{-\mathrm{T}}\boldsymbol{n} & \text{on} \quad \Sigma \end{cases}$$

⟵ geometric compatibility

⟵ kinematic continuity

⟵ kinetic continuity

Major issue:

Computational complexity: efficient partitioning extremely difficult

Why FSI in a DD framework?

Why FSI in a DD framework?

Revisit vocabulary

Coupling schemes

- **Explicit (weak)** coupling vs **Implicit (strong)** and **semi-implicit** coupling

Wikipedia

Two main approaches exist for the simulation of fluid–structure interaction problems:

- **Monolithic** approach: the equations governing the flow and the displacement of the structure are solved simultaneously, with a single solver
- **Partitioned** approach: the equations governing the flow and the displacement of the structure are solved separately, with two distinct solvers

Why FSI in a DD framework?

Revisit vocabulary

Coupling schemes

- **Explicit (weak)** coupling vs **Implicit (strong)** and **semi-implicit** coupling

Wikipedia

Two main approaches exist for the simulation of fluid–structure interaction problems:

- **Monolithic** approach: the equations governing the flow and the displacement of the structure are solved simultaneously, with a single solver
- **Partitioned** approach: the equations governing the flow and the displacement of the structure are solved separately, with two distinct solvers

Domain decomposition framework

- Formulation : monolithic
- Solution algorithm : DD (partitioned...)
- Weak coupling : one iteration of a given DD algorithm (with an appropriate initialization)
- Strong coupling : iterate till convergence

FSI = Heterogenous domain decomposition
(Not a new idea!)

DD5 (1991) *Quarteroni, Pasquarelli,Valli*

In introduction FSI is mentioned,

Focus on problems homogenous in nature can be faced in a heterogeneous fashion after reducing the given problem to a simplified one in a subregion ex in fluid dynamics NS and Boltzmann Kinetic models

(1999) *Le Tallec, Mouro* *Fluid structure interaction with large structural displacements*

(2001---> today) a lot of people!

Design efficient and reliable parallel methods

✱ Methodology :

- Take advantage of modularity and use robust well validated components
- Numerical methods:

  □ Fluid : ALE Navier-Stokes
  □ Structure : Non-linear elasto-dynamic (shells)
  ☑ FSI : Explicit coupling

- Design algorithms **well suited for parallel computing** : **use domain decomposition**

  □ Additive Schwarz for the fluid (PETSCI)
  ☑ Balanced domain decomposition method for the solid solver

✱ Implementation

- ● Use different solvers for fluid and solid validation platform
- ● Use the **most appropriate** parallelization technique (PVM, MPI....)

✱ Performance

☑ **Robustness and efficiency** of the algorithms, ***numerical scalability***

□ Optimize the decomposition, level of parallelism....

Monolithic or partitioned approach?

Monolithic or partitioned approach?

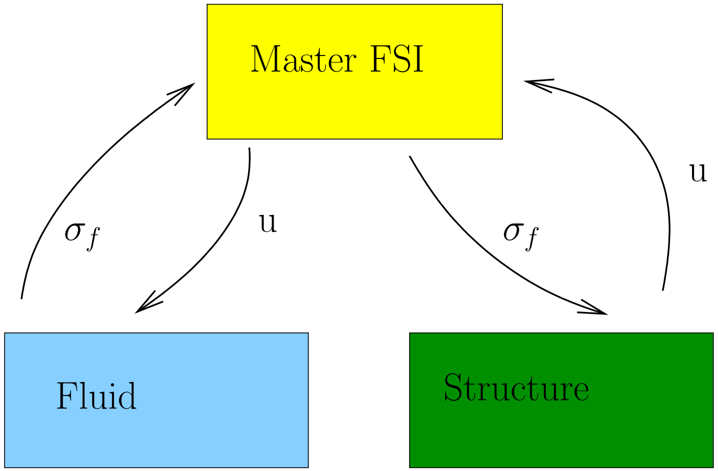
Domain decomposition technique

- **Monolithic** approach (formulation view point)
- **Partitioned** approach (implementation view point)
- Possible to use *state of the art* **different solvers**

Monolithic or partitioned approach?

Domain decomposition technique

- **Monolithic** approach (formulation view point)
- **Partitioned** approach (implementation view point)
- Possible to use *state of the art* **different solvers**



Monolithic or partitioned approach?

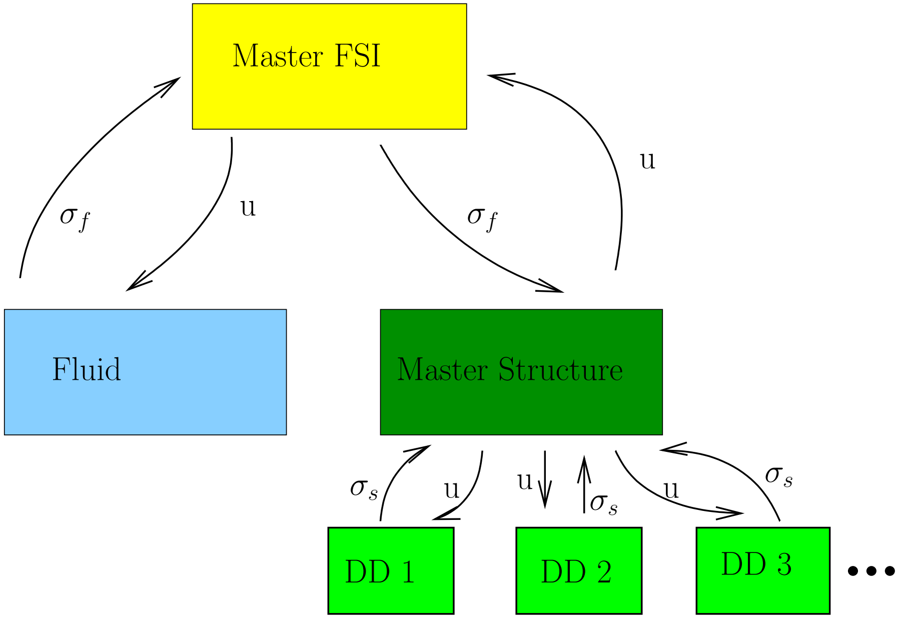
Domain decomposition technique

- **Monolithic** approach (formulation view point)
- **Partitioned** approach (implementation view point)
- Possible to use *state of the art* **different solvers**

Monolithic or partitioned approach?

Domain decomposition technique

- **Monolithic** approach (formulation view point)
- **Partitioned** approach (implementation view point)
- Possible to use *state of the art* **different solvers**



Monolithic or partitioned approach?

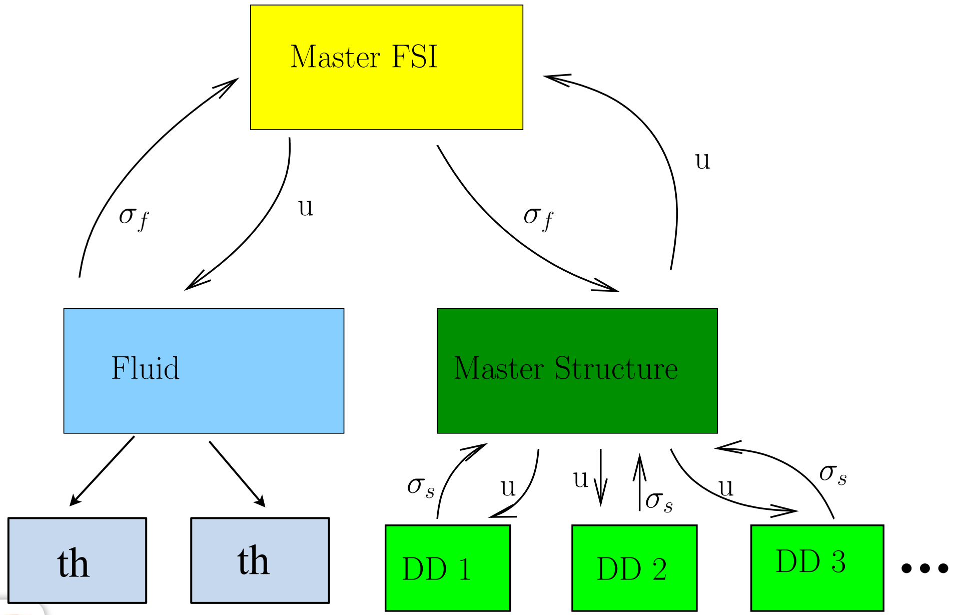
Domain decomposition technique

- **Monolithic** approach (formulation view point)
- **Partitioned** approach (implementation view point)
- Possible to use *state of the art* **different solvers**

Monolithic or partitioned approach?

Domain decomposition technique

- **Monolithic** approach (formulation view point)
- **Partitioned** approach (implementation view point)
- Possible to use *state of the art* **different solvers**



Difficulties

Scientific limitations:

Software limitations:

Difficulties

Scientific limitations:

- **pertinence** of the models used
- **efficiency** and **reliability** of the numerical methods
- how to validate ?

Software limitations:

Difficulties

Scientific limitations:

- **pertinence** of the models used
- **efficiency** and **reliability** of the numerical methods
- how to validate ?

Software limitations:

- **large** number of software components: mesh generation, fluid solver, solid solver, coupling algorithm
- limitations due to the state of the art

Difficulties

Scientific limitations:

- **pertinence** of the models used
- **efficiency** and **reliability** of the numerical methods
- how to validate ?

Software limitations:

- **large** number of software components: mesh generation, fluid solver, solid solver, coupling algorithm
- limitations due to the state of the art

Old example: the reliable elements in the discretization of the solid problems are quadrangles, automatic mesh generator produces triangles

Difficulties

Scientific limitations:

- **pertinence** of the models used ← easy to change with partitioned approach
- **efficiency** and **reliability** of the numerical methods
- how to validate ?

Software limitations:

- **large** number of software components: mesh generation, fluid solver, solid solver, coupling algorithm
- limitations due to the state of the art

Old example: the reliable elements in the discretization of the solid problems are quadrangles, automatic mesh generator produces triangles

Difficulties

Scientific limitations:

- **pertinence** of the models used ← easy to change with partitioned approach
- **efficiency** and **reliability** of the numerical methods ← explicit stable and accurate alg
- how to validate ?

Software limitations:

- **large** number of software components: mesh generation, fluid solver, solid solver, coupling algorithm
- limitations due to the state of the art

Old example: the reliable elements in the discretization of the solid problems are quadrangles, automatic mesh generator produces triangles

Difficulties

Scientific limitations:

- **pertinence** of the models used ← easy to change with partitioned approach
- **efficiency** and **reliability** of the numerical methods ← explicit stable and accurate alg
- how to validate ? ← use benchmarks from literature

Software limitations:

- **large** number of software components: mesh generation, fluid solver, solid solver, coupling algorithm
- limitations due to the state of the art

Old example: the reliable elements in the discretization of the solid problems are quadrangles, automatic mesh generator produces triangles

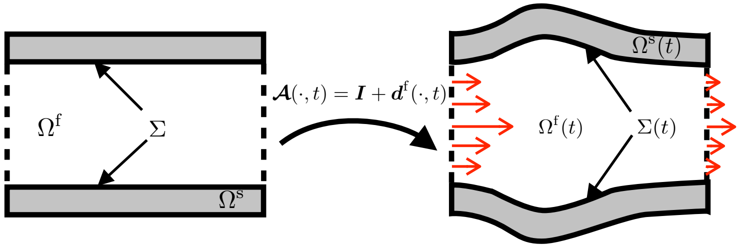
Standard 3D Model of Blood Flow in Arteries

- Structure: non-linear elastodynamics

$$\begin{cases} \begin{pmatrix}\rho^{\mathrm{s}}\epsilon\partial_t\dot{\boldsymbol{d}}\\ 0\end{pmatrix} + \begin{pmatrix}\boldsymbol{L}^{\mathrm{e}}_{\boldsymbol{d}}((\boldsymbol{d},\boldsymbol{\theta}))\\ L^{\mathrm{e}}_{\boldsymbol{\theta}}((\boldsymbol{d},\boldsymbol{\theta}))\end{pmatrix} = \begin{pmatrix}-J\boldsymbol{\sigma}(\boldsymbol{u},p)\boldsymbol{F}^{-\mathrm{T}}\boldsymbol{n}\\ \boldsymbol{0}\end{pmatrix} & \text{on} \quad \Sigma \\ \dot{\boldsymbol{d}} = \partial_t \boldsymbol{d} & \text{on} \quad \Sigma \end{cases}$$

- Fluid: Navier-Stokes *(ALE formalism)*

$$\begin{cases} \rho^{\mathrm{f}}\partial_t\boldsymbol{u}|_{\boldsymbol{\mathcal{A}}} + \rho^{\mathrm{f}}(\boldsymbol{u}-\boldsymbol{w})\cdot\boldsymbol{\nabla}\boldsymbol{u} - \mathbf{div}\,\boldsymbol{\sigma}(\boldsymbol{u},p) = \boldsymbol{0} & \text{in} \quad \Omega^{\mathrm{f}}(t) \\ \operatorname{div}\boldsymbol{u} = 0 & \text{in} \quad \Omega^{\mathrm{f}}(t) \\ \boldsymbol{d}^{\mathrm{f}} = \operatorname{Ext}(\boldsymbol{d}|_{\Sigma}), \quad \boldsymbol{w} = \partial_t\boldsymbol{d}^{\mathrm{f}} & \text{in} \quad \Omega^{\mathrm{f}} \\ \boldsymbol{u} = \partial_t\boldsymbol{d} & \text{on} \quad \Sigma \end{cases}$$



reference configuration

current configuration

Standard 3D Model of Blood Flow in Arteries

- Structure: non-linear elastodynamics

$$\begin{cases} \begin{pmatrix}\rho^{\mathrm{s}} \epsilon \partial_t \dot{\boldsymbol{d}} \\ 0\end{pmatrix} + \begin{pmatrix} \boldsymbol{L}^{\mathrm{e}}_{\boldsymbol{d}}((\boldsymbol{d},\boldsymbol{\theta})) \\ L^{\mathrm{e}}_{\boldsymbol{\theta}}((\boldsymbol{d},\boldsymbol{\theta}))\end{pmatrix} = \begin{pmatrix} -J\boldsymbol{\sigma}(\boldsymbol{u},p)\boldsymbol{F}^{-\mathrm{T}}\boldsymbol{n} \\ \boldsymbol{0}\end{pmatrix} & \text{on} \quad \Sigma \\ \dot{\boldsymbol{d}} = \partial_t \boldsymbol{d} & \text{on} \quad \Sigma \end{cases}$$

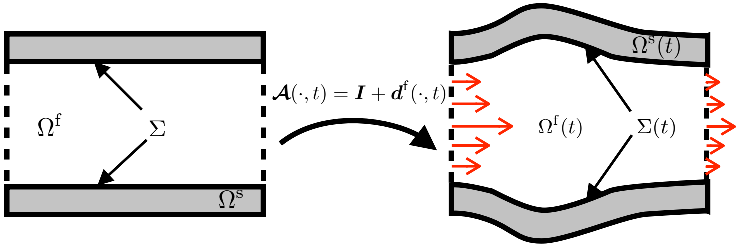
← kinetic continuity

- Fluid: Navier-Stokes *(ALE formalism)*

$$\begin{cases} \rho^{\mathrm{f}} \partial_t \boldsymbol{u}|_{\boldsymbol{\mathcal{A}}} + \rho^{\mathrm{f}} (\boldsymbol{u} - \boldsymbol{w}) \cdot \boldsymbol{\nabla} \boldsymbol{u} - \mathbf{div}\, \boldsymbol{\sigma}(\boldsymbol{u},p) = \boldsymbol{0} & \text{in} \quad \Omega^{\mathrm{f}}(t) \\ \operatorname{div} \boldsymbol{u} = 0 & \text{in} \quad \Omega^{\mathrm{f}}(t) \\ \boldsymbol{d}^{\mathrm{f}} = \operatorname{Ext}(\boldsymbol{d}|_{\Sigma}), \quad \boldsymbol{w} = \partial_t \boldsymbol{d}^{\mathrm{f}} & \text{in} \quad \Omega^{\mathrm{f}} \\ \boldsymbol{u} = \partial_t \boldsymbol{d} & \text{on} \quad \Sigma \end{cases}$$

← geometric compatibility

← kinematic continuity



reference configuration

current configuration

Standard 3D Model of Blood Flow in Arteries

- Structure: non-linear elastodynamics

$$\left\{\begin{aligned} &\begin{pmatrix}\rho^{\mathrm{s}}\epsilon\partial_t\dot{\boldsymbol{d}}\\ 0\end{pmatrix}+\begin{pmatrix}\boldsymbol{L}^{\mathrm{e}}_{\boldsymbol{d}}((\boldsymbol{d},\boldsymbol{\theta}))\\ L^{\mathrm{e}}_{\boldsymbol{\theta}}((\boldsymbol{d},\boldsymbol{\theta}))\end{pmatrix}=\begin{pmatrix}-J\boldsymbol{\sigma}(\boldsymbol{u},p)\boldsymbol{F}^{-\mathrm{T}}\boldsymbol{n}\\ \boldsymbol{0}\end{pmatrix} \quad \text{on} \quad \Sigma\\ &\dot{\boldsymbol{d}}=\partial_t\boldsymbol{d} \quad \text{on} \quad \Sigma\end{aligned}\right.$$

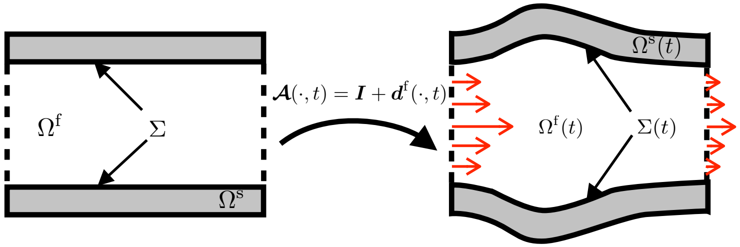
← kinetic continuity

- Fluid: Navier-Stokes *(ALE formalism)*

$$\left\{\begin{aligned} \rho^{\mathrm{f}}\partial_t\boldsymbol{u}|_{\boldsymbol{\mathcal{A}}}+\rho^{\mathrm{f}}(\boldsymbol{u}-\boldsymbol{w})\cdot\boldsymbol{\nabla}\boldsymbol{u}-\mathbf{div}\,\boldsymbol{\sigma}(\boldsymbol{u},p)=\boldsymbol{0} &\quad \text{in} \quad \Omega^{\mathrm{f}}(t)\\ \operatorname{div}\boldsymbol{u}=0 &\quad \text{in} \quad \Omega^{\mathrm{f}}(t)\\ \boldsymbol{d}^{\mathrm{f}}=\operatorname{Ext}(\boldsymbol{d}|_{\Sigma}),\quad \boldsymbol{w}=\partial_t\boldsymbol{d}^{\mathrm{f}} &\quad \text{in} \quad \Omega^{\mathrm{f}}\\ \boldsymbol{u}=\partial_t\boldsymbol{d} &\quad \text{on} \quad \Sigma\end{aligned}\right.$$

← geometric compatibility

← kinematic continuity



reference configuration

current configuration

Major issue:

Computational complexity: efficient partitioning extremely difficult

The Failure of (D-N) Explicit Coupled Schemes

- Explicit treatment of the geometric and kinematic compatibility:

$$\begin{cases} \boldsymbol{d}^{\mathrm{f},n} = \mathrm{Ext}(\boldsymbol{d}^{n-1}|_{\Sigma}), \quad \boldsymbol{w}^n = \partial_\tau \boldsymbol{d}^{\mathrm{f},n} & \text{in} \quad \Omega^{\mathrm{f}} \\ \boldsymbol{u}^n = \dot{\boldsymbol{d}}^{n-1} & \text{on} \quad \Sigma \\ \rho^{\mathrm{s}} \epsilon \partial_\tau \dot{\boldsymbol{d}}^n - \boldsymbol{L}^{\mathrm{e}}_{\boldsymbol{d}}((\boldsymbol{d}^n, \boldsymbol{\theta})) = -J^n \boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)(\boldsymbol{F}^n)^{-\mathrm{T}} \boldsymbol{n} & \text{on} \quad \Sigma \end{cases}$$

Notation: backward difference

$$\partial_\tau x^n \overset{\mathrm{def}}{=} \frac{x^n - x^{n-1}}{\tau}$$

The Failure of (D-N) Explicit Coupled Schemes

- Explicit treatment of the geometric and kinematic compatibility:

$$\left\{\begin{aligned} \boldsymbol{d}^{\mathrm{f},n} = \mathrm{Ext}(\boldsymbol{d}^{n-1}|_{\Sigma}), \quad \boldsymbol{w}^n = \partial_\tau \boldsymbol{d}^{\mathrm{f},n} & \quad \text{in} \quad \Omega^{\mathrm{f}} \\ \boldsymbol{u}^n = \dot{\boldsymbol{d}}^{n-1} & \quad \text{on} \quad \Sigma \\ \rho^{\mathrm{s}} \epsilon \partial_\tau \dot{\boldsymbol{d}}^n - \boldsymbol{L}^{\mathrm{e}}_{\boldsymbol{d}}((\boldsymbol{d}^n, \boldsymbol{\theta})) = -J^n \boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)(\boldsymbol{F}^n)^{-\mathrm{T}} \boldsymbol{n} & \quad \text{on} \quad \Sigma \end{aligned}\right.$$

- Low computational cost: uncoupled fluid-solid time-marching

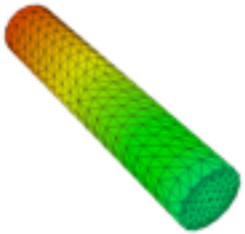
The Failure of (D-N) Explicit Coupled Schemes

- Explicit treatment of the geometric and kinematic compatibility:

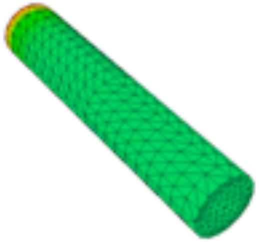
$$\left\{\begin{array}{rll} \boldsymbol{d}^{\mathrm{f},n} = \mathrm{Ext}(\boldsymbol{d}^{n-1}|_{\Sigma}), \quad \boldsymbol{w}^{n} = \partial_\tau \boldsymbol{d}^{\mathrm{f},n} & \text{in} & \Omega^{\mathrm{f}} \\ \boldsymbol{u}^{n} = \dot{\boldsymbol{d}}^{n-1} & \text{on} & \Sigma \\ \rho^{\mathrm{s}} \epsilon \partial_\tau \dot{\boldsymbol{d}}^{n} - \boldsymbol{L}^{\mathrm{e}}_{\boldsymbol{d}}((\boldsymbol{d}^{n}, \boldsymbol{\theta})) = -J^{n} \boldsymbol{\sigma}(\boldsymbol{u}^{n}, p^{n})(\boldsymbol{F}^{n})^{-\mathrm{T}} \boldsymbol{n} & \text{on} & \Sigma \end{array}\right.$$

- Low computational cost: uncoupled fluid-solid time-marching

- Unconditionally unstable:



D-N loosely coupled scheme



reference solution

The Failure of (D-N) Explicit Coupled Schemes

- Explicit treatment of the geometric and kinematic compatibility:

$$\begin{cases} \boldsymbol{d}^{\mathrm{f},n} = \mathrm{Ext}(\boldsymbol{d}^{n-1}|_{\Sigma}), \quad \boldsymbol{w}^n = \partial_\tau \boldsymbol{d}^{\mathrm{f},n} & \text{in} \quad \Omega^{\mathrm{f}} \\ \boldsymbol{u}^n = \dot{\boldsymbol{d}}^{n-1} & \text{on} \quad \Sigma \\ \rho^{\mathrm{s}} \epsilon \partial_\tau \dot{\boldsymbol{d}}^n - \boldsymbol{L}^{\mathrm{e}}_{\boldsymbol{d}}((\boldsymbol{d}^n, \boldsymbol{\theta})) = -J^n \boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)(\boldsymbol{F}^n)^{-\mathrm{T}} \boldsymbol{n} & \text{on} \quad \Sigma \end{cases}$$

- Low computational cost: uncoupled fluid-solid time-marching

- Unconditionally unstable:

D-N loosely coupled scheme

reference solution

- Added-mass INStability condition (simplified model): $\dfrac{\rho^{\mathrm{s}} \epsilon}{\rho^{\mathrm{f}} \lambda_{\mathrm{add}}} < 1$

  (Causin, Gerbeau, Nobile '05)

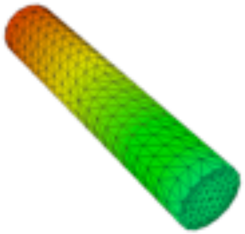
The Failure of (D-N) Explicit Coupled Schemes

- Explicit treatment of the geometric and kinematic compatibility:

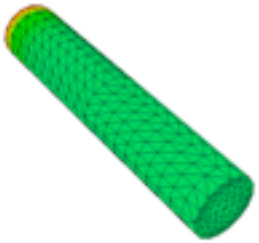
$$\begin{cases} \boldsymbol{d}^{\mathrm{f},n} = \mathrm{Ext}(\boldsymbol{d}^{n-1}|_{\Sigma}), \quad \boldsymbol{w}^{n} = \partial_\tau \boldsymbol{d}^{\mathrm{f},n} & \text{in} \quad \Omega^{\mathrm{f}} \\ \boldsymbol{u}^{n} = \dot{\boldsymbol{d}}^{n-1} & \text{on} \quad \Sigma \\ \rho^{\mathrm{s}} \epsilon \partial_\tau \dot{\boldsymbol{d}}^{n} - \boldsymbol{L}^{\mathrm{e}}_{\boldsymbol{d}}((\boldsymbol{d}^{n}, \boldsymbol{\theta})) = -J^{n} \boldsymbol{\sigma}(\boldsymbol{u}^{n}, p^{n})(\boldsymbol{F}^{n})^{-\mathrm{T}} \boldsymbol{n} & \text{on} \quad \Sigma \end{cases}$$

- Low computational cost: uncoupled fluid-solid time-marching

- Unconditionally unstable:



D-N loosely coupled scheme



reference solution

- Added-mass INStability condition (simplified model): $\dfrac{\rho^{\mathrm{s}} \epsilon}{\rho^{\mathrm{f}} \lambda_{\mathrm{add}}} < 1$ ⬅ **satisfied by blood flows**

  (Causin, Gerbeau, Nobile '05)

Standard Cures, Trends and Alternatives

- Implicit the coupling schemes:
  - Unconditionally energy stable, but computationally demanding
  - Vast literature (partitioned, monolithic,...)

(Mok et al. ’01,Heil ’04, Fernández, Moubachir ’05, Dettmer, Peric ’06, Badia et al.’08, Gee et al. ’11,...)

Domain decomposition approach

Option 1 : decompose first then linearize

- Dirichlet-Neumann
  - Fixed-point *Le Tallec-Mouro ’99,Wall-Ramm ’01....*
  - Newton *Fernàndez-Moubachir ’03....*
  - Inexact Newton *Matthies-Steindorf ’03, Gerbeau,MV. ’03,Mischler-van Brummelen-de- Borst ’05*
- Neumann-Neumann *Deparis-Discacciati-Quarteroni ’05*
- Robin-Neumann *Badia-Nobile-Vergara ’07*

Option 2 : linearize first then decompose *Fernàndez-Gerbeau-Cloria,MV*

- Dirichlet-Neumann
- Neumann-Neumann
  - Does not work $M = \frac{1}{2}S_f^{-1} + \frac{1}{2}S_s^{-1}$
  - Seams to work $M = \alpha_1 S_f^{-1} + \alpha_2 S_s^{-1}$ but $\alpha_1 \approx 0$

Standard Cures, Trends and Alternatives (cont)

Standard Cures, Trends and Alternatives (cont)

- Semi-implicit coupling schemes:
  - Conditionally energy stable, but fractional-step scheme required and not fully explicit
    (Fernández, Gerbeau, Grandmont '07, Quarteroni, Quaini '08,...)

Standard Cures, Trends and Alternatives (cont)

- Semi-implicit coupling schemes:
  - Conditionally energy stable, but fractional-step scheme required and not fully explicit

    (Fernández, Gerbeau, Grandmont '07, Quarteroni, Quaini '08,...)

- Stable loose couple alternatives (added-mass free):
  - Nitsche's based *stabilized explicit coupling*

    (Burman, Fernández'07, '09)

  - For thin-structures, *kinematically coupled scheme*

    (Guidoboni, Glowinski, Cavallini, Canic '09)

Standard Cures, Trends and Alternatives (cont)

- Semi-implicit coupling schemes:
  - Conditionally energy stable, but fractional-step scheme required and not fully explicit

    (Fernández, Gerbeau, Grandmont '07, Quarteroni, Quaini '08,...)

- Stable loose couple alternatives (added-mass free):
  - Nitsche's based *stabilized explicit coupling*

    (Burman, Fernández'07, '09)
  - For thin-structures, *kinematically coupled scheme*

    (Guidoboni, Glowinski, Cavallini, Canic '09)

optimal accuracy demands restrictive CFL

$$\tau = \mathcal{O}(h^2)$$

Standard Cures, Trends and Alternatives (cont)

- Semi-implicit coupling schemes:
  - Conditionally energy stable, but fractional-step scheme required and not fully explicit

    (Fernández, Gerbeau, Grandmont '07, Quarteroni, Quaini '08,...)

- Stable loose couple alternatives (added-mass free):
  - Nitsche's based *stabilized explicit coupling*

    (Burman, Fernández'07, '09)

  - For thin-structures, *kinematically coupled scheme*

    (Guidoboni, Glowinski, Cavallini, Canic '09)

optimal accuracy demands restrictive CFL

$$\tau = \mathcal{O}(h^2)$$

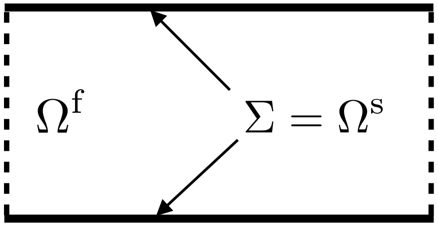
Main issue:

Stable and optimally accurate loosely coupled schemes (& mathematically sound)

Linear Model Problem with thin-structure

- Fluid: Stokes flow

$$\left\{\begin{aligned} \rho^{\mathrm{f}}\partial_t\boldsymbol{u} - \mathbf{div}\,\boldsymbol{\sigma}(\boldsymbol{u},p) &= \mathbf{0} \quad \text{in} \quad \Omega^{\mathrm{f}}\\ \mathrm{div}\,\boldsymbol{u} &= 0 \quad \text{in} \quad \Omega^{\mathrm{f}}\\ \boldsymbol{u} &= \dot{\boldsymbol{d}} \quad \text{on} \quad \Sigma \end{aligned}\right.$$



- Thin-solid: shell

$$\left\{\begin{aligned} \begin{pmatrix}\rho^{\mathrm{s}}\epsilon\partial_t\dot{\boldsymbol{d}}\\ 0\end{pmatrix} - \begin{pmatrix}\boldsymbol{L}^{\mathrm{e}}_{\boldsymbol{d}}(\boldsymbol{d},\boldsymbol{\theta})\\ L^{\mathrm{e}}_{\boldsymbol{\theta}}(\boldsymbol{d},\boldsymbol{\theta})\end{pmatrix} &= \begin{pmatrix}-\boldsymbol{\sigma}(\boldsymbol{u},p)\boldsymbol{n}\\ \mathbf{0}\end{pmatrix} \quad \text{on} \quad \Sigma\\ \dot{\boldsymbol{d}} &= \partial_t\boldsymbol{d} \quad \text{on} \quad \Sigma \end{aligned}\right.$$

Key Feature: Interface Robin Coupling

- Fluid:

$$\left\{\begin{aligned} \rho^{\mathrm{f}}\partial_t \boldsymbol{u} - \mathbf{div}\,\boldsymbol{\sigma}(\boldsymbol{u},p) = \mathbf{0} \quad & \text{in} \quad \Omega^{\mathrm{f}} \\ \operatorname{div}\boldsymbol{u} = 0 \quad & \text{in} \quad \Omega^{\mathrm{f}} \\ \boldsymbol{u} = \dot{\boldsymbol{d}} \quad & \text{on} \quad \Sigma \end{aligned}\right.$$

- Thin-solid:

$$\left\{\begin{aligned} \begin{pmatrix} \rho^{\mathrm{s}}\epsilon\partial_t\dot{\boldsymbol{d}} \\ 0 \end{pmatrix} - \begin{pmatrix} \boldsymbol{L}^{\mathrm{e}}_{\boldsymbol{d}}(\boldsymbol{d},\boldsymbol{\theta}) \\ L^{\mathrm{e}}_{\boldsymbol{\theta}}(\boldsymbol{d},\boldsymbol{\theta}) \end{pmatrix} = \begin{pmatrix} -\boldsymbol{\sigma}(\boldsymbol{u},p)\boldsymbol{n} \\ \mathbf{0} \end{pmatrix} \quad & \text{on} \quad \Sigma \\ \dot{\boldsymbol{d}} = \partial_t\boldsymbol{d} \quad & \text{on} \quad \Sigma \end{aligned}\right.$$

Key Feature: Interface Robin Coupling

- Fluid:

$$\left\{\begin{aligned} \rho^{\mathrm{f}}\partial_t \boldsymbol{u} - \mathbf{div}\,\boldsymbol{\sigma}(\boldsymbol{u},p) = \mathbf{0} \quad & \text{in} \quad \Omega^{\mathrm{f}} \\ \operatorname{div} \boldsymbol{u} = 0 \quad & \text{in} \quad \Omega^{\mathrm{f}} \\ \boldsymbol{u} = \dot{\boldsymbol{d}} \quad & \text{on} \quad \Sigma \end{aligned}\right.$$

- Thin-solid:

$$\left\{\begin{aligned} \begin{pmatrix} \rho^{\mathrm{s}}\epsilon\partial_t \dot{\boldsymbol{d}} \\ 0 \end{pmatrix} - \begin{pmatrix} \boldsymbol{L}^{\mathrm{e}}_{\boldsymbol{d}}(\boldsymbol{d},\boldsymbol{\theta}) \\ \boldsymbol{L}^{\mathrm{e}}_{\boldsymbol{\theta}}(\boldsymbol{d},\boldsymbol{\theta}) \end{pmatrix} = \begin{pmatrix} -\boldsymbol{\sigma}(\boldsymbol{u},p)\boldsymbol{n} \\ \mathbf{0} \end{pmatrix} \quad & \text{on} \quad \Sigma \\ \dot{\boldsymbol{d}} = \partial_t \boldsymbol{d} \quad & \text{on} \quad \Sigma \end{aligned}\right.$$

$$\boldsymbol{\sigma}(\boldsymbol{u},p)\boldsymbol{n} + \rho^{\mathrm{s}}\epsilon\partial_t \boldsymbol{u} = -\boldsymbol{L}^{\mathrm{e}}_{\boldsymbol{d}}(\boldsymbol{d},\boldsymbol{\theta}) \quad \text{on} \quad \Sigma$$

(Nobile, Vergara '08, Badia et al. '08, Guidoboni et al. '09,...)

Key Feature: Interface Robin Coupling

- Fluid:

$$\begin{cases} \rho^{\mathrm{f}}\partial_t \boldsymbol{u} - \mathbf{div}\,\boldsymbol{\sigma}(\boldsymbol{u},p) = \mathbf{0} & \text{in} \quad \Omega^{\mathrm{f}}\\ \operatorname{div}\boldsymbol{u} = 0 & \text{in} \quad \Omega^{\mathrm{f}}\\ \boldsymbol{u} = \dot{\boldsymbol{d}} & \text{on} \quad \Sigma \end{cases}$$

- Thin-solid:

$$\begin{cases} \begin{pmatrix}\rho^{\mathrm{s}}\epsilon\partial_t\dot{\boldsymbol d}\\ 0\end{pmatrix} - \begin{pmatrix}\boldsymbol L^{\mathrm{e}}_{\boldsymbol d}(\boldsymbol d,\boldsymbol\theta)\\ L^{\mathrm{e}}_{\boldsymbol\theta}(\boldsymbol d,\boldsymbol\theta)\end{pmatrix} = \begin{pmatrix}-\boldsymbol\sigma(\boldsymbol u,p)\boldsymbol n\\ \mathbf 0\end{pmatrix} & \text{on} \quad \Sigma\\ \dot{\boldsymbol d} = \partial_t \boldsymbol d & \text{on} \quad \Sigma\end{cases}$$

$$\boldsymbol\sigma(\boldsymbol u,p)\boldsymbol n + \rho^{\mathrm{s}}\epsilon\partial_t\boldsymbol u = -\boldsymbol L^{\mathrm{e}}_{\boldsymbol d}(\boldsymbol d,\boldsymbol\theta) \quad \text{on} \quad \Sigma$$

(Nobile, Vergara '08, Badia et al. '08, Guidoboni et al. '09,...)

☑ Splitting via displacement extrapolation:

$$\boldsymbol\sigma(\boldsymbol u^n,p^n)\boldsymbol n + \frac{\rho^{\mathrm{s}}\epsilon}{\tau}\boldsymbol u^n = \frac{\rho^{\mathrm{s}}\epsilon}{\tau}\dot{\boldsymbol d}^{n-1} - \boldsymbol L^{\mathrm{e}}_{\boldsymbol d}(\boldsymbol d^\star,\boldsymbol\theta^\star) \quad \text{on} \quad \Sigma\,, \qquad \boldsymbol d^\star = \begin{cases}\boldsymbol d^{n-1}\\ \boldsymbol d^{n-1} + \tau\dot{\boldsymbol d}^{n-1}\end{cases}$$

Robin-Neumann Loosely Coupled Schemes

1) Solve fluid:

$$\begin{cases} \rho^{\mathrm{f}} \partial_\tau \boldsymbol{u}^n - \mathbf{div}\, \boldsymbol{\sigma}(\boldsymbol{u}^n, p^n) = \mathbf{0} \quad \text{in} \quad \Omega^{\mathrm{f}} \\ \mathrm{div}\, \boldsymbol{u}^n = 0 \quad \text{in} \quad \Omega^{\mathrm{f}} \\ \color{red}{\boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)\boldsymbol{n} + \frac{\rho^{\mathrm{s}}\epsilon}{\tau}\boldsymbol{u}^n} = \frac{\rho^{\mathrm{s}}\epsilon}{\tau}\dot{\boldsymbol{d}}^{n-1} - \boldsymbol{L}^{\mathrm{e}}_{\boldsymbol{d}}(\color{blue}{\boldsymbol{d}^\star, \boldsymbol{\theta}^\star}) \quad \text{on} \quad \Sigma \end{cases}$$

Robin-Neumann Loosely Coupled Schemes

1) Solve fluid:

$$
\begin{cases}
\rho^{\mathrm{f}} \partial_\tau \boldsymbol{u}^n - \mathbf{div}\, \boldsymbol{\sigma}(\boldsymbol{u}^n, p^n) = \mathbf{0} & \text{in} \quad \Omega^{\mathrm{f}} \\
\mathrm{div}\, \boldsymbol{u}^n = 0 & \text{in} \quad \Omega^{\mathrm{f}} \\
\boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)\boldsymbol{n} + \dfrac{\rho^{\mathrm{s}}\epsilon}{\tau}\boldsymbol{u}^n = \dfrac{\rho^{\mathrm{s}}\epsilon}{\tau}\dot{\boldsymbol{d}}^{n-1} - \boldsymbol{L}^{\mathrm{e}}_{\boldsymbol{d}}(\boldsymbol{d}^\star, \boldsymbol{\theta}^\star) & \text{on} \quad \Sigma
\end{cases}
$$

2) Solve solid:

$$
\begin{cases}
\begin{pmatrix} \rho^{\mathrm{s}}\epsilon\partial_\tau \dot{\boldsymbol{d}}^n \\ 0 \end{pmatrix} - \begin{pmatrix} \boldsymbol{L}^{\mathrm{e}}_{\boldsymbol{d}}(\boldsymbol{d}^n, \boldsymbol{\theta}^n) \\ L^{\mathrm{e}}_{\boldsymbol{\theta}}(\boldsymbol{d}^n, \boldsymbol{\theta}^n) \end{pmatrix} = \begin{pmatrix} -\boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)\boldsymbol{n} \\ \mathbf{0} \end{pmatrix} & \text{on} \quad \Sigma \\
\dot{\boldsymbol{d}}^n = \partial_\tau \boldsymbol{d}^n & \text{on} \quad \Sigma
\end{cases}
$$

Robin-Neumann Loosely Coupled Schemes

1) Solve fluid:

$$\begin{cases} \rho^{\mathrm{f}}\partial_\tau \boldsymbol{u}^n - \mathbf{div}\,\boldsymbol{\sigma}(\boldsymbol{u}^n, p^n) = \mathbf{0} \quad \text{in} \quad \Omega^{\mathrm{f}} \\ \operatorname{div} \boldsymbol{u}^n = 0 \quad \text{in} \quad \Omega^{\mathrm{f}} \\ \boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)\boldsymbol{n} + \dfrac{\rho^{\mathrm{s}}\epsilon}{\tau}\boldsymbol{u}^n = \dfrac{\rho^{\mathrm{s}}\epsilon}{\tau}\dot{\boldsymbol{d}}^{n-1} - \boldsymbol{L}^{\mathrm{e}}_{\boldsymbol{d}}(\boldsymbol{d}^\star, \boldsymbol{\theta}^\star) \quad \text{on} \quad \Sigma \end{cases}$$

2) Solve solid:

$$\begin{cases} \begin{pmatrix} \rho^{\mathrm{s}}\epsilon\partial_\tau\dot{\boldsymbol{d}}^n \\ 0 \end{pmatrix} - \begin{pmatrix} \boldsymbol{L}^{\mathrm{e}}_{\boldsymbol{d}}(\boldsymbol{d}^n, \boldsymbol{\theta}^n) \\ L^{\mathrm{e}}_{\boldsymbol{\theta}}(\boldsymbol{d}^n, \boldsymbol{\theta}^n) \end{pmatrix} = \begin{pmatrix} -\boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)\boldsymbol{n} \\ \mathbf{0} \end{pmatrix} \quad \text{on} \quad \Sigma \\ \dot{\boldsymbol{d}}^n = \partial_\tau \boldsymbol{d}^n \quad \text{on} \quad \Sigma \end{cases}$$

Remarks:

- Semi-implicit coupling scheme which becomes explicit (thin-solid model)

Alternative Formulations (same method)

- Explicit Robin-Neumann coupling:

$$\left\{\begin{aligned} \boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)\boldsymbol{n} + \frac{\rho^{\mathrm{s}}\epsilon}{\tau}\boldsymbol{u}^n &= \frac{\rho^{\mathrm{s}}\epsilon}{\tau}\dot{\boldsymbol{d}}^{n-1} - \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^{\star} \quad \text{on} \quad \Sigma \\ \rho^{\mathrm{s}}\epsilon\partial_\tau\dot{\boldsymbol{d}}^n + \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^n &= -\boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)\boldsymbol{n} \quad \text{on} \quad \Sigma \end{aligned}\right.$$

Alternative Formulations (same method)

- Explicit Robin-Neumann coupling:

$$\left\{\begin{aligned} \boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)\boldsymbol{n} + \frac{\rho^{\mathrm{s}}\epsilon}{\tau}\boldsymbol{u}^n &= \frac{\rho^{\mathrm{s}}\epsilon}{\tau}\dot{\boldsymbol{d}}^{n-1} - \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^{\star} \quad \text{on} \quad \Sigma \\ \rho^{\mathrm{s}}\epsilon\partial_\tau\dot{\boldsymbol{d}}^{n} + \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^n &= -\boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)\boldsymbol{n} \quad \text{on} \quad \Sigma \end{aligned}\right.$$

omit rotations

Alternative Formulations (same method)

- Explicit Robin-Neumann coupling:

$$\left\{\begin{aligned} \boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)\boldsymbol{n} + \frac{\rho^{\mathrm{s}}\epsilon}{\tau}\boldsymbol{u}^n &= \frac{\rho^{\mathrm{s}}\epsilon}{\tau}\dot{\boldsymbol{d}}^{n-1} - \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^{\star} \quad \text{on} \quad \Sigma\\ \rho^{\mathrm{s}}\epsilon\partial_{\tau}\dot{\boldsymbol{d}}^{n} + \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^{n} &= -\boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)\boldsymbol{n} \quad \text{on} \quad \Sigma \end{aligned}\right.$$

Alternative Formulations (same method)

- Explicit Robin-Neumann coupling:

$$\left\{\begin{aligned} \boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)\boldsymbol{n} + \frac{\rho^{\mathrm{s}}\epsilon}{\tau}\boldsymbol{u}^n &= \frac{\rho^{\mathrm{s}}\epsilon}{\tau}\dot{\boldsymbol{d}}^{n-1} - \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^\star \quad \text{on} \quad \Sigma \\ \rho^{\mathrm{s}}\epsilon\partial_\tau\dot{\boldsymbol{d}}^n + \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^n &= -\boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)\boldsymbol{n} \quad \text{on} \quad \Sigma \end{aligned}\right.$$

- Robin based kinematic relaxation:

$$\boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)\boldsymbol{n} + \frac{\rho^{\mathrm{s}}\epsilon}{\tau}\boldsymbol{u}^n = \frac{\rho^{\mathrm{s}}\epsilon}{\tau}\left(\dot{\boldsymbol{d}}^{n-1} + \tau\partial_\tau\dot{\boldsymbol{d}}^\star\right) + \boldsymbol{\sigma}(\boldsymbol{u}^\star, p^\star)\boldsymbol{n} \quad \text{on} \quad \Sigma$$

Alternative Formulations (same method)

- Explicit Robin-Neumann coupling:

$$\left\{\begin{aligned} \boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)\boldsymbol{n} + \frac{\rho^{\mathrm{s}}\epsilon}{\tau}\boldsymbol{u}^n &= \frac{\rho^{\mathrm{s}}\epsilon}{\tau}\dot{\boldsymbol{d}}^{n-1} - \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^{\star} \quad \text{on} \quad \Sigma \\ \rho^{\mathrm{s}}\epsilon\partial_\tau\dot{\boldsymbol{d}}^n + \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^n &= -\boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)\boldsymbol{n} \quad \text{on} \quad \Sigma \end{aligned}\right.$$

- Robin based kinematic relaxation:

$$\boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)\boldsymbol{n} + \frac{\rho^{\mathrm{s}}\epsilon}{\tau}\boldsymbol{u}^n = \frac{\rho^{\mathrm{s}}\epsilon}{\tau}\left(\dot{\boldsymbol{d}}^{n-1} + \tau\partial_\tau\dot{\boldsymbol{d}}^{\star}\right) + \boldsymbol{\sigma}(\boldsymbol{u}^{\star}, p^{\star})\boldsymbol{n} \quad \text{on} \quad \Sigma$$

- *Incremental* displacement-correction:

$$\frac{\rho^{\mathrm{s}}\epsilon}{\tau}\left(\dot{\boldsymbol{d}}^n - \boldsymbol{u}^n\right) + \boldsymbol{L}^{\mathrm{e}}\left(\boldsymbol{d}^n - \boldsymbol{d}^{\star}\right) = 0 \quad \text{on} \quad \Sigma$$

Alternative Formulations (same method)

- Explicit Robin-Neumann coupling:

$$\left\{\begin{aligned} \boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)\boldsymbol{n} + \frac{\rho^{\mathrm{s}}\epsilon}{\tau}\boldsymbol{u}^n &= \frac{\rho^{\mathrm{s}}\epsilon}{\tau}\dot{\boldsymbol{d}}^{n-1} - \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^{\star} \quad \text{on} \quad \Sigma \\ \rho^{\mathrm{s}}\epsilon\partial_\tau\dot{\boldsymbol{d}}^{n} + \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^n &= -\boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)\boldsymbol{n} \quad \text{on} \quad \Sigma \end{aligned}\right.$$

- Robin based kinematic relaxation:

$$\boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)\boldsymbol{n} + \frac{\rho^{\mathrm{s}}\epsilon}{\tau}\boldsymbol{u}^n = \frac{\rho^{\mathrm{s}}\epsilon}{\tau}\left(\dot{\boldsymbol{d}}^{n-1} + \tau\partial_\tau\dot{\boldsymbol{d}}^{\star}\right) + \boldsymbol{\sigma}(\boldsymbol{u}^{\star}, p^{\star})\boldsymbol{n} \quad \text{on} \quad \Sigma$$

- *Incremental* displacement-correction:

$$\frac{\rho^{\mathrm{s}}\epsilon}{\tau}\left(\dot{\boldsymbol{d}}^{n} - \boldsymbol{u}^n\right) + \boldsymbol{L}^{\mathrm{e}}\left(\boldsymbol{d}^n - \boldsymbol{d}^{\star}\right) = 0 \quad \text{on} \quad \Sigma$$

Remark:

For $\boldsymbol{d}^{\star} = \boldsymbol{0}$ (*non-incremetal* displacement-correction) and membrane we retrieve the kinematically coupled scheme *(Guidoboni et al. '09)*

Stability and Accuracy: Main Principle

(M. Fernández '11, '12)

- These loosely coupled schemes enforce:

$$\begin{cases} \rho^{\mathrm{s}}\epsilon\partial_\tau \dot{\boldsymbol{d}}^n + \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^n = -\boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)\boldsymbol{n} & \text{on} \quad \Sigma \\ \dfrac{\rho^{\mathrm{s}}\epsilon}{\tau}\left(\dot{\boldsymbol{d}}^n - \boldsymbol{u}^n\right) + \boldsymbol{L}^{\mathrm{e}}\left(\boldsymbol{d}^n - \boldsymbol{d}^\star\right) = 0 & \text{on} \quad \Sigma \end{cases}$$

Stability and Accuracy: Main Principle

(M. Fernández '11, '12)

- These loosely coupled schemes enforce:

$$\left\{\begin{aligned} &\rho^{\mathrm{s}}\epsilon\partial_\tau\dot{\boldsymbol{d}}^n + \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^n = -\boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)\boldsymbol{n} \quad \text{on} \quad \Sigma \\ &\frac{\rho^{\mathrm{s}}\epsilon}{\tau}\left(\dot{\boldsymbol{d}}^n - \boldsymbol{u}^n\right) + \boldsymbol{L}^{\mathrm{e}}\left(\boldsymbol{d}^n - \boldsymbol{d}^\star\right) = 0 \quad \text{on} \quad \Sigma \end{aligned}\right.$$

$\Big\}$ **implicit kinetic coupling**

Stability and Accuracy: Main Principle

(M. Fernández '11, '12)

- These loosely coupled schemes enforce:

$$\left\{\begin{aligned} \rho^{\mathrm{s}}\epsilon\partial_\tau\dot{\boldsymbol{d}}^n + \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^n &= -\boldsymbol{\sigma}(\boldsymbol{u}^n,p^n)\boldsymbol{n} \quad \text{on} \quad \Sigma \\ \frac{\rho^{\mathrm{s}}\epsilon}{\tau}\big(\dot{\boldsymbol{d}}^n - \boldsymbol{u}^n\big) + \boldsymbol{L}^{\mathrm{e}}\left(\boldsymbol{d}^n - \boldsymbol{d}^\star\right) &= 0 \quad \text{on} \quad \Sigma \end{aligned}\right.$$

$$\Big\} \text{ implicit kinetic coupling}$$

$$\Downarrow$$

$$\boldsymbol{u}^n = \dot{\boldsymbol{d}}^n + \frac{\tau}{\rho^{\mathrm{s}}\epsilon}\boldsymbol{L}^{\mathrm{e}}\big(\boldsymbol{d}^n - \boldsymbol{d}^\star\big) \quad \text{on} \quad \Sigma$$

Stability and Accuracy: Main Principle

(M. Fernández '11, '12)

- These loosely coupled schemes enforce:

$$\left\{\begin{aligned} &\rho^{\mathrm{s}}\epsilon\partial_\tau\dot{\boldsymbol{d}}^n + \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^n = -\boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)\boldsymbol{n} \quad \text{on} \quad \Sigma \Big\}\ \textbf{implicit kinetic coupling}\\ &\frac{\rho^{\mathrm{s}}\epsilon}{\tau}\big(\dot{\boldsymbol{d}}^n - \boldsymbol{u}^n\big) + \boldsymbol{L}^{\mathrm{e}}\left(\boldsymbol{d}^n - \boldsymbol{d}^\star\right) = 0 \quad \text{on} \quad \Sigma \end{aligned}\right.$$

$$\Downarrow$$

$$\boldsymbol{u}^n = \dot{\boldsymbol{d}}^n + \frac{\tau}{\rho^{\mathrm{s}}\epsilon}\boldsymbol{L}^{\mathrm{e}}\big(\boldsymbol{d}^n - \boldsymbol{d}^\star\big) \quad \text{on} \quad \Sigma \Big\}$$

kinematic perturbation of implicit coupling

Stability and Accuracy: Main Principle

(M. Fernández '11, '12)

- These loosely coupled schemes enforce:

$$\left\{\begin{aligned} \rho^{\mathrm{s}}\epsilon\partial_\tau\dot{\boldsymbol{d}}^n + \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^n &= -\boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)\boldsymbol{n} \quad \text{on} \quad \Sigma \;\Big\}\ \textbf{implicit kinetic coupling}\\ \boldsymbol{u}^n &= \dot{\boldsymbol{d}}^n + \frac{\tau}{\rho^{\mathrm{s}}\epsilon}\boldsymbol{L}^{\mathrm{e}}\left(\boldsymbol{d}^n - \boldsymbol{d}^\star\right) \quad \text{on} \quad \Sigma \;\Big\}\ \begin{array}{c}\textbf{kinematic perturbation of}\\ \textbf{implicit coupling}\end{array} \end{aligned}\right.$$

Stability and Accuracy: Main Principle

(M. Fernández '11, '12)

- These loosely coupled schemes enforce:

$$\left\{\begin{array}{rll} \rho^{\mathrm{s}}\epsilon\partial_\tau \dot{\boldsymbol{d}}^n + \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^n = -\boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)\boldsymbol{n} & \text{on} & \Sigma \Big\} \ \textbf{implicit kinetic coupling} \\[2ex] \boldsymbol{u}^n = \dot{\boldsymbol{d}}^n + \dfrac{\tau}{\rho^{\mathrm{s}}\epsilon}\boldsymbol{L}^{\mathrm{e}}\left(\boldsymbol{d}^n - \boldsymbol{d}^\star\right) & \text{on} & \Sigma \Big\} \ \begin{array}{c}\textbf{kinematic perturbation of} \\ \textbf{implicit coupling}\end{array} \end{array}\right.$$

- Key issue: how does this kinematic perturbation affect the stability and accuracy of the 'underlying' implicit coupling scheme?

Stability and Accuracy: Main Principle

(M. Fernández '11, '12)

- These loosely coupled schemes enforce:

$$\left\{\begin{array}{ll} \rho^{\mathrm{s}}\epsilon\partial_\tau \dot{\boldsymbol{d}}^n + \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^n = -\boldsymbol{\sigma}(\boldsymbol{u}^n,p^n)\boldsymbol{n} \quad \text{on} \quad \Sigma & \Big\} \ \textbf{implicit kinetic coupling} \\[2ex] \boldsymbol{u}^n = \dot{\boldsymbol{d}}^n + \dfrac{\tau}{\rho^{\mathrm{s}}\epsilon}\boldsymbol{L}^{\mathrm{e}}(\boldsymbol{d}^n - \boldsymbol{d}^\star) \quad \text{on} \quad \Sigma & \Big\} \ \begin{array}{c}\textbf{kinematic perturbation of} \\ \textbf{implicit coupling}\end{array} \end{array}\right.$$

- Key issue: how does this kinematic perturbation affect the stability and accuracy of the 'underlying' implicit coupling scheme?

Remark:

The size of the perturbation depends on the displacement extrapolation

$$\boldsymbol{d}^\star = \boldsymbol{0} \qquad \text{(sub-optimal?)}$$

$$\boldsymbol{d}^\star = \left\{\begin{array}{l} \boldsymbol{d}^{n-1} \\ \boldsymbol{d}^{n-1} + \tau\dot{\boldsymbol{d}}^{n-1} \end{array}\right. \qquad \text{(optimal?)}$$

Stability: A Priori Energy Estimates

- Energy-norm:

$$E^n \overset{\text{def}}{=} \frac{\rho^{\mathrm{f}}}{2}\|\boldsymbol{u}^n\|^2_{0,\Omega^{\mathrm{f}}} + \frac{\rho^{\mathrm{s}}\epsilon}{2}\|\dot{\boldsymbol{d}}^n\|^2_{0,\Sigma} + \frac{1}{2}\|\boldsymbol{d}^n\|^2_{\mathrm{e}}$$

Proposition:

For $n \geq 1$, there holds

$$E^n \lesssim E^0 \quad \begin{cases} \text{if} \quad \boldsymbol{d}^\star = \boldsymbol{0} \\ \text{if} \quad \boldsymbol{d}^\star = \boldsymbol{d}^{n-1} \\ \text{if} \quad \begin{cases} \boldsymbol{d}^\star = \boldsymbol{d}^{n-1} + \tau\dot{\boldsymbol{d}}^{n-1} \\ \tau\omega_{\mathrm{e}}^{\frac{6}{5}} = \mathcal{O}\big(h^{\frac{6}{5}}\big) \end{cases} \end{cases}$$

with $\omega_{\mathrm{e}} \overset{\text{def}}{=} \sqrt{\beta_{\mathrm{e}}/(\rho^{\mathrm{s}}\epsilon)}$.

Stability: A Priori Energy Estimates

- Energy-norm:

$$E^n \overset{\text{def}}{=} \frac{\rho^{\text{f}}}{2}\|\boldsymbol{u}^n\|^2_{0,\Omega^{\text{f}}} + \frac{\rho^{\text{s}}\epsilon}{2}\|\dot{\boldsymbol{d}}^n\|^2_{0,\Sigma} + \frac{1}{2}\|\boldsymbol{d}^n\|^2_{\text{e}}$$

Proposition:

For $n \geq 1$, there holds

$$E^n \lesssim E^0 \quad \begin{cases} \text{if} \quad \boldsymbol{d}^\star = \boldsymbol{0} \\ \text{if} \quad \boldsymbol{d}^\star = \boldsymbol{d}^{n-1} \\ \text{if} \quad \begin{cases} \boldsymbol{d}^\star = \boldsymbol{d}^{n-1} + \tau\dot{\boldsymbol{d}}^{n-1} \\ \tau\omega_{\text{e}}^{\frac{6}{5}} = \mathcal{O}(h^{\frac{6}{5}}) \end{cases} \end{cases}$$

with $\omega_{\text{e}} \overset{\text{def}}{=} \sqrt{\beta_{\text{e}}/(\rho^{\text{s}}\epsilon)}$.

Remarks:

- Incremental 1st-order extrap. unconditionnally stable
- Incremental 2nd-order extrap. stable under 6/5-CFL condition
- Stability independent of the added-mass effect

Convergence: A Priori Error Estimates

- Energy-norm error:

$$e^n \overset{\text{def}}{=} \sqrt{\frac{\rho^{\mathrm{f}}}{2}\|\boldsymbol{u}^n - \boldsymbol{u}(t_n)\|^2_{0,\Omega^{\mathrm{f}}} + \frac{\rho^{\mathrm{s}}\epsilon}{2}\|\dot{\boldsymbol{d}}^n - \dot{\boldsymbol{d}}(t_n)\|^2_{0,\Sigma} + \frac{1}{2}\|\boldsymbol{d}^n - \boldsymbol{d}(t_n)\|^2_{\mathrm{e}}}$$

Proposition:
For smooth enough solutions and $n \geq 1$, there holds:

$$e^n \lesssim h^k + \tau + \frac{\beta_{\mathrm{e}}}{\sqrt{\rho^{\mathrm{s}}\epsilon}} \cdot \begin{cases} \tau^{\frac{1}{2}} & \text{if} \quad \boldsymbol{d}^\star = \boldsymbol{0} \\ \tau & \text{if} \quad \boldsymbol{d}^\star = \boldsymbol{d}^{n-1} \\ \tau^2 & \text{if} \quad \boldsymbol{d}^\star = \boldsymbol{d}^{n-1} + \tau\dot{\boldsymbol{d}}^{n-1} \end{cases}$$

with $k \geq 1$ the convergence order of the Stokes-projection.

Convergence: A Priori Error Estimates

- Energy-norm error:

$$e^n \overset{\text{def}}{=} \sqrt{\frac{\rho^{\mathrm{f}}}{2}\|\boldsymbol{u}^n - \boldsymbol{u}(t_n)\|^2_{0,\Omega^{\mathrm{f}}} + \frac{\rho^{\mathrm{s}}\epsilon}{2}\|\dot{\boldsymbol{d}}^n - \dot{\boldsymbol{d}}(t_n)\|^2_{0,\Sigma} + \frac{1}{2}\|\boldsymbol{d}^n - \boldsymbol{d}(t_n)\|^2_{\mathrm{e}}}$$

Proposition:

For smooth enough solutions and $n \geq 1$, there holds:

$$e^n \lesssim h^k + \tau + \frac{\beta_{\mathrm{e}}}{\sqrt{\rho^{\mathrm{s}}\epsilon}} \cdot \begin{cases} \tau^{\frac{1}{2}} & \text{if} \quad \boldsymbol{d}^\star = \boldsymbol{0} \\ \tau & \text{if} \quad \boldsymbol{d}^\star = \boldsymbol{d}^{n-1} \\ \tau^2 & \text{if} \quad \boldsymbol{d}^\star = \boldsymbol{d}^{n-1} + \tau \dot{\boldsymbol{d}}^{n-1} \end{cases}$$

with $k \geq 1$ the convergence order of the Stokes-projection.

Remarks:

- Non-incremental: expected sub-optimal time accuracy
- Incremental: overall optimal accuracy
- Splitting error constant depends on physical parameters

Generalization to non-linear

☑ Fluid: Navier-Stokes (ALE formalism)

☑ Solid: non-linear shell (complete strain tensor)

Ambiguity in the computation of the **second order extrapolation**

$$\begin{aligned} \boldsymbol{d}^\star &= \boldsymbol{d}^{n-1} + \tau \dot{\boldsymbol{d}}^{n-1} = 2\boldsymbol{d}^{n-1} - \boldsymbol{d}^{n-2} \\ \boldsymbol{\theta}^\star &= 2\boldsymbol{\theta}^{n-1} - \boldsymbol{\theta}^{n-2} \end{aligned}$$

$$\boldsymbol{L}^{\mathrm{e}}_{\boldsymbol{d}}((\boldsymbol{d}^\star, \boldsymbol{\theta}^\star)) = \begin{cases} \boldsymbol{L}^{\mathrm{e}}_{\boldsymbol{d}}((2\boldsymbol{d}^{n-1} - \boldsymbol{d}^{n-2}, 2\boldsymbol{\theta}^{n-1} - \boldsymbol{\theta}^{n-2})) \\ 2\boldsymbol{L}^{\mathrm{e}}_{\boldsymbol{d}}((\boldsymbol{d}^{n-1}, \boldsymbol{\theta}^{n-1})) - \boldsymbol{L}^{\mathrm{e}}_{\boldsymbol{d}}\boldsymbol{d}((\boldsymbol{d}^{n-2}, \boldsymbol{\theta}^{n-2})) \end{cases}$$

Generalization to non-linear

☑ Fluid: Navier-Stokes (ALE formalism)
☑ Solid: non-linear shell (complete strain tensor)

Ambiguity in the computation of the **second order extrapolation**

$$\begin{aligned} \boldsymbol{d}^\star &= \boldsymbol{d}^{n-1} + \tau \dot{\boldsymbol{d}}^{n-1} = 2\boldsymbol{d}^{n-1} - \boldsymbol{d}^{n-2} \\ \boldsymbol{\theta}^\star &= 2\boldsymbol{\theta}^{n-1} - \boldsymbol{\theta}^{n-2} \end{aligned}$$

$$\boldsymbol{L}^{\mathrm{e}}_{\boldsymbol{d}}((\boldsymbol{d}^\star, \boldsymbol{\theta}^\star)) = \begin{cases} \boldsymbol{L}^{\mathrm{e}}_{\boldsymbol{d}}((2\boldsymbol{d}^{n-1} - \boldsymbol{d}^{n-2}, 2\boldsymbol{\theta}^{n-1} - \boldsymbol{\theta}^{n-2})) & \longleftarrow \text{does not work} \\ 2\boldsymbol{L}^{\mathrm{e}}_{\boldsymbol{d}}((\boldsymbol{d}^{n-1}, \boldsymbol{\theta}^{n-1})) - \boldsymbol{L}^{\mathrm{e}}_{\boldsymbol{d}}\boldsymbol{d}((\boldsymbol{d}^{n-2}, \boldsymbol{\theta}^{n-2})) \end{cases}$$

Generalization to non-linear

☑ Fluid: Navier-Stokes (ALE formalism)
☑ Solid: non-linear shell (complete strain tensor)

Ambiguity in the computation of the **second order extrapolation**

$$\begin{aligned} \boldsymbol{d}^\star &= \boldsymbol{d}^{n-1} + \tau \dot{\boldsymbol{d}}^{n-1} = 2\boldsymbol{d}^{n-1} - \boldsymbol{d}^{n-2} \\ \boldsymbol{\theta}^\star &= 2\boldsymbol{\theta}^{n-1} - \boldsymbol{\theta}^{n-2} \end{aligned}$$

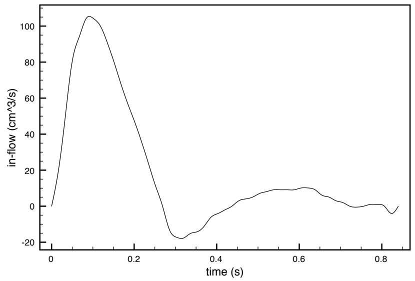
$$\boldsymbol{L}^{\mathrm{e}}_{\boldsymbol{d}}((\boldsymbol{d}^\star, \boldsymbol{\theta}^\star)) = \begin{cases} \boldsymbol{L}^{\mathrm{e}}_{\boldsymbol{d}}((2\boldsymbol{d}^{n-1} - \boldsymbol{d}^{n-2}, 2\boldsymbol{\theta}^{n-1} - \boldsymbol{\theta}^{n-2})) \\ 2\boldsymbol{L}^{\mathrm{e}}_{\boldsymbol{d}}((\boldsymbol{d}^{n-1}, \boldsymbol{\theta}^{n-1})) - \boldsymbol{L}^{\mathrm{e}}_{\boldsymbol{d}}\boldsymbol{d}((\boldsymbol{d}^{n-2}, \boldsymbol{\theta}^{n-2})) \end{cases}$$

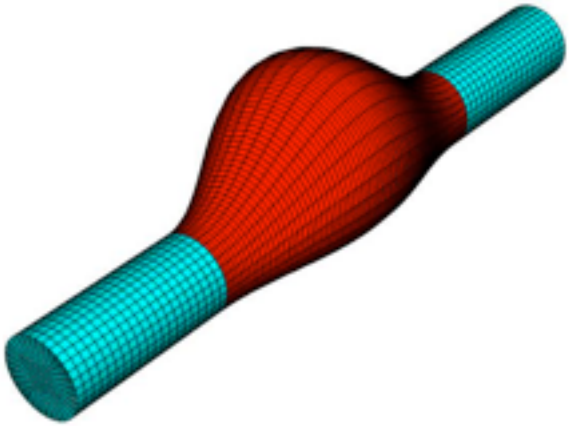
does not work

Our choice

ALE Navier-Stokes/Linear-Shell: In-vitro Abdominal Aortic Aneuvrism

- In-vitro abdominal aortic aneurysm:





$\boldsymbol{\sigma}(\boldsymbol{u},p)\boldsymbol{n} = -R_{\text{out}}Q_{\text{out}}\boldsymbol{n}$

$\boldsymbol{u} = \boldsymbol{u}_{in}$

(Salsac et al. ‘05)

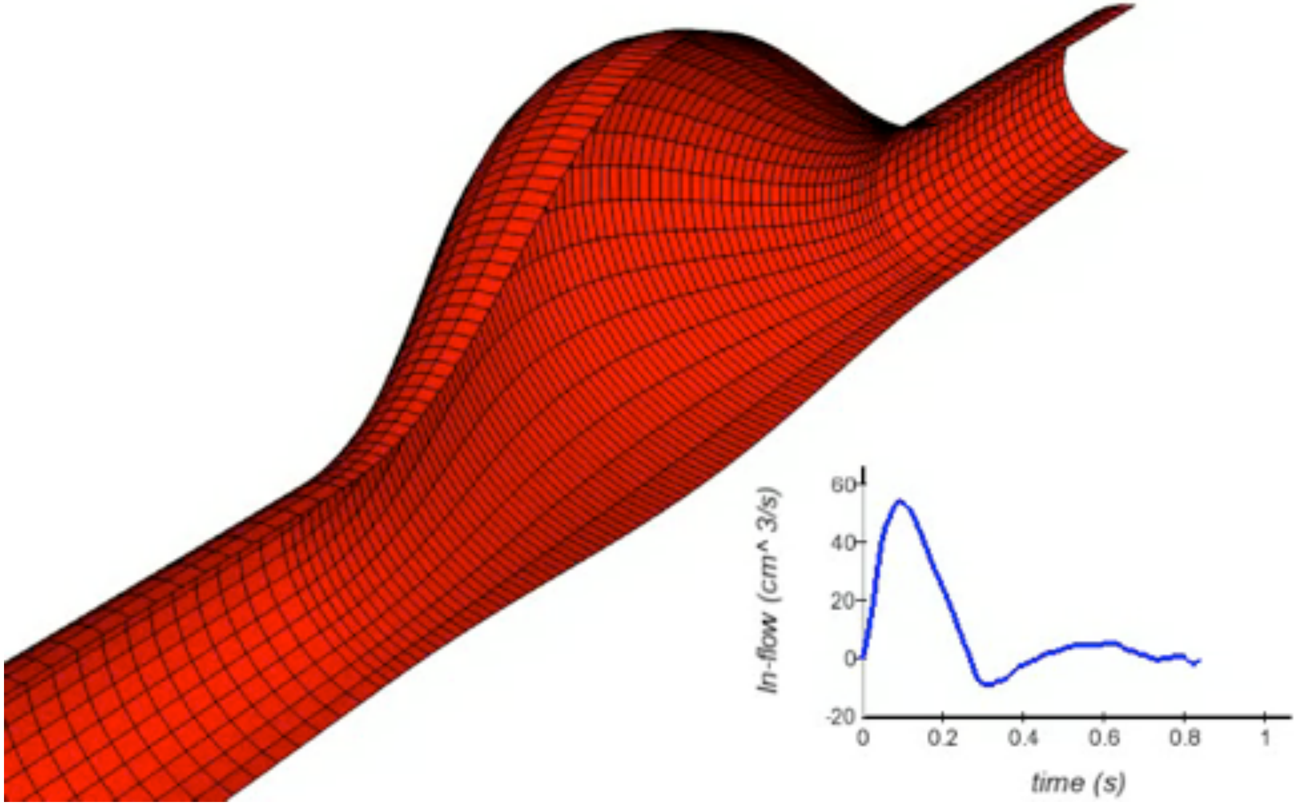
- Thin-solid: based on general shell elements
  (Chapelle, Bathe ’01)

- Physical data:

$$
\begin{aligned}
\epsilon &= 0.17\,\text{cm} & \rho^{\text{f}} &= 1.1\,\text{g}/\text{cm}^3\\
\rho^{\text{s}} &= 1.2\,\text{g}/\text{cm}^3 & \mu &= 0.035\,\text{P}\\
E &= 6\times 10^6\,\text{dyne}/\text{cm}^2 & R_{\text{out}} &= 300\,\text{dyne}\,\text{s}/\text{cm}^5\\
\nu &= 0.3
\end{aligned}
$$

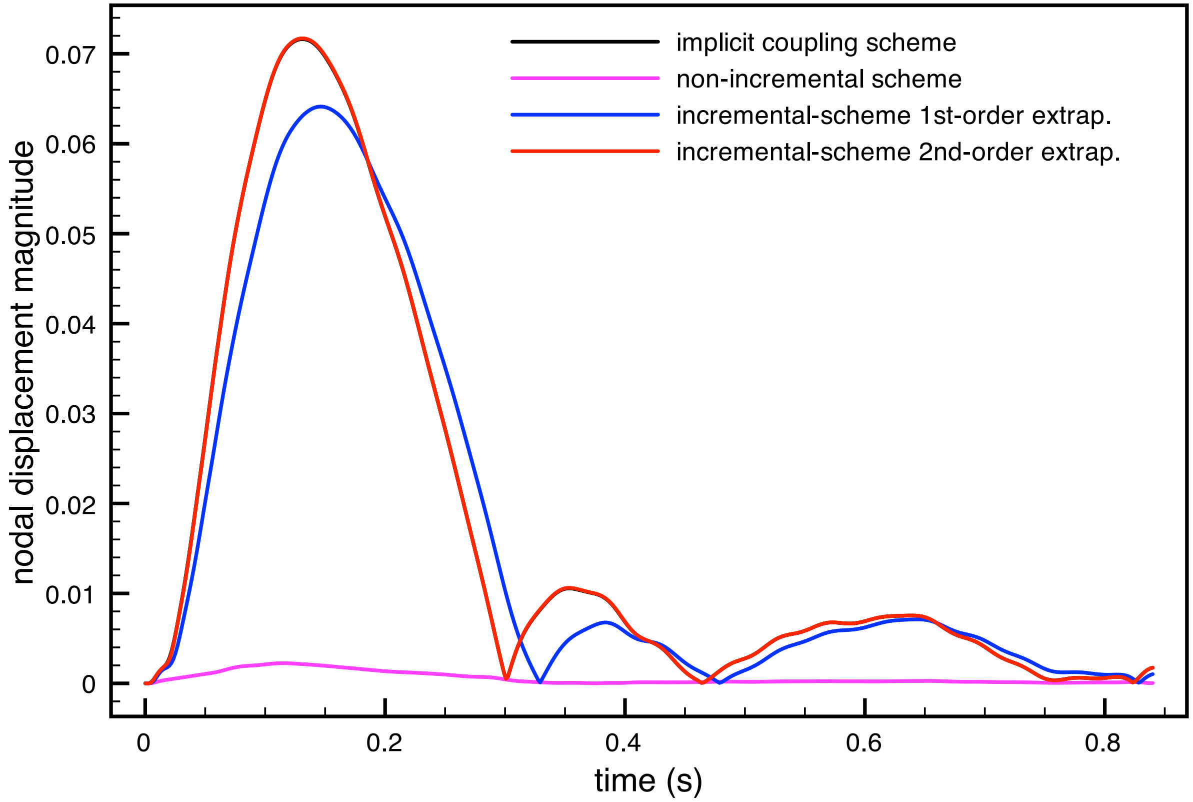
- Space discretization: MITC4 for the solid, $\mathbb{Q}_1/\mathbb{Q}_1$ stabilized for the fluid

ALE Navier-Stokes/Linear-Shell (cont.)



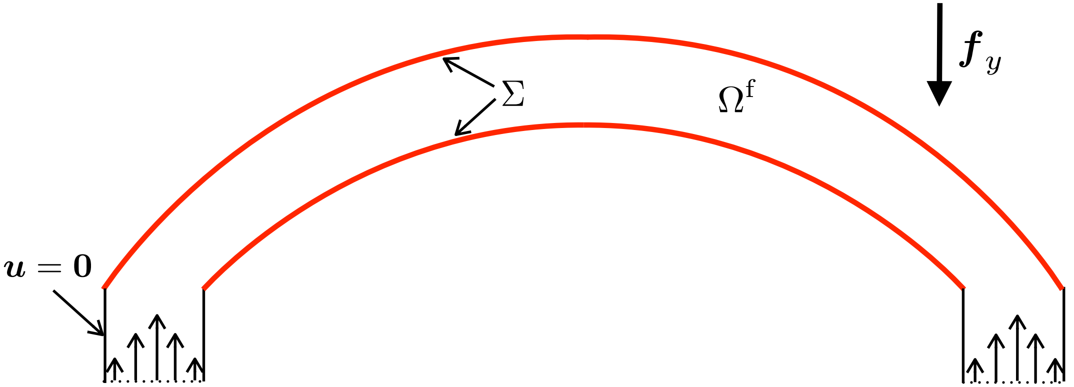
Accuracy

- $\tau = 4.2 \times 10^{-4}$



ALE Navier-Stokes/Non-Linear Shell: Inflating balloon

- Inflating balloon problem (incompressible ‘dilemma’):



(Wall et al. ’06)

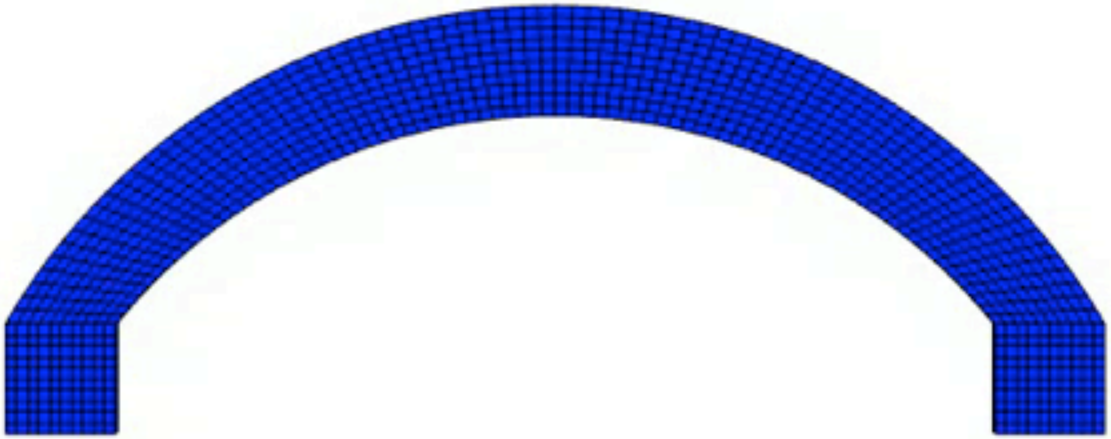
- Thin-solid: based on general shell elements
  (Chapelle, Bathe ’01)
- Physical data:

$$\begin{aligned} \epsilon &= 0.1\,\mathrm{m} & E_{\mathrm{top}} &= 9\times 10^5 \mathrm{N/m^2} & \rho^{\mathrm{f}} &= 1\,\mathrm{kg/m^3} \\ \rho^{\mathrm{s}} &= 500\,\mathrm{kg/m^3} & E_{\mathrm{bottom}} &= 9\times 10^8 \mathrm{N/m^2} & \mu &= 9\,\mathrm{Pa\,s} \\ \nu &= 0.3 \end{aligned}$$

- Space discretization: MITC4 for the solid, $\mathbb{Q}_1/\mathbb{Q}_1$ stabilized for the fluid

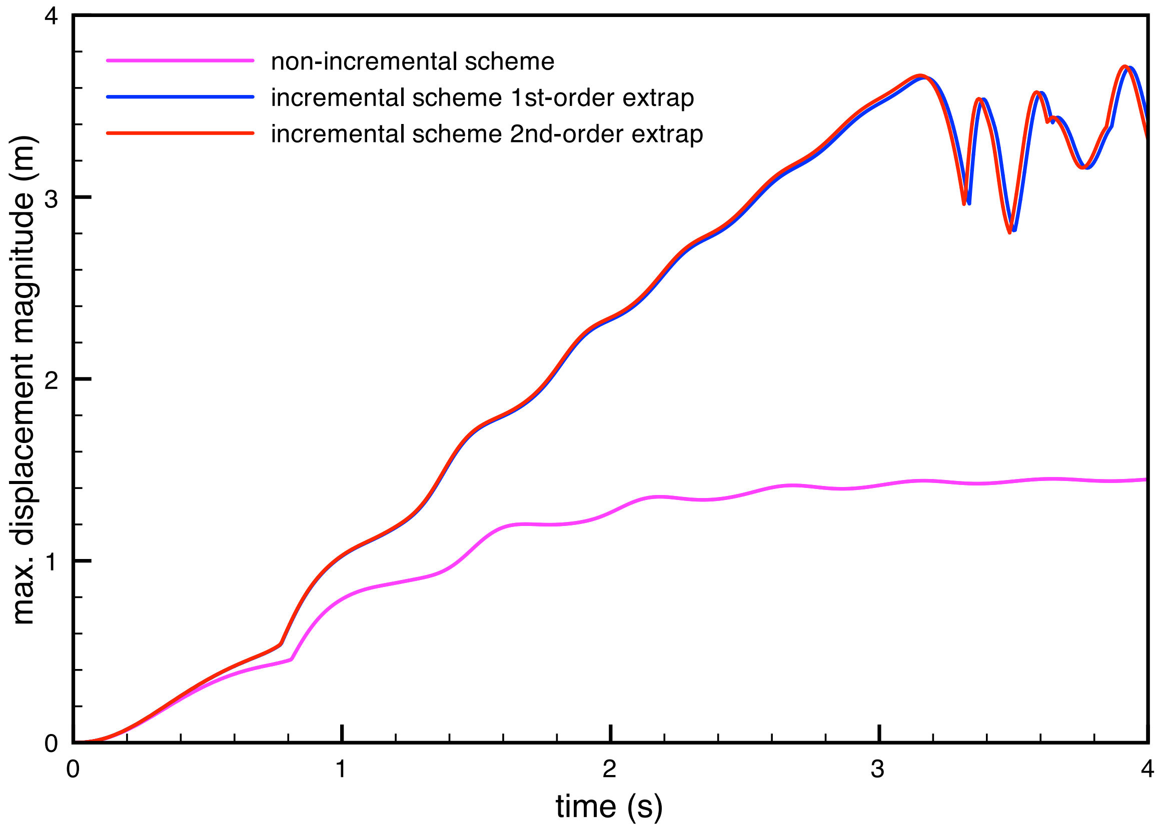
ALE Navier-Stokes/Non-Linear Shell (cont.)

ALE Navier-Stokes/Non-Linear Shell (cont.)



Accuracy

- $\tau = 0.1$



Solid Damping Effects (Viscoelasticity)

$$\rho^{\mathrm{s}}\epsilon\partial_t\dot{\boldsymbol{d}} + \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d} + \boldsymbol{L}^{\mathrm{v}}\dot{\mathrm{d}} = -\boldsymbol{\sigma}(\boldsymbol{u},p)\boldsymbol{n} \quad \text{on} \quad \Sigma$$

Solid Damping Effects (Viscoelasticity)

$$\rho^{\mathrm{s}}\epsilon\partial_t\dot{\boldsymbol{d}} + \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d} + \boldsymbol{L}^{\mathrm{v}}\dot{\boldsymbol{d}} = -\boldsymbol{\sigma}(\boldsymbol{u},p)\boldsymbol{n} \quad \text{on} \quad \Sigma$$

- Explicit Robin-Neumann coupling (incremental 1st-order extrap):

$$\begin{cases} \boldsymbol{\sigma}(\boldsymbol{u}^n,p^n)\boldsymbol{n} + \dfrac{\rho^{\mathrm{s}}\epsilon}{\tau}\boldsymbol{u}^n = \dfrac{\rho^{\mathrm{s}}\epsilon}{\tau}\dot{\boldsymbol{d}}^{n-1} - \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^{n-1} & \text{on} \quad \Sigma \\ \rho^{\mathrm{s}}\epsilon\partial_\tau\dot{\boldsymbol{d}}^n + \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^n = -\boldsymbol{\sigma}(\boldsymbol{u}^n,p^n)\boldsymbol{n} & \text{on} \quad \Sigma \end{cases}$$

Solid Damping Effects (Viscoelasticity)

$$\rho^{\mathrm{s}}\epsilon\partial_t\dot{\boldsymbol{d}} + \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d} + \boldsymbol{L}^{\mathrm{v}}\dot{\boldsymbol{d}} = -\boldsymbol{\sigma}(\boldsymbol{u},p)\boldsymbol{n} \quad \text{on} \quad \Sigma$$

- Explicit Robin-Neumann coupling (incremental 1st-order extrap):

$$\begin{cases} \boldsymbol{\sigma}(\boldsymbol{u}^n,p^n)\boldsymbol{n} + \dfrac{\rho^{\mathrm{s}}\epsilon}{\tau}\boldsymbol{u}^n = \dfrac{\rho^{\mathrm{s}}\epsilon}{\tau}\dot{\boldsymbol{d}}^{n-1} - \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^{n-1} - \boldsymbol{L}^{\mathrm{v}}\dot{\boldsymbol{d}}^{n-1} & \text{on} \quad \Sigma \\ \rho^{\mathrm{s}}\epsilon\partial_\tau\dot{\boldsymbol{d}}^n + \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^n = -\boldsymbol{\sigma}(\boldsymbol{u}^n,p^n)\boldsymbol{n} & \text{on} \quad \Sigma \end{cases}$$

Solid Damping Effects (Viscoelasticity)

$$\rho^{\mathrm{s}}\epsilon\partial_t\dot{\boldsymbol{d}} + \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d} + \boldsymbol{L}^{\mathrm{v}}\dot{\boldsymbol{d}} = -\boldsymbol{\sigma}(\boldsymbol{u},p)\boldsymbol{n} \quad \text{on} \quad \Sigma$$

- Explicit Robin-Neumann coupling (incremental 1st-order extrap):

$$\begin{cases} \boldsymbol{\sigma}(\boldsymbol{u}^n,p^n)\boldsymbol{n} + \dfrac{\rho^{\mathrm{s}}\epsilon}{\tau}\boldsymbol{u}^n = \dfrac{\rho^{\mathrm{s}}\epsilon}{\tau}\dot{\boldsymbol{d}}^{n-1} - \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^{n-1} - \boldsymbol{L}^{\mathrm{v}}\dot{\boldsymbol{d}}^{n-1} & \text{on} \quad \Sigma \\ \rho^{\mathrm{s}}\epsilon\partial_\tau\dot{\boldsymbol{d}}^n + \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^n + \boldsymbol{L}^{\mathrm{v}}\dot{\boldsymbol{d}}^n = -\boldsymbol{\sigma}(\boldsymbol{u}^n,p^n)\boldsymbol{n} & \text{on} \quad \Sigma \end{cases}$$

Solid Damping Effects (Viscoelasticity)

$$\rho^{\mathrm{s}}\epsilon\partial_t\dot{\boldsymbol{d}} + \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d} + \boldsymbol{L}^{\mathrm{v}}\dot{\boldsymbol{d}} = -\boldsymbol{\sigma}(\boldsymbol{u},p)\boldsymbol{n} \quad \text{on} \quad \Sigma$$

- Explicit Robin-Neumann coupling (incremental 1st-order extrap):

$$\begin{cases} \boldsymbol{\sigma}(\boldsymbol{u}^n,p^n)\boldsymbol{n} + \dfrac{\rho^{\mathrm{s}}\epsilon}{\tau}\boldsymbol{u}^n = \dfrac{\rho^{\mathrm{s}}\epsilon}{\tau}\dot{\boldsymbol{d}}^{n-1} - \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^{n-1} - \boldsymbol{L}^{\mathrm{v}}\dot{\boldsymbol{d}}^{n-1} & \text{on} \quad \Sigma \\ \rho^{\mathrm{s}}\epsilon\partial_\tau\dot{\boldsymbol{d}}^n + \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^n + \boldsymbol{L}^{\mathrm{v}}\dot{\boldsymbol{d}}^n = -\boldsymbol{\sigma}(\boldsymbol{u}^n,p^n)\boldsymbol{n} & \text{on} \quad \Sigma \end{cases}$$

Proposition: *(Fernández, Mullaert, MV '13)*

For $n \geq 1$, there holds

$$E^n \lesssim E^0 + \tau^2\|\dot{\boldsymbol{d}}^0\|_{\mathrm{e}}^2 + \frac{\tau^2}{\rho^{\mathrm{s}}\epsilon}\|\boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^0 + \boldsymbol{L}^{\mathrm{v}}\dot{\boldsymbol{d}}^0\|_{0,\Sigma}^2.$$

Solid Damping Effects (Viscoelasticity)

$$\rho^{\mathrm{s}}\epsilon\partial_t\dot{\boldsymbol{d}} + \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d} + \boldsymbol{L}^{\mathrm{v}}\dot{\boldsymbol{d}} = -\boldsymbol{\sigma}(\boldsymbol{u},p)\boldsymbol{n} \quad \text{on} \quad \Sigma$$

- Explicit Robin-Neumann coupling (incremental 1st-order extrap):

$$\begin{cases} \boldsymbol{\sigma}(\boldsymbol{u}^n,p^n)\boldsymbol{n} + \dfrac{\rho^{\mathrm{s}}\epsilon}{\tau}\boldsymbol{u}^n = \dfrac{\rho^{\mathrm{s}}\epsilon}{\tau}\dot{\boldsymbol{d}}^{n-1} - \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^{n-1} - \boldsymbol{L}^{\mathrm{v}}\dot{\boldsymbol{d}}^{n-1} & \text{on} \quad \Sigma \\ \rho^{\mathrm{s}}\epsilon\partial_\tau\dot{\boldsymbol{d}}^n + \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^n + \boldsymbol{L}^{\mathrm{v}}\dot{\boldsymbol{d}}^n = -\boldsymbol{\sigma}(\boldsymbol{u}^n,p^n)\boldsymbol{n} & \text{on} \quad \Sigma \end{cases}$$

Proposition: *(Fernández, Mullaert, MV '13)*

For $n \geq 1$, there holds

$$E^n \lesssim E^0 + \tau^2\|\dot{\boldsymbol{d}}^0\|_{\mathrm{e}}^2 + \frac{\tau^2}{\rho^{\mathrm{s}}\epsilon}\|\boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^0 + \boldsymbol{L}^{\mathrm{v}}\dot{\boldsymbol{d}}^0\|_{0,\Sigma}^2.$$

Remarks:

- Explicit treatment of damping does not compromise unconditional stability

Solid Damping Effects (Viscoelasticity)

$$\rho^{\mathrm{s}}\epsilon\partial_t\dot{\boldsymbol{d}} + \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d} + \boldsymbol{L}^{\mathrm{v}}\dot{\boldsymbol{d}} = -\boldsymbol{\sigma}(\boldsymbol{u},p)\boldsymbol{n} \quad \text{on} \quad \Sigma$$

- Explicit Robin-Neumann coupling (incremental 1st-order extrap):

$$\begin{cases} \boldsymbol{\sigma}(\boldsymbol{u}^n,p^n)\boldsymbol{n} + \dfrac{\rho^{\mathrm{s}}\epsilon}{\tau}\boldsymbol{u}^n = \dfrac{\rho^{\mathrm{s}}\epsilon}{\tau}\dot{\boldsymbol{d}}^{n-1} - \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^{n-1} - \boldsymbol{L}^{\mathrm{v}}\dot{\boldsymbol{d}}^{n-1} & \text{on} \quad \Sigma \\ \rho^{\mathrm{s}}\epsilon\partial_\tau\dot{\boldsymbol{d}}^n + \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^n + \boldsymbol{L}^{\mathrm{v}}\dot{\boldsymbol{d}}^n = -\boldsymbol{\sigma}(\boldsymbol{u}^n,p^n)\boldsymbol{n} & \text{on} \quad \Sigma \end{cases}$$

Proposition: *(Fernández, Mullaert, MV '13)*
For $n \geq 1$, there holds

$$E^n \lesssim E^0 + \tau^2\|\dot{\boldsymbol{d}}^0\|_{\mathrm{e}}^2 + \frac{\tau^2}{\rho^{\mathrm{s}}\epsilon}\|\boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^0 + \boldsymbol{L}^{\mathrm{v}}\dot{\boldsymbol{d}}^0\|_{0,\Sigma}^2.$$

Remarks:

- Explicit treatment of damping does not compromise unconditional stability
- Implicit treatment of damping (kinematically coupled scheme)

$$\boldsymbol{\sigma}(\boldsymbol{u}^n,p^n)\boldsymbol{n} + \frac{\rho^{\mathrm{s}}\epsilon}{\tau}\boldsymbol{u}^n + \boldsymbol{L}^{\mathrm{v}}\boldsymbol{u}^n = \frac{\rho^{\mathrm{s}}\epsilon}{\tau}\dot{\boldsymbol{d}}^{n-1} \quad \text{on} \quad \Sigma$$

yields a non-standard Robin condition *(Guidoboni et al. '09)*

Parallelization of the solid and fluid solvers

- Design algorithms **well suited for parallel computing** : **use domain decomposition**

  □ Additive Schwarz for the fluid (PETSCI)
  ☑ **BDD** Balanced domain decomposition method for the solid solver

Alternatives for the solid solver

- **FETI** (finite element tearing and interconnect) *Farhat, Roux*
- **BDDC** (balancing domain decomposition by constraints) *C. R. Dohrmann*

Revisit domain decomposition (linear elasticity)

Solve by an iterative method the primal Shur complement (interface problem)

$$\begin{aligned} S\bar{X} &= \bar{F} \\ S &= \sum_i \mathbf{R}^{i^t}\mathbf{S}^i\mathbf{R}^i \end{aligned}$$

Neumann-Neumann preconditioner (*Bourgat, Glowinski, Le Tallec, De Roeck, MV*)

$$\mathbf{M}^{-1} = (\sum_i \mathbf{D}^i\tilde{\mathbf{S}}_i^{-1}\mathbf{D}^{i^t})$$

Balanced domain decomposition brilliant idea *J. Mandel '92 '93*

Instead of tacking **arbitrary rigid bodies** in the solution of the Neumann problems choose them in order **to minimize residual of the next iteration**

Balanced Domain Decomposition Method

Particular case of the additive Schwarz method applied to interface problem

$$(\sum_i \mathbf{R}^{i^t}\mathbf{S}^i\mathbf{R}^i)\bar{X} = \bar{F}$$

Define:

- the space of global interface values $\mathbf{V} = \{\bar{v} = Tr\ v|_\Gamma, v \in \mathbf{H}(\Omega)\}$
- a partition of unity $\mathbf{D}^i : Tr\mathbf{V}_{|\Gamma_i} \to \mathbf{V}$
- an approximate local operator $\tilde{\mathbf{S}}^i$ st $\tilde{\mathbf{S}} = \sum \mathbf{R}^{i^t}\tilde{\mathbf{S}}^i\mathbf{R}^i.$
- a $\tilde{\mathbf{S}}^i$orthogonal decomposition $Tr\mathbf{V}_{|\Gamma_i} = \mathbf{V}_i \oplus \mathbf{Z}_i.$

Neumann-Neumann : **additive Schwarz** (solving S on V)

i) $\mathbf{V}_0 = \sum_{i=1}^N \mathbf{D}^i\mathbf{Z}_i \subset \mathbf{V}$, (scalar product $\tilde{\mathbf{S}}$), coarse space

ii) $\mathbf{V}_i$ (scalar product $\mathbf{B}_i = \tilde{\mathbf{S}}^i$) local spaces

iii) $I_i = (I - \mathbf{P})\mathbf{D}^i$ $\mathbf{P}$ the $\tilde{\mathbf{S}}$ orth projection of $\mathbf{V} \to \mathbf{V}_0$. (extensions)

Balanced Domain Decomposition Method

Particular case of the additive Schwarz method applied to interface problem

$$(\sum_i \mathbf{R}^{i^t}\mathbf{S}^i\mathbf{R}^i)\bar{X} = \bar{F}$$

Define:

- the space of global interface values $\mathbf{V} = \{\bar{v} = Tr\ v|_\Gamma, v \in \mathbf{H}(\Omega)\}$
- a partition of unity $\mathbf{D}^i : Tr\mathbf{V}_{|\Gamma_i} \rightarrow \mathbf{V}$
- an approximate local operator $\tilde{\mathbf{S}}^i$ st $\tilde{\mathbf{S}} = \sum \mathbf{R}^{i^t}\tilde{\mathbf{S}}^i\mathbf{R}^i$.
- a $\tilde{\mathbf{S}}^i$orthogonal decomposition $Tr\mathbf{V}_{|\Gamma_i} = \mathbf{V}_i \oplus \mathbf{Z}_i$.

Neumann-Neumann : **additive Schwarz** (solving S on V)

i) $\mathbf{V}_0 = \sum_{i=1}^N \mathbf{D}^i\mathbf{Z}_i \subset \mathbf{V}$, (scalar product $\tilde{\mathbf{S}}$), coarse space

ii) $\mathbf{V}_i$ (scalar product $\mathbf{B}_i = \tilde{\mathbf{S}}^i$) local spaces

iii) $I_i = (I - \mathbf{P})\mathbf{D}^i$ $\mathbf{P}$ the $\tilde{\mathbf{S}}$ orth projection of $\mathbf{V} \rightarrow \mathbf{V}_0$. (extensions)

The BDD preconditioner

$$\mathbf{M}^{-1}\mathbf{S}\bar{u} = \mathbf{P}\bar{u} + (\mathbf{I}-\mathbf{P})(\sum_i \mathbf{D}^i \tilde{\mathbf{S}}_i^{-1} \mathbf{D}^{i^t})(\mathbf{I}-\mathbf{P})^t \mathbf{S}\bar{u}$$

Remarks :

- coarse space contains **local singularities** (rigid bodies....)
- standard N-N : S exact Schur interface operator $(\mathbf{I}-\mathbf{P})^t\mathbf{S}\bar{u} = \mathbf{S}\bar{u}$
- interest to use an **approximate local operator** easy to generalize the method

The BDD preconditioner

$$\mathbf{M}^{-1}\mathbf{S}\bar{u} = \mathbf{P}\bar{u} + (\mathbf{I}-\mathbf{P})(\sum_i \mathbf{D}^i \tilde{\mathbf{S}}_i^{-1} \mathbf{D}^{i^t})(\mathbf{I}-\mathbf{P})^t \mathbf{S}\bar{u}$$

Remarks :

- coarse space contains **local singularities** (rigid bodies....)
- standard N-N : S exact Schur interface operator $(\mathbf{I}-\mathbf{P})^t\mathbf{S}\bar{u} = \mathbf{S}\bar{u}$
- interest to use an **approximate local operator** easy to generalize the method

Nonlinear elasticity

☑ use a Newton Algorithm
☑ use domain decomposition for each linearized problem
☑ construct the preconditioner once for the first linearised problem and reuse it

Time dependent problems

☑ use a Newmark or Euler time discretisation
☑ solve by domain decomposition at each time step
☑ construct the preconditioner using the rigid bodies of the linearized stiffness

Time dependent problem (cont)

After discretization in space, at each time step solve the **non-linear** problem

$$\frac{\rho M}{\Delta t^2} u + \mathcal{G}(u) = rhs$$

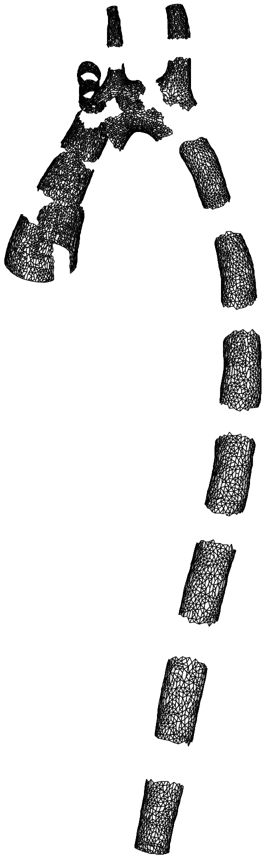
Use **Newton** algorithm, at each iteration solve

$$(\frac{\rho M}{\Delta t^2} + K^n) u = rhs$$

Remarks :

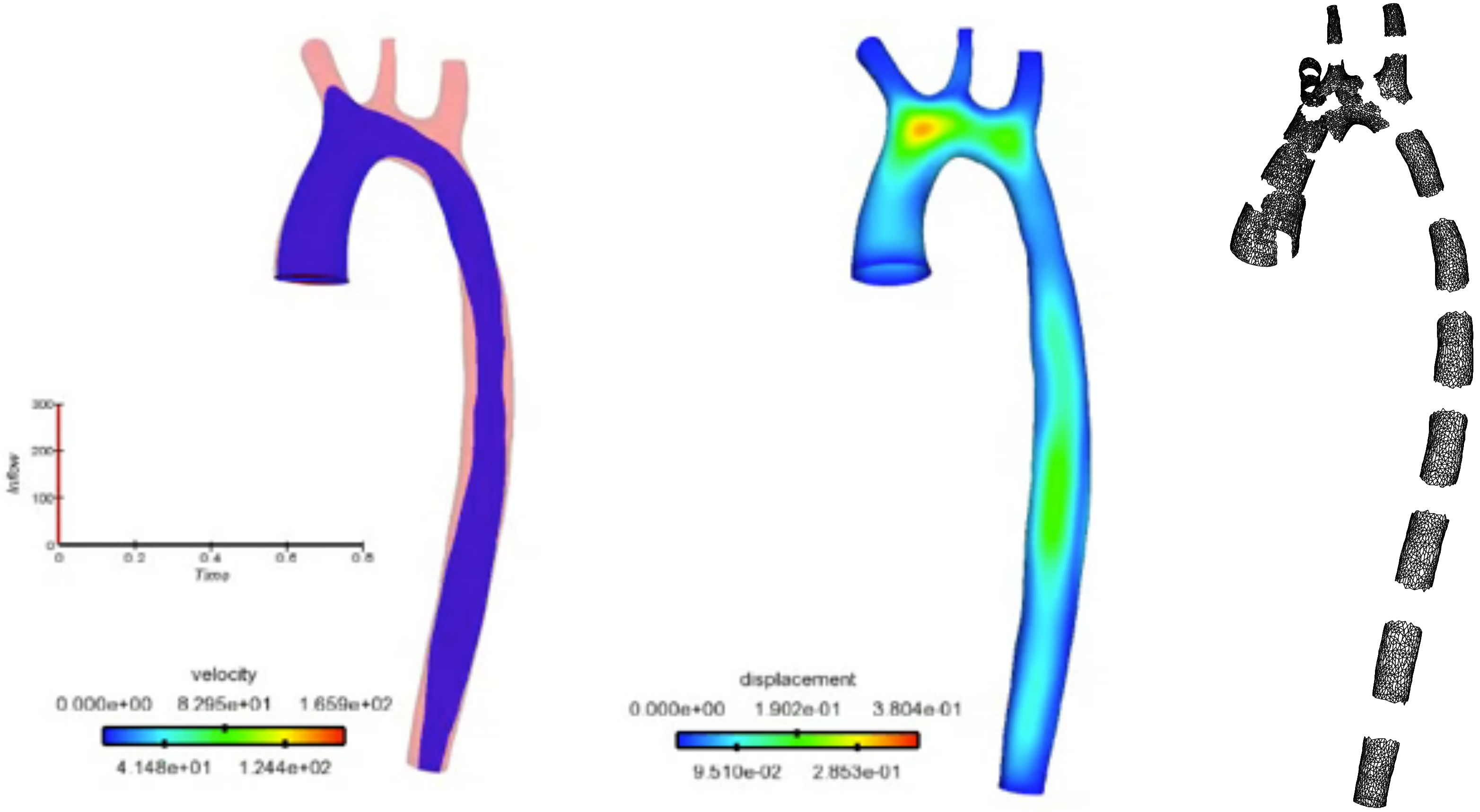
- the presence of the **mass-matrix** regularize the linear system
- thus the size of the **coarse space** based on **rigid bodies** is **zero**
- more **robust** approach **coarse space** based on the **stiffness matrix** only

Coarctated Aorta Blood Flow



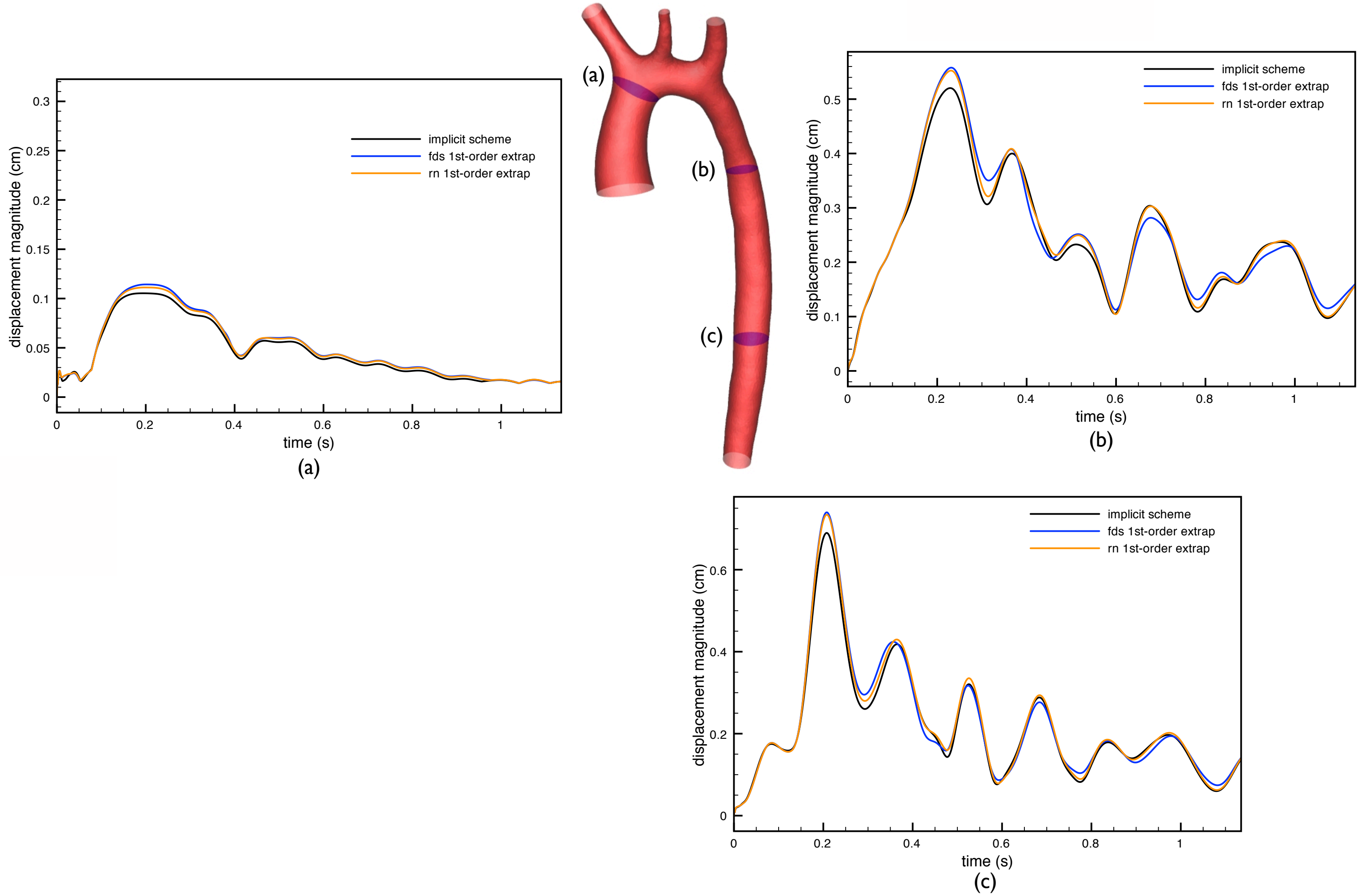
Simulation by M. Landajuela

Coarctated Aorta Blood Flow



Simulation by M. Landajuela

Accuracy: Explicit *vs.* Implicit



Simulation by M. Landajuela

Partitioned Solution Of Implicit Coupling

- Explicit Robin-Neumann coupling:

$$\begin{cases} \boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)\boldsymbol{n} + \dfrac{\rho^{\mathrm{s}}\epsilon}{\tau}\boldsymbol{u}^n = \dfrac{\rho^{\mathrm{s}}\epsilon}{\tau}\left(\dot{\boldsymbol{d}}^{n-1} + \tau\partial_\tau\dot{\boldsymbol{d}}^{\star}\right) + \boldsymbol{\sigma}(\boldsymbol{u}^{\star}, p^{\star})\boldsymbol{n} \quad \text{on} \quad \Sigma \\ \rho^{\mathrm{s}}\epsilon\partial_\tau\dot{\boldsymbol{d}}^{n} + \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^n = -\boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)\boldsymbol{n} \quad \text{on} \quad \Sigma \end{cases}$$

Partitioned Solution Of Implicit Coupling

- Explicit Robin-Neumann coupling:

$$\begin{cases} \boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)\boldsymbol{n} + \dfrac{\rho^{\mathrm{s}}\epsilon}{\tau}\boldsymbol{u}^n = \dfrac{\rho^{\mathrm{s}}\epsilon}{\tau}\left(\dot{\boldsymbol{d}}^{n-1} + \tau\partial_\tau\dot{\boldsymbol{d}}^{\star}\right) + \boldsymbol{\sigma}(\boldsymbol{u}^\star, p^\star)\boldsymbol{n} \quad \text{on} \quad \Sigma \\ \rho^{\mathrm{s}}\epsilon\partial_\tau\dot{\boldsymbol{d}}^n + \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^n = -\boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)\boldsymbol{n} \quad \text{on} \quad \Sigma \end{cases}$$

- Single iteration of a Robin-Neumann implicit solution algorithm *(Badia, Nobile, Vergara '08)*:

$$\begin{cases} \boldsymbol{\sigma}(\boldsymbol{u}_k, p_k)\boldsymbol{n} + \alpha\boldsymbol{u}_k = \alpha\dot{\boldsymbol{d}}_{k-1} + \boldsymbol{\sigma}(\boldsymbol{u}_{k-1}, p_{k-1})\boldsymbol{n} \quad \text{on} \quad \Sigma \\ \rho^{\mathrm{s}}\epsilon\partial_\tau\dot{\boldsymbol{d}}_k + \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}_k = -\boldsymbol{\sigma}(\boldsymbol{u}_k, p_k)\boldsymbol{n} \quad \text{on} \quad \Sigma \end{cases}$$

Robin parameter

$$\alpha \overset{\text{def}}{=} \frac{\rho^{\mathrm{s}}\epsilon}{\tau}$$

Partitioned Solution Of Implicit Coupling

- Explicit Robin-Neumann coupling:

$$\begin{cases} \boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)\boldsymbol{n} + \dfrac{\rho^{\mathrm{s}}\epsilon}{\tau}\boldsymbol{u}^n = \dfrac{\rho^{\mathrm{s}}\epsilon}{\tau}\left(\dot{\boldsymbol{d}}^{n-1} + \tau\partial_\tau \dot{\boldsymbol{d}}^{\star}\right) + \boldsymbol{\sigma}(\boldsymbol{u}^{\star}, p^{\star})\boldsymbol{n} \quad \text{on} \quad \Sigma \\ \rho^{\mathrm{s}}\epsilon\partial_\tau \dot{\boldsymbol{d}}^n + \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^n = -\boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)\boldsymbol{n} \quad \text{on} \quad \Sigma \end{cases}$$

- Single iteration of a Robin-Neumann implicit solution algorithm *(Badia, Nobile, Vergara '08)*:

$$\begin{cases} \boldsymbol{\sigma}(\boldsymbol{u}_k, p_k)\boldsymbol{n} + \alpha\boldsymbol{u}_k = \alpha\dot{\boldsymbol{d}}_{k-1} + \boldsymbol{\sigma}(\boldsymbol{u}_{k-1}, p_{k-1})\boldsymbol{n} \quad \text{on} \quad \Sigma \\ \rho^{\mathrm{s}}\epsilon\partial_\tau \dot{\boldsymbol{d}}_k + \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}_k = -\boldsymbol{\sigma}(\boldsymbol{u}_k, p_k)\boldsymbol{n} \quad \text{on} \quad \Sigma \end{cases}$$

Robin parameter

$$\alpha \overset{\text{def}}{=} \frac{\rho^{\mathrm{s}}\epsilon}{\tau}$$

- Energy norm error at iteration *k*:

$$e_k \overset{\text{def}}{=} \rho^{\mathrm{f}}\|\boldsymbol{u}_k - \boldsymbol{u}^n_{\mathrm{imp}}\|^2_{0,\Omega^{\mathrm{f}}} + \rho^{\mathrm{s}}\epsilon\|\dot{\boldsymbol{d}}_k - \dot{\boldsymbol{d}}^n_{\mathrm{imp}}\|^2_{0,\Sigma} + \|\boldsymbol{d}_k - \boldsymbol{d}^n_{\mathrm{imp}}\|^2_{\mathrm{e}}$$

Partitioned Solution Of Implicit Coupling

- Explicit Robin-Neumann coupling:

$$\begin{cases} \boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)\boldsymbol{n} + \dfrac{\rho^{\mathrm{s}}\epsilon}{\tau}\boldsymbol{u}^n = \dfrac{\rho^{\mathrm{s}}\epsilon}{\tau}\left(\dot{\boldsymbol{d}}^{n-1} + \tau\partial_\tau\dot{\boldsymbol{d}}^{\star}\right) + \boldsymbol{\sigma}(\boldsymbol{u}^{\star}, p^{\star})\boldsymbol{n} \quad \text{on} \quad \Sigma \\ \rho^{\mathrm{s}}\epsilon\partial_\tau\dot{\boldsymbol{d}}^n + \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}^n = -\boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)\boldsymbol{n} \quad \text{on} \quad \Sigma \end{cases}$$

- Single iteration of a Robin-Neumann implicit solution algorithm *(Badia, Nobile, Vergara '08)*:

$$\begin{cases} \boldsymbol{\sigma}(\boldsymbol{u}_k, p_k)\boldsymbol{n} + \alpha\boldsymbol{u}_k = \alpha\dot{\boldsymbol{d}}_{k-1} + \boldsymbol{\sigma}(\boldsymbol{u}_{k-1}, p_{k-1})\boldsymbol{n} \quad \text{on} \quad \Sigma \\ \rho^{\mathrm{s}}\epsilon\partial_\tau\dot{\boldsymbol{d}}_k + \boldsymbol{L}^{\mathrm{e}}\boldsymbol{d}_k = -\boldsymbol{\sigma}(\boldsymbol{u}_k, p_k)\boldsymbol{n} \quad \text{on} \quad \Sigma \end{cases}$$

Robin parameter

$$\alpha \overset{\text{def}}{=} \frac{\rho^{\mathrm{s}}\epsilon}{\tau}$$

- Energy norm error at iteration k:

$$e_k \overset{\text{def}}{=} \rho^{\mathrm{f}}\|\boldsymbol{u}_k - \boldsymbol{u}^n_{\mathrm{imp}}\|^2_{0,\Omega^{\mathrm{f}}} + \rho^{\mathrm{s}}\epsilon\|\dot{\boldsymbol{d}}_k - \dot{\boldsymbol{d}}^n_{\mathrm{imp}}\|^2_{0,\Sigma} + \|\boldsymbol{d}_k - \boldsymbol{d}^n_{\mathrm{imp}}\|^2_{\mathrm{e}}$$

Proposition: *(Fernández, Mullaert, MV '13)*

$$\sum_{k=1}^{\infty} e_k \leq \tau\|\boldsymbol{d}_0 - \boldsymbol{d}^n_{\mathrm{imp}}\|^2_{\mathrm{e}} + \frac{\tau^2}{\rho^{\mathrm{s}}\epsilon}\|\boldsymbol{L}^{\mathrm{e}}(\boldsymbol{d}_0 - \boldsymbol{d}^n_{\mathrm{imp}})\|^2_{0,\Sigma}.$$

Concluding Remarks

☑ Domain Decomposition is a powerful tool to solve large multi-scale problems

- Heterogenous methods adapted to be design ***parallel scalable algorithms***
- Heterogenous methods well suited for ***FSI***

☑ Stable explicit coupling schemes based on a built-in Robin interface consistency

- Only solid-inertia needs to be implicitly coupled with the fluid
- Elastic, viscous (*and incompressible*) solid contributions treated explicitly

Concluding Remarks

☑ Domain Decomposition is a powerful tool to solve large multi-scale problems

- Heterogenous methods adapted to be design ***parallel scalable algorithms***
- Heterogenous methods well suited for ***FSI***

☑ Stable explicit coupling schemes based on a built-in Robin interface consistency

- Only solid-inertia needs to be implicitly coupled with the fluid
- Elastic, viscous (*and incompressible*) solid contributions treated explicitly

☑ Robin-Neumann schemes with interesting features:

- Stability (added-mass free)
- Optimally first-order accurate (coupling with thin-solid)
- Kinematic perturbations of implicit coupling (fundamental for the analysis)
- Single iteration of a strong coupling solution procedure
- Parameter free

Joyeux anniversaire, Alain!!!

Joyeux anniversaire, Alain!!!

Une retraite heureuse!

Joyeux anniversaire, Alain!!!

Joyeux anniversaire, Alain!!!

Le travail c'est la santé

Joyeux anniversaire, Alain!!!

Le travail c'est la santé

Bonne chance dans ta nouvelle vie!