

Fluid-Structure Interaction and Multi-Body Contact.

Application to Biological Flows.

Matteo Astorino

Project REO

INRIA Paris-Rocquencourt

France

In collaboration with

J.F. Gerbeau, C. Grandmont, O. Pantz and K.F. Traoré



INSTITUT NATIONAL
DE RECHERCHE
EN INFORMATIQUE
ET EN AUTOMATIQUE



centre de recherche **PARIS - ROCQUENCOURT**



Motivations

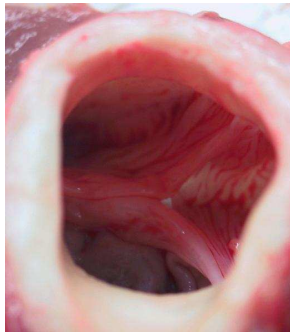
Cardiovascular Diseases

- Aneurysms
- Valvular regurgitation
- ...

Motivations

Cardiovascular Diseases

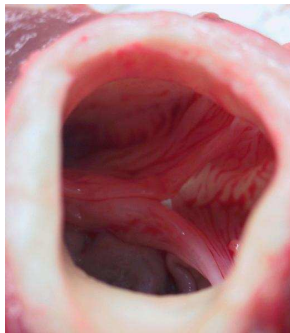
- Aneurysms
- Valvular regurgitation
- ...



Motivations

Cardiovascular Diseases

- Aneurysms
- Valvular regurgitation
- ...



Mathematical problems

- **Fluid-Structure Interaction**
blood, vessels and leaflets
- **Contacts** among leaflets

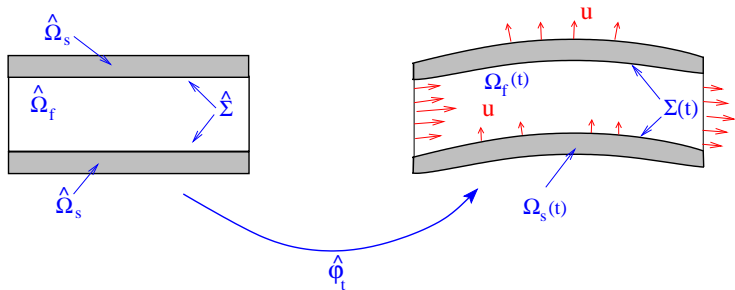
Outline

- 1 Blood-Vessel Wall Interaction
 - Arbitrary Lagrangian-Eulerian
 - FSI Semi-Implicit Projection algorithm
- 2 Blood-Leaflets Interactions
 - Fictitious Domain
- 3 Contact
 - Solid-wall contact
 - Multi-body contact
- 4 Conclusions

Outline

- 1 **Blood-Vessel Wall Interaction**
 - Arbitrary Lagrangian-Eulerian
 - FSI Semi-Implicit Projection algorithm
- 2 **Blood-Leaflets Interactions**
 - Fictitious Domain
- 3 **Contact**
 - Solid-wall contact
 - Multi-body contact
- 4 **Conclusions**

Blood-Vessel Wall Interaction



Requirements

- moving fluid domain surrounded by the structure domain
- non-linear structure models
- possibly thin structures

Arbitrary Lagrangian-Eulerian (ALE) formulation

Arbitrary Lagrangian-Eulerian (ALE) formulation

- Fluid problem:

$$\begin{aligned}\rho_f \frac{\partial \mathbf{u}_f}{\partial t} \Big|_{\hat{\mathbf{x}}} + \rho_f (\mathbf{u}_f - \mathbf{w}) \cdot \nabla \mathbf{u}_f - \operatorname{div} \boldsymbol{\sigma}_f(\mathbf{u}_f, p) &= 0 \quad \text{in } \Omega_f(t) \\ \operatorname{div} \mathbf{u}_f &= 0 \quad \text{in } \Omega_f(t)\end{aligned}$$

Arbitrary Lagrangian-Eulerian (ALE) formulation

- Fluid problem:

$$\begin{aligned} \rho_f \frac{\partial \mathbf{u}_f}{\partial t} \Big|_{\hat{\mathbf{x}}} + \rho_f (\mathbf{u}_f - \mathbf{w}) \cdot \nabla \mathbf{u}_f - \operatorname{div} \boldsymbol{\sigma}_f(\mathbf{u}_f, p) &= 0 \quad \text{in } \Omega_f(t) \\ \operatorname{div} \mathbf{u}_f &= 0 \quad \text{in } \Omega_f(t) \end{aligned}$$

- Fluid domain velocity:

$$\hat{\boldsymbol{\eta}}_f = \operatorname{Ext}(\hat{\boldsymbol{\eta}}_s |_{\hat{\Sigma}}), \quad \hat{\mathbf{w}} = \frac{\partial \hat{\boldsymbol{\eta}}_f}{\partial t} \quad \text{in } \Omega_f(t)$$

Arbitrary Lagrangian-Eulerian (ALE) formulation

- Fluid problem:

$$\begin{aligned} \rho_f \frac{\partial \mathbf{u}_f}{\partial t} \Big|_{\hat{\mathbf{x}}} + \rho_f (\mathbf{u}_f - \mathbf{w}) \cdot \nabla \mathbf{u}_f - \operatorname{div} \boldsymbol{\sigma}_f(\mathbf{u}_f, p) &= 0 \quad \text{in } \Omega_f(t) \\ \operatorname{div} \mathbf{u}_f &= 0 \quad \text{in } \Omega_f(t) \end{aligned}$$

- Fluid domain velocity:

$$\hat{\boldsymbol{\eta}}_f = \operatorname{Ext}(\hat{\boldsymbol{\eta}}_s |_{\hat{\Sigma}}), \quad \hat{\mathbf{w}} = \frac{\partial \hat{\boldsymbol{\eta}}_f}{\partial t} \quad \text{in } \Omega_f(t)$$

- Structure problem:

$$J \hat{\rho}_s \frac{\partial^2 \hat{\boldsymbol{\eta}}_s}{\partial t^2} - \operatorname{div}_{\hat{\mathbf{x}}} \hat{\boldsymbol{\Pi}}_s = 0 \quad \text{in } \hat{\Omega}_s$$

Arbitrary Lagrangian-Eulerian (ALE) formulation

- Fluid problem:

$$\begin{aligned} \rho_f \frac{\partial \mathbf{u}_f}{\partial t} \Big|_{\hat{\mathbf{x}}} + \rho_f (\mathbf{u}_f - \mathbf{w}) \cdot \nabla \mathbf{u}_f - \operatorname{div} \boldsymbol{\sigma}_f(\mathbf{u}_f, p) &= 0 \quad \text{in } \Omega_f(t) \\ \operatorname{div} \mathbf{u}_f &= 0 \quad \text{in } \Omega_f(t) \end{aligned}$$

- Fluid domain velocity:

$$\hat{\boldsymbol{\eta}}_f = \operatorname{Ext}(\hat{\boldsymbol{\eta}}_s|_{\hat{\Sigma}}), \quad \hat{\mathbf{w}} = \frac{\partial \hat{\boldsymbol{\eta}}_f}{\partial t} \quad \text{in } \Omega_f(t)$$

- Structure problem:

$$J \hat{\rho}_s \frac{\partial^2 \hat{\boldsymbol{\eta}}_s}{\partial t^2} - \operatorname{div}_{\hat{\mathbf{x}}} \hat{\boldsymbol{\Pi}}_s = 0 \quad \text{in } \hat{\Omega}_s$$

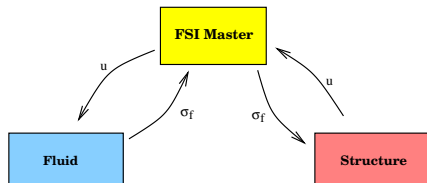
Coupling : transmission relations

$$\begin{cases} \mathbf{u}_f(\mathbf{x}, t) = \hat{\mathbf{u}}_s(\hat{\boldsymbol{\varphi}}_t^{-1}(\mathbf{x}), t) & \text{on } \Sigma(t) \\ \hat{\boldsymbol{\Pi}}_s \cdot \mathbf{n}_s + \hat{J} \hat{\boldsymbol{\sigma}}_f(\mathbf{u}_f, p) \hat{\mathbf{F}}^{-T} \hat{\boldsymbol{\eta}}_f = 0 & \text{on } \hat{\Sigma} \end{cases}$$

Coupling algorithms

Partitioned Scheme

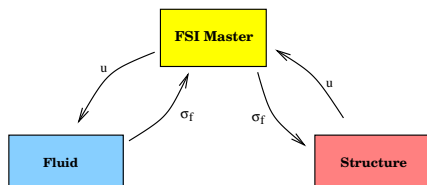
Use different solvers for the fluid and the solid(s).



Coupling algorithms

Partitioned Scheme

Use different solvers for the fluid and the solid(s).



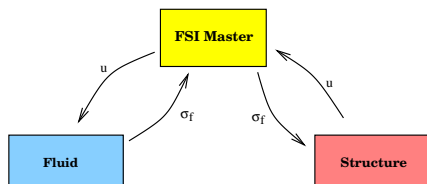
Coupling strategy

- **Explicit (or “Weak”)** coupling:

Coupling algorithms

Partitioned Scheme

Use different solvers for the fluid and the solid(s).



Coupling strategy

- **Explicit (or “Weak”) coupling:**

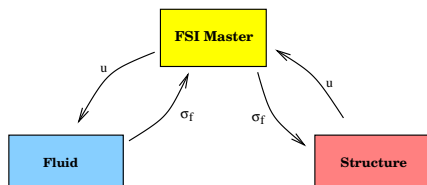
- ▶ Cheap...
- ▶ ...but produces a **spurious power at the interface:**

$$\int_{\Sigma^{n+1}} \sigma_f^{n+1} \cdot \mathbf{n} \left(\mathbf{u}_f^{n+1} - \frac{\hat{\eta}_s^n - \hat{\eta}_s^{n-1}}{\Delta t} \right)$$

Coupling algorithms

Partitioned Scheme

Use different solvers for the fluid and the solid(s).



Coupling strategy

- **Explicit (or “Weak”) coupling:**

- ▶ Cheap...
- ▶ ...but produces a **spurious power at the interface:**

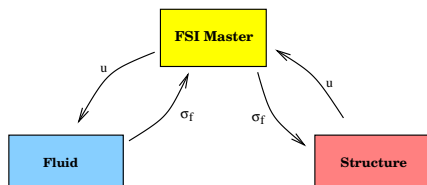
$$\int_{\Sigma^{n+1}} \sigma_f^{n+1} \cdot \mathbf{n} \left(\mathbf{u}_f^{n+1} - \frac{\hat{\eta}_s^n - \hat{\eta}_s^{n-1}}{\Delta t} \right)$$

- **Implicit (or “Strong”) coupling:**

Coupling algorithms

Partitioned Scheme

Use different solvers for the fluid and the solid(s).



Coupling strategy

- **Explicit (or “Weak”) coupling:**

- ▶ Cheap...
- ▶ ...but produces a **spurious power at the interface:**

$$\int_{\Sigma^{n+1}} \sigma_f^{n+1} \cdot \mathbf{n} \left(\mathbf{u}_f^{n+1} - \frac{\hat{\eta}_s^n - \hat{\eta}_s^{n-1}}{\Delta t} \right)$$

- **Implicit (or “Strong”) coupling:**

- ▶ Much more expensive;
- ▶ Conservation of the energy at the interface;

Implicit Coupling - Heterogeneous Domain Decomposition

- Dirichlet-Neumann

- ▶ Accelerated fixed point *Le Tallec-Mouro (1999), Mok-Wall-Ramm (2001)*
- ▶ Newton *Tezduyar (2001), Fernández-Moubachir (2003)*
- ▶ Inexact-Newton *Matthies-Steindorf (2003) Gerbeau-Vidrascu (2003)*
Mischler-van Brummelen-de Borst (2005)

- Neumann-Neumann

- ▶ *Deparis-Discacciati-Quarteroni (2005)*

- Robin-Neumann

- ▶ *Badia-Nobile-Vergara (2007)*

Implicit Coupling - Heterogeneous Domain Decomposition

- Dirichlet-Neumann
 - ▶ Accelerated fixed point *Le Tallec-Mouro (1999), Mok-Wall-Ramm (2001)*
 - ▶ Newton *Tezduyar (2001), Fernández-Moubachir (2003)*
 - ▶ Inexact-Newton *Matthies-Steindorf (2003) Gerbeau-Vidrascu (2003) Mischler-van Brummelen-de Borst (2005)*
- Neumann-Neumann
 - ▶ *Deparis-Discacciati-Quarteroni (2005)*
- Robin-Neumann
 - ▶ *Badia-Nobile-Vergara (2007)*

Added-Mass Effect

The explicit coupling is unconditionally unstable for application with **strong added-mass effect** (for example when $\rho_s \approx \rho_f$). (*Causin-Gerbeau-Nobile, 2004, Förster-Wall-Ramm, 2006*)

Implicit Coupling - Heterogeneous Domain Decomposition

- Dirichlet-Neumann
 - ▶ Accelerated fixed point *Le Tallec-Mouro (1999), Mok-Wall-Ramm (2001)*
 - ▶ Newton *Tezduyar (2001), Fernández-Moubachir (2003)*
 - ▶ Inexact-Newton *Matthies-Steindorf (2003) Gerbeau-Vidrascu (2003) Mischler-van Brummelen-de Borst (2005)*
- Neumann-Neumann
 - ▶ *Deparis-Discacciati-Quarteroni (2005)*
- Robin-Neumann
 - ▶ *Badia-Nobile-Vergara (2007)*

Added-Mass Effect

The explicit coupling is unconditionally unstable for application with **strong added-mass effect** (for example when $\rho_s \approx \rho_f$). (*Causin-Gerbeau-Nobile, 2004, Förster-Wall-Ramm, 2006*)

Perfect 2D fluid

$$\begin{aligned}\rho_f \partial_t \mathbf{u}_f + \nabla p &= 0, & \text{in } \Omega_f, \\ \operatorname{div} \mathbf{u}_f &= 0, & \text{in } \Omega_f, \\ \mathbf{u}_f \cdot \mathbf{n} &= \partial_t \hat{\boldsymbol{\eta}}_s, & \text{on } \Sigma\end{aligned}$$

Generalized string model

$$\rho_s \partial_{tt} \hat{\boldsymbol{\eta}}_s + \mathbf{a} \hat{\boldsymbol{\eta}}_s - b \partial_{xx} \hat{\boldsymbol{\eta}}_s = \mathbf{p} \quad \text{on } \Sigma$$

Outline

- 1 **Blood-Vessel Wall Interaction**
 - Arbitrary Lagrangian-Eulerian
 - **FSI Semi-Implicit Projection algorithm**

- 2 **Blood-Leaflets Interactions**
 - Fictitious Domain

- 3 **Contact**
 - Solid-wall contact
 - Multi-body contact

- 4 **Conclusions**

The FSI Semi-Implicit Projection algorithm

(M.A. Fernández, J.F. Gerbeau and C. Grandmont, 2006)

Projection method: Chorin-Temam (1969)

Step 1: advection-diffusion

$$\rho_f \frac{\tilde{\mathbf{u}}^{n+1} - \mathbf{u}^n}{\delta t} + \rho_f \tilde{\mathbf{u}}^n \cdot \nabla \tilde{\mathbf{u}}^{n+1} - \mu \Delta \tilde{\mathbf{u}}^{n+1} = 0$$

Step 2: projection

$$\left\{ \begin{array}{l} \rho_f \frac{\mathbf{u}^{n+1} - \tilde{\mathbf{u}}^{n+1}}{\delta t} + \nabla p^{n+1} = 0 \\ \operatorname{div} \mathbf{u}^{n+1} = 0 \end{array} \right.$$

The FSI Semi-Implicit Projection algorithm

(M.A. Fernández, J.F. Gerbeau and C. Grandmont, 2006)

Projection method: Chorin-Temam (1969)

Step 1: advection-diffusion

$$\rho_f \frac{\tilde{\mathbf{u}}^{n+1} - \mathbf{u}^n}{\delta t} + \rho_f \tilde{\mathbf{u}}^n \cdot \nabla \tilde{\mathbf{u}}^{n+1} - \mu \Delta \tilde{\mathbf{u}}^{n+1} = 0$$

Step 2: projection

$$\begin{cases} \rho_f \frac{\mathbf{u}^{n+1} - \tilde{\mathbf{u}}^{n+1}}{\delta t} + \nabla p^{n+1} = 0 \\ \operatorname{div} \mathbf{u}^{n+1} = 0 \end{cases}$$

Basic idea of the scheme

Step 1: advection-diffusion-ALE

← explicit coupling for **efficiency**

The FSI Semi-Implicit Projection algorithm

(M.A. Fernández, J.F. Gerbeau and C. Grandmont, 2006)

Projection method: Chorin-Temam (1969)

Step 1: advection-diffusion

$$\rho_f \frac{\tilde{\mathbf{u}}^{n+1} - \mathbf{u}^n}{\delta t} + \rho_f \tilde{\mathbf{u}}^n \cdot \nabla \tilde{\mathbf{u}}^{n+1} - \mu \Delta \tilde{\mathbf{u}}^{n+1} = 0$$

Step 2: projection

$$\begin{cases} \rho_f \frac{\mathbf{u}^{n+1} - \tilde{\mathbf{u}}^{n+1}}{\delta t} + \nabla p^{n+1} = 0 \\ \operatorname{div} \mathbf{u}^{n+1} = 0 \end{cases}$$

Basic idea of the scheme

Step 1: advection-diffusion-ALE

← explicit coupling for **efficiency**

Step 2: projection

← implicit coupling for **stability**

The FSI Semi-Implicit Projection algorithm

Let Ω^n , \mathbf{u}^n , p^n , $\hat{\eta}^n$:

Step 1: Domain extrapolation

$$\tilde{\eta}_{\Sigma}^{n+1} = \hat{\eta}^n + \delta t \left(\frac{3}{2} \hat{\mathbf{u}}_S^n - \frac{1}{2} \hat{\mathbf{u}}_S^{n-1} \right)$$

$$\mathbf{w}_{\Sigma}^{n+1} = \frac{\tilde{\eta}_{\Sigma}^{n+1} - \hat{\eta}^n|_{\Sigma}}{\delta t}$$

The FSI Semi-Implicit Projection algorithm

Let Ω^n , \mathbf{u}^n , p^n , $\hat{\eta}^n$:

Step 1: Domain extrapolation

$$\tilde{\eta}_{\Sigma}^{n+1} = \hat{\eta}^n + \delta t \left(\frac{3}{2} \hat{\mathbf{u}}_S^n - \frac{1}{2} \hat{\mathbf{u}}_S^{n-1} \right)$$

$$\mathbf{w}_{\Sigma}^{n+1} = \frac{\tilde{\eta}_{\Sigma}^{n+1} - \hat{\eta}^n|_{\Sigma}}{\delta t}$$

Step 2: ALE-Advection-diffusion (explicit coupling)

$$\left\{ \begin{array}{ll} \rho_f \frac{\tilde{\mathbf{u}}_f^{n+1} - \mathbf{u}_f^n}{\delta t} \Big|_{\hat{x}} + \rho_f (\tilde{\mathbf{u}}_f^n - \mathbf{w}^{n+1}) \cdot \nabla \tilde{\mathbf{u}}_f^{n+1} - \mu \Delta \tilde{\mathbf{u}}_f^{n+1} = 0 & \text{in } \Omega_f^{n+1} \\ \tilde{\mathbf{u}}_f^{n+1} = \mathbf{w}_{\Sigma}^{n+1} & \text{on } \Sigma^{n+1} \end{array} \right.$$

$$\mathbf{w}^{n+1} = Tr^{-1}(\mathbf{w}_{\Sigma}^{n+1})$$

Remark: The geometric non-linearity, the convective and the viscous terms are treated explicitly.

Step 3: Projection (implicit coupling)

$$\left\{ \begin{array}{l} \rho_f \frac{\mathbf{u}_f^{n+1} - \tilde{\mathbf{u}}_f^{n+1}}{\delta t} + \nabla p^{n+1} = 0 \quad \text{in } \Omega_f^{n+1} \\ \operatorname{div} \mathbf{u}_f^{n+1} = 0 \quad \text{in } \Omega_f^{n+1} \\ \mathbf{u}_f^{n+1} \cdot \mathbf{n} = \frac{\hat{\eta}^{n+1} - \hat{\eta}^n}{\delta t} \cdot \mathbf{n} \quad \text{on } \Sigma^{n+1} \end{array} \right.$$

$$\left\{ \begin{array}{l} \rho_s \frac{\hat{\mathbf{u}}_S^{n+1} - \hat{\mathbf{u}}_S^n}{\delta t} - \hat{\operatorname{div}} \left(\frac{\hat{\mathbf{\Pi}}^{n+1} + \hat{\mathbf{\Pi}}^n}{2} \right) = 0 \quad \text{in } \hat{\Omega}_s \\ \frac{\hat{\eta}^{n+1} - \hat{\eta}^n}{\delta t} = \frac{\hat{\mathbf{u}}_S^{n+1} + \hat{\mathbf{u}}_S^n}{2} \quad \text{in } \hat{\Omega}_s \\ \hat{\mathbf{\Pi}}^{n+1} \cdot \hat{\mathbf{n}}_s = p^{n+1} \mathbf{n}_s - \mu (\nabla \tilde{\mathbf{u}}_f^{n+1} + (\nabla \tilde{\mathbf{u}}_f^{n+1})^T) \cdot \mathbf{n}_s \end{array} \right.$$

Step 3: Projection (implicit coupling)

$$\left\{ \begin{array}{l} \rho_f \frac{\mathbf{u}_f^{n+1} - \tilde{\mathbf{u}}_f^{n+1}}{\delta t} + \nabla p^{n+1} = 0 \quad \text{in } \Omega_f^{n+1} \\ \operatorname{div} \mathbf{u}_f^{n+1} = 0 \quad \text{in } \Omega_f^{n+1} \\ \mathbf{u}_f^{n+1} \cdot \mathbf{n} = \frac{\hat{\eta}^{n+1} - \hat{\eta}^n}{\delta t} \cdot \mathbf{n} \quad \text{on } \Sigma^{n+1} \end{array} \right.$$

$$\left\{ \begin{array}{l} \rho_s \frac{\hat{\mathbf{u}}_S^{n+1} - \hat{\mathbf{u}}_S^n}{\delta t} - \hat{\operatorname{div}} \left(\frac{\hat{\mathbf{\Pi}}^{n+1} + \hat{\mathbf{\Pi}}^n}{2} \right) = 0 \quad \text{in } \hat{\Omega}_s \\ \frac{\hat{\eta}^{n+1} - \hat{\eta}^n}{\delta t} = \frac{\hat{\mathbf{u}}_S^{n+1} + \hat{\mathbf{u}}_S^n}{2} \quad \text{in } \hat{\Omega}_s \\ \hat{\mathbf{\Pi}}^{n+1} \cdot \hat{\mathbf{n}}_s = p^{n+1} \mathbf{n}_s - \mu (\nabla \tilde{\mathbf{u}}_f^{n+1} + (\nabla \tilde{\mathbf{u}}_f^{n+1})^T) \cdot \mathbf{n}_s \end{array} \right.$$

Remarks

- Fixed fluid domain in the projection step;
- Cheap inner iterations.
- Other “projection schemes” can be used, for example Algebraic Projection Methods, (*Badia-Quarteroni-Quaini, 2007*).

Numerical Comparison

Test Case (*M.A. Fernández, J.F. Gerbeau, C. Grandmont, 2006*)

Straight cylinder, 50 time steps, $\delta t = 2 \times 10^{-4}$ s.

COUPLING	ALGORITHM	Total CPU (dimensionless)
Implicit	Fixed point	approx. 50
	Fixed point + Aitken (2001)	24.86
	quasi-Newton (2003)	6.05
	Newton (2003)	4.77
Projection Semi-Implicit	Newton (2005)	1

Remark: The coupling is indeed “weak”

Spurious energy :

$$\int_{\hat{\Sigma}} 2\mu \mathbf{n} \cdot \mathbf{D}(\tilde{\mathbf{u}}_f^{n+1}) \cdot (\tilde{\mathbf{u}}_f^{n+1} - \mathbf{u}_s^{n+1}) - p \mathbf{Id} \cdot (\mathbf{u}_f^{n+1} - \mathbf{u}_s^{n+1})$$

Remark: The coupling is indeed “weak”

Spurious energy :

$$\int_{\hat{\Sigma}} 2\mu \mathbf{n} \cdot \mathbf{D}(\tilde{\mathbf{u}}_f^{n+1}) \cdot (\tilde{\mathbf{u}}_f^{n+1} - \mathbf{u}_s^{n+1})$$

How to control this spurious energy?

Remark: The coupling is indeed “weak”

Spurious energy :

$$\int_{\hat{\Sigma}} 2\mu \mathbf{n} \cdot \mathbf{D}(\tilde{\mathbf{u}}_f^{n+1}) \cdot (\tilde{\mathbf{u}}_f^{n+1} - \mathbf{u}_s^{n+1})$$

How to control this spurious energy?

Stability Condition (*M.A. Fernández, J.F. Gerbeau, C. Grandmont, 2006*)

The numerical scheme is stable in the energy norm under the following condition:

$$\rho_s \geq C \left(\rho_f h + 2 \frac{\mu \delta t}{h} \right)$$

(h : fluid discretization step, δt : time step)

Remark: The coupling is indeed “weak”

Spurious energy :

$$\int_{\hat{\Sigma}} 2\mu \mathbf{n} \cdot \mathbf{D}(\tilde{\mathbf{u}}_f^{n+1}) \cdot (\tilde{\mathbf{u}}_f^{n+1} - \mathbf{u}_s^{n+1})$$

How to control this spurious energy?

Stability Condition (*M.A. Fernández, J.F. Gerbeau, C. Grandmont, 2006*)

The numerical scheme is stable in the energy norm under the following condition:

$$\rho_s \geq C \left(\rho_f h + 2 \frac{\mu \delta t}{h} \right)$$

(h : fluid discretization step, δt : time step)

- The leap-frog scheme is used for the time discretization of structure equation.
- With the Newmark scheme the analysis is not trivial (work in progress *M.A. and C. Grandmont*).
- **The scheme proves to be stable in all practical situations .**

Convergence of the scheme

Convergence of the scheme

Theorem (M. A. and C. Grandmont, 2008)

Provided the sufficient stability conditions, the numerical scheme converges and the error (for velocity and displacements) is of order $O(\sqrt{\delta t}) + O(H^l) + O(h^k) + O(h^\sigma)$, where σ depends on the matching interface operator.

(H : structure discretization step, l and k : finite element order)

σ	3D-3D Coupling	3D-2D: 2° order op.	3D-2D: 4° order op
Interpolation Mortar	non optimal optimal $\forall k$	$k = 1$ optimal optimal $\forall k$	$k \leq 2$ optimal optimal $\forall k$

Convergence of the scheme

Theorem (M. A. and C. Grandmont, 2008)

Provided the sufficient stability conditions, the numerical scheme converges and the error (for velocity and displacements) is of order $O(\sqrt{\delta t}) + O(H^l) + O(h^k) + O(h^\sigma)$, where σ depends on the matching interface operator.

(H : structure discretization step, l and k : finite element order)

σ	3D-3D Coupling	3D-2D: 2° order op.	3D-2D: 4° order op
Interpolation Mortar	non optimal optimal $\forall k$	$k = 1$ optimal optimal $\forall k$	$k \leq 2$ optimal optimal $\forall k$

Accuracy in time

- The accuracy for the pressure in norm $\ell^\infty([0, T]; L^2(\Omega))$ for the standard Chorin-Temam method is $O(\sqrt{\delta t})$.

Convergence of the scheme

Theorem (M. A. and C. Grandmont, 2008)

Provided the sufficient stability conditions, the numerical scheme converges and the error (for velocity and displacements) is of order $O(\sqrt{\delta t}) + O(H^l) + O(h^k) + O(h^\sigma)$, where σ depends on the matching interface operator.

(H : structure discretization step, l and k : finite element order)

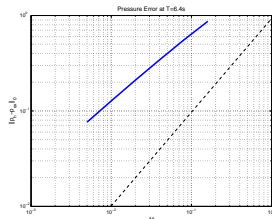
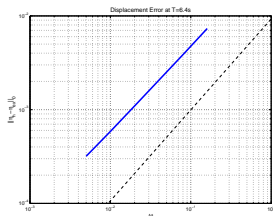
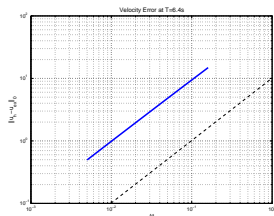
σ	3D-3D Coupling	3D-2D: 2° order op.	3D-2D: 4° order op
Interpolation Mortar	non optimal optimal $\forall k$	$k = 1$ optimal optimal $\forall k$	$k \leq 2$ optimal optimal $\forall k$

Accuracy in time

- The accuracy for the pressure in norm $\ell^\infty([0, T]; L^2(\Omega))$ for the standard Chorin-Temam method is $O(\sqrt{\delta t})$.
- In our convergence analysis the pressure terms affect the accuracy of velocity and displacements.

Convergence Result

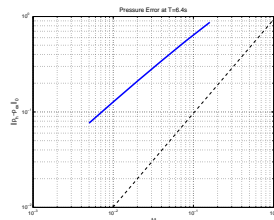
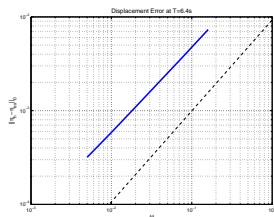
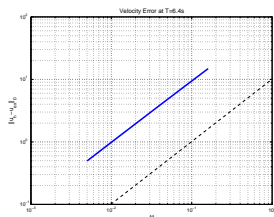
Numerically,



Linear convergence in time for velocity and displacements

Convergence Result

Numerically,



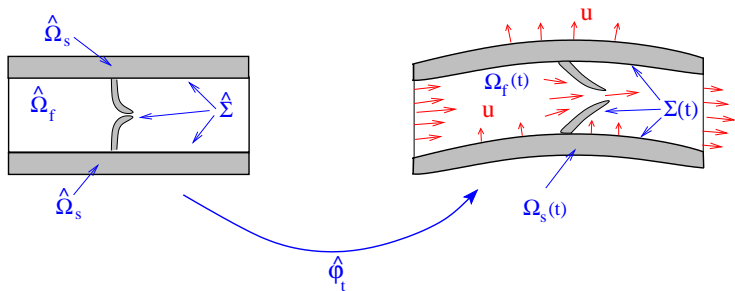
Linear convergence in time for velocity and displacements

We are currently investigating other possible directions in the demonstration procedure, in order to retrieve $O(\delta t)$ for velocity and displacements.

Outline

- 1 Blood-Vessel Wall Interaction
 - Arbitrary Lagrangian-Eulerian
 - FSI Semi-Implicit Projection algorithm
- 2 Blood-Leaflets Interactions
 - Fictitious Domain
- 3 Contact
 - Solid-wall contact
 - Multi-body contact
- 4 Conclusions

Blood-Leaflets Interactions



Requirements

- Very large displacement of the immersed structure
- Topological changes in the fluid domain

Fictitious Domain formulation *(Glowinski et al., 1994)*

Main Ideas

- Independent meshes for the fluid and the structure
- The coupling is obtained by enforcing the kinematic condition with Lagrange multipliers

Remark: Other methods could be used, for example: Immersed Boundary Method, *(D. M. McQueen and C.S. Peskin, 2000)*.

Fictitious Domain formulation *(Glowinski et al., 1994)*

Main Ideas

- Independent meshes for the fluid and the structure
- The coupling is obtained by enforcing the kinematic condition with Lagrange multipliers

Remark: Other methods could be used, for example: Immersed Boundary Method, *(D. M. McQueen and C.S. Peskin, 2000)*.

Formally

$$\left\{ \begin{array}{l} a_f(\mathbf{u}_f, \mathbf{v}_f) \\ \hat{a}_s(\hat{\mathbf{u}}_s, \hat{\mathbf{v}}_s) \end{array} \right. = \left\{ \begin{array}{l} \int_{\Omega_f} \mathbf{f}_f \cdot \mathbf{v}_f, \quad \forall \mathbf{v}_f \in X_f \\ \int_{\hat{\Omega}_s} \hat{\mathbf{f}}_s \cdot \hat{\mathbf{v}}_s, \quad \forall \hat{\mathbf{v}}_s \in \hat{X}_s \end{array} \right.$$

Fictitious Domain formulation *(Glowinski et al., 1994)*

Main Ideas

- Independent meshes for the fluid and the structure
- The coupling is obtained by enforcing the kinematic condition with Lagrange multipliers

Remark: Other methods could be used, for example: Immersed Boundary Method, *(D. M. McQueen and C.S. Peskin, 2000)*.

Formally

$$\left\{ \begin{array}{l} a_f(\mathbf{u}_f, \mathbf{v}_f) + \langle \boldsymbol{\lambda}, \mathbf{v}_f \rangle_\Sigma = \int_{\Omega_f} \mathbf{f}_f \cdot \mathbf{v}_f, \quad \forall \mathbf{v}_f \in X_f \\ \langle \boldsymbol{\mu}, \mathbf{u}_f \rangle_\Sigma - \langle \boldsymbol{\mu}, \mathbf{u}_s \rangle_\Sigma = 0 \quad \forall \boldsymbol{\mu} \in \Lambda \\ \hat{a}_s(\hat{\mathbf{u}}_s, \hat{\mathbf{v}}_s) - \langle \boldsymbol{\lambda}, \mathbf{v}_s \rangle_\Sigma = \int_{\hat{\Omega}_s} \hat{\mathbf{f}}_s \cdot \hat{\mathbf{v}}_s \quad \forall \hat{\mathbf{v}}_s \in \hat{X}_s \end{array} \right.$$

Fictitious Domain formulation *(Glowinski et al., 1994)*

Main Ideas

- Independent meshes for the fluid and the structure
- The coupling is obtained by enforcing the kinematic condition with Lagrange multipliers

Remark: Other methods could be used, for example: Immersed Boundary Method, *(D. M. McQueen and C.S. Peskin, 2000)*.

Formally

$$\left\{ \begin{array}{l} a_f(\mathbf{u}_f, \mathbf{v}_f) + \langle \boldsymbol{\lambda}, \mathbf{v}_f \rangle_\Sigma = \int_{\Omega_f} \mathbf{f}_f \cdot \mathbf{v}_f, \quad \forall \mathbf{v}_f \in X_f \\ \langle \boldsymbol{\mu}, \mathbf{u}_f \rangle_\Sigma - \langle \boldsymbol{\mu}, \mathbf{u}_s \rangle_\Sigma = 0 \quad \forall \boldsymbol{\mu} \in \Lambda \\ \hat{a}_s(\hat{\mathbf{u}}_s, \hat{\mathbf{v}}_s) - \langle \boldsymbol{\lambda}, \mathbf{v}_s \rangle_\Sigma = \int_{\hat{\Omega}_s} \hat{\mathbf{f}}_s \cdot \hat{\mathbf{v}}_s \quad \forall \hat{\mathbf{v}}_s \in \hat{X}_s \end{array} \right.$$

Load on the structure:

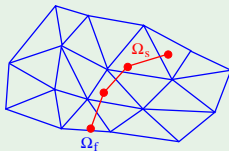
$$\langle \boldsymbol{\lambda}, \mathbf{v}_f \rangle_\Sigma = -\langle \boldsymbol{\sigma}_f \cdot \mathbf{n}_f, \mathbf{v}_f \rangle_\Sigma = \langle [\hat{\mathbf{J}} \hat{\boldsymbol{\sigma}}_f \hat{\mathbf{F}}^{-T} \hat{\mathbf{n}}_s], \hat{\mathbf{v}}_s \rangle_\Sigma = \langle \boldsymbol{\lambda}, \mathbf{v}_s \rangle_\Sigma$$

Finite element approximation

$$\begin{cases} A_f U_f + B_f^T \Lambda & = F_f \\ B_f U_f - B_s U_s & = 0 \\ A_s U_s - B_s^T \Lambda & = F_s \end{cases}$$

Finite element approximation

$$\begin{cases} A_f U_f + B_f^T \Lambda & = F_f \\ B_f U_f - B_s U_s & = 0 \\ A_s U_s - B_s^T \Lambda & = F_s \end{cases}$$



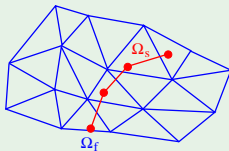
$$\Lambda_h^{n+1} = \left\{ \mu_h \text{ measure on } \Sigma^{n+1}, \mu_h = \sum_{i=1}^{N_\Sigma} \mu_i \delta(\mathbf{x}_i^{n+1}), \mu_i \in \mathbb{R}^3 \right\},$$

Remark

Other choices for μ_h are possible, for example L^2 functions. (*Baaijens, 2001. Baaijens-De Hart, 2003, van Loan-De Hart-Baaijens, 2004*)

Finite element approximation

$$\begin{cases} A_f U_f + B_f^T \Lambda & = F_f \\ B_f U_f - B_s U_s & = 0 \\ A_s U_s - B_s^T \Lambda & = F_s \end{cases}$$



$$\Lambda_h^{n+1} = \left\{ \mu_h \text{ measure on } \Sigma^{n+1}, \mu_h = \sum_{i=1}^{N_\Sigma} \mu_i \delta(\mathbf{x}_i^{n+1}), \mu_i \in \mathbb{R}^3 \right\},$$

With this choice:

$$\begin{cases} A_f U_f + K^T \Lambda & = F_f \\ K U_f - U_s & = 0 \\ A_s U_s - \Lambda & = F_s \end{cases}$$

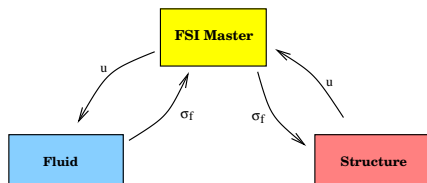
where K is the fluid-to-structure interpolation matrix.

Remark

Other choices for μ_h are possible, for example L^2 functions. (*Baaijens, 2001. Baaijens-De Hart, 2003, van Loan-De Hart-Baaijens, 2004*)

Partitioned scheme

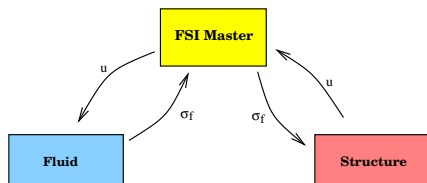
Use different solvers for the fluid and the solid(s)



(Dos Santos-Gerbeau-Bourgat, 2008)

Partitioned scheme

Use different solvers for the fluid and the solid(s)



- Step 1

$$\begin{cases} A_f U_f + K^T \Lambda = F_f \\ K U_f = U_s \end{cases}$$

- Step 2

$$A_s U_s = F_s + \Lambda$$

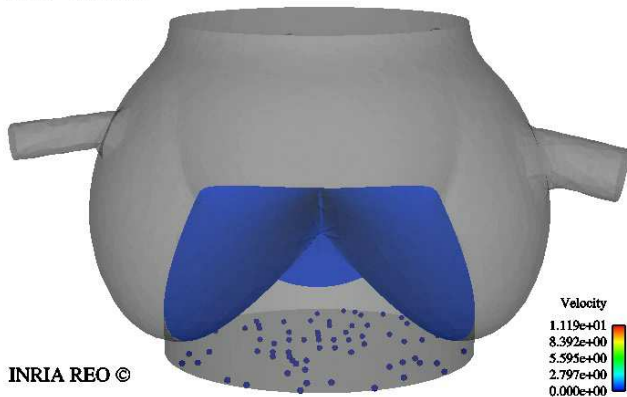
- FSI Coupling: Accelerated Fixed Point algorithm
(Wall-Mok-Ramm, 2001)

(Dos Santos-Gerbeau-Bourgat, 2008)

Remark

Others FSI algorithms can be used with this partitioned scheme.
(Work in progress for valves)

Time = 0.00000



Aortic valve

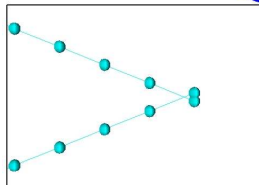
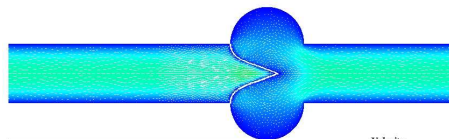
(M.A., J.F. Gerbeau, O. Pantz and K.F. Traoré, 2008)

Outline

- 1 Blood-Vessel Wall Interaction
 - Arbitrary Lagrangian-Eulerian
 - FSI Semi-Implicit Projection algorithm
- 2 Blood-Leaflets Interactions
 - Fictitious Domain
- 3 **Contact**
 - **Solid-wall contact**
 - Multi-body contact
- 4 Conclusions

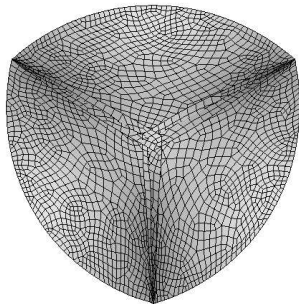
Preliminary remarks

- What happened if contact is not handled:



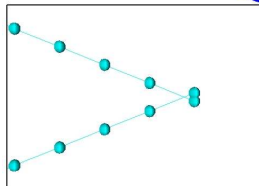
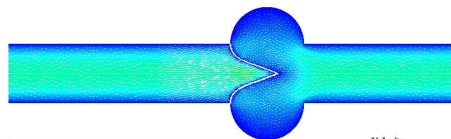
Velocity
1.600e+01
1.200e+01
8.000e+00
4.000e+00
0.000e+00

Time = 0.2000



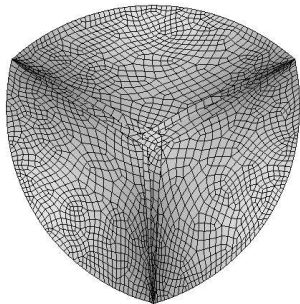
Preliminary remarks

- What happened if contact is not handled:



Velocity
1.600e+01
1.200e+01
8.000e+00
4.000e+00
0.000e+00

Time = 0.2000



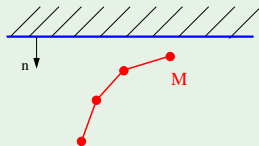
- Our approach for contact is basic:
 - ▶ no friction
 - ▶ no specific treatment for the fluid

A first example: solid-wall contact

Let M be a solid and \mathcal{T}_h a P_1 mesh of M :

$$X_h = \{\varphi_h \in C^0(M; \mathbb{R}^3), \varphi_h|_T \in P_1, \forall T \in \mathcal{T}_h\}.$$

Let J be the energy of the solid.



A first example: solid-wall contact

Let M be a solid and \mathcal{T}_h a P_1 mesh of M :

$$X_h = \{\varphi_h \in C^0(M; \mathbb{R}^3), \varphi_h|_T \in P_1, \forall T \in \mathcal{T}_h\}.$$

Let J be the energy of the solid.

The deformation $\varphi_h : M \rightarrow \mathbb{R}^3$ is solution to

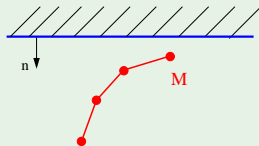
$$\inf_{\varphi_h \in \mathcal{U}_h} J(\varphi_h)$$

with

$$\mathcal{U}_h = \{\varphi_h \in X_h, F_{x_i}(\varphi_h) \leq 0, \forall x_i \in M\}$$

and

$$F_{x_i}(\varphi_h) = \varepsilon - \mathbf{n} \cdot \varphi(x_i) - c$$



A first example: solid-wall contact

Let M be a solid and \mathcal{T}_h a P_1 mesh of M :

$$X_h = \{\varphi_h \in C^0(M; \mathbb{R}^3), \varphi_h|_T \in P_1, \forall T \in \mathcal{T}_h\}.$$

Let J be the energy of the solid.

The deformation $\varphi_h : M \rightarrow \mathbb{R}^3$ is solution to

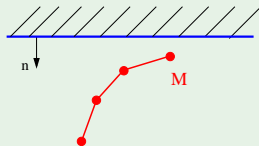
$$\inf_{\varphi_h \in \mathcal{U}_h} J(\varphi_h)$$

with

$$\mathcal{U}_h = \{\varphi_h \in X_h, F_{x_i}(\varphi_h) \leq 0, \forall x_i \in M\}$$

and

$$F_{x_i}(\varphi_h) = \varepsilon - \mathbf{n} \cdot \varphi(x_i) - c$$



Remark

The constraint is convex

Solid-wall contact

Requirement

- Our solid solvers are not supposed to manage contact

Solid-wall contact

Requirement

- Our solid solvers are not supposed to manage contact

Dual approach

- Dual energy of the problem:

$$G(\mu) = \inf_{\varphi \in X_h} \left[J(\varphi) + \sum_{i=1}^{N_\Sigma} \mu_i F_{x_i}(\varphi) \right]$$

- Solve

$$G(\lambda_c) = \max_{\mu_i \geq 0} G(\mu)$$

with a gradient method with projection.

- λ_c is the **contact pressure**.

Solid-wall contact

Algorithm (convex constraint)

(i) Initial guess: λ_c^0 .

Solid-wall contact

Algorithm (convex constraint)

- (i) Initial guess: λ_c^0 .
- (ii) Solve the structure problem:

$$\langle \mathbf{J}'(\varphi^k), \boldsymbol{\xi} \rangle = - \sum_{i=1}^{N_\Sigma} \lambda_{c,i}^k \langle \mathbf{F}'_{x_i}(\varphi^k), \boldsymbol{\xi} \rangle = \sum_{i=1}^{N_\Sigma} \lambda_{c,i}^k \mathbf{n} \cdot \boldsymbol{\xi}(x_i)$$

Solid-wall contact

Algorithm (convex constraint)

- (i) Initial guess: λ_c^0 .
- (ii) Solve the structure problem:

$$\langle \mathbf{J}'(\varphi^k), \boldsymbol{\xi} \rangle = - \sum_{i=1}^{N_\Sigma} \lambda_{c,i}^k \langle \mathbf{F}'_{x_i}(\varphi^k), \boldsymbol{\xi} \rangle = \sum_{i=1}^{N_\Sigma} \lambda_{c,i}^k \mathbf{n} \cdot \boldsymbol{\xi}(x_i)$$

- (iii) Gradient iteration with projection:

$$\begin{aligned} \lambda_{c,i}^{k+1} &= \mathbf{P}_{\mathbb{R}^+} \left(\lambda_{c,i}^k + \alpha^k \nabla G(\lambda_c^k)_i \right) \\ &= \mathbf{P}_{\mathbb{R}^+} \left(\lambda_{c,i}^k + \alpha^k F_{x_i}(\varphi^k) \right) \\ &= \mathbf{P}_{\mathbb{R}^+} \left(\lambda_{c,i}^k + \alpha^k (\varepsilon - \mathbf{n} \cdot \varphi^k(x_i) - c) \right) \end{aligned}$$

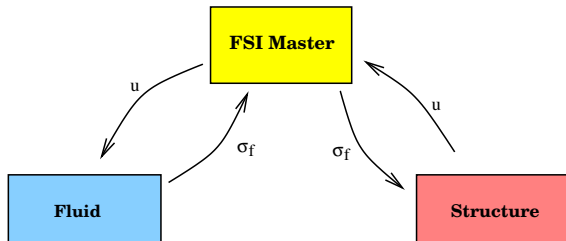
- (iv) Go to (ii) until convergence.

Contact problem

- This algorithm fits in our general coupling
- The same data as for FSI are exchanged

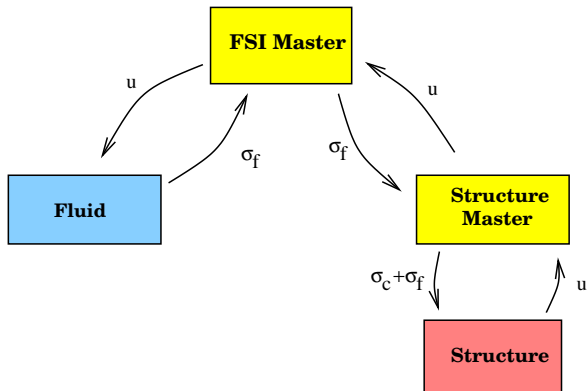
Contact problem

- This algorithm fits in our general coupling
- The same data as for FSI are exchanged



Contact problem

- This algorithm fits in our general coupling
- The same data as for FSI are exchanged



Outline

- 1 Blood-Vessel Wall Interaction
 - Arbitrary Lagrangian-Eulerian
 - FSI Semi-Implicit Projection algorithm
- 2 Blood-Leaflets Interactions
 - Fictitious Domain
- 3 **Contact**
 - Solid-wall contact
 - **Multi-body contact**
- 4 Conclusions

Multi-body contact

Let M be a **family** of solids $M = (M_1, M_2, \dots)$
and let \mathcal{T}_h be a P_1 finite element mesh of M :

$$X_h = \{\varphi_h \in C^0(M; \mathbb{R}^3), \varphi_h|_T \in P_1, \forall T \in \mathcal{T}_h\}$$

Let J be the energy of the solids.



Multi-body contact

Let M be a **family** of solids $M = (M_1, M_2, \dots)$
and let \mathcal{T}_h be a P_1 finite element mesh of M :

$$X_h = \{\varphi_h \in C^0(M; \mathbb{R}^3), \varphi_h|_T \in P_1, \forall T \in \mathcal{T}_h\}$$

Let J be the energy of the solids.

The deformation $\varphi_h : \cup_i M_i \rightarrow \mathbb{R}^3$ is determined by solving:

$$\inf_{\varphi_h \in \mathcal{U}_h} J(\varphi_h)$$

with

$$\mathcal{U}_h = \{\varphi_h \in X_h, \text{dist}(\varphi_h(T_1), \varphi_h(T_2)) \geq \varepsilon, \forall T_1, T_2 \in \mathcal{T}_h\}$$



Multi-body contact

Let M be a **family** of solids $M = (M_1, M_2, \dots)$
and let \mathcal{T}_h be a P_1 finite element mesh of M :

$$X_h = \{\varphi_h \in C^0(M; \mathbb{R}^3), \varphi_h|_T \in P_1, \forall T \in \mathcal{T}_h\}$$

Let J be the energy of the solids.

The deformation $\varphi_h : \cup_i M_i \rightarrow \mathbb{R}^3$ is determined by solving:

$$\inf_{\varphi_h \in \mathcal{U}_h} J(\varphi_h)$$

with

$$\mathcal{U}_h = \{\varphi_h \in X_h, \text{dist}(\varphi_h(T_1), \varphi_h(T_2)) \geq \varepsilon, \forall T_1, T_2 \in \mathcal{T}_h\}$$



Remark

The constraint is non-convex

Multi-body contact

The algorithm *(O. Pantz, 2007)*

- **The idea:** replace this non-convex optimization problem with a sequence of convex ones.
- It manages directly self-contacts and thin-structures

Remark

Other contact algorithms has been used in FSI, for example:

Stadler-Holzapfel-Korelc, 2003, Tezduyar-Sathe, 2007

Multi-body contact

The algorithm *(O. Pantz, 2007)*

- **The idea:** replace this non-convex optimization problem with a sequence of convex ones.
- It manages directly self-contacts and thin-structures

Main Steps

- Initial guess: φ_0

Remark

Other contact algorithms has been used in FSI, for example:

Stadler-Holzapfel-Korelc, 2003, Tezduyar-Sathe, 2007

Multi-body contact

The algorithm (O. Pantz, 2007)

- **The idea:** replace this non-convex optimization problem with a sequence of convex ones.
- It manages directly self-contacts and thin-structures

Main Steps

- Initial guess: φ_0
- Solve for $k \geq 0$

$$J(\varphi_h^{k+1}) = \inf_{\psi_h \in T(\varphi_h^k)} J(\psi_h)$$

where $T(\varphi_h^k)$ is a convex neighborhood of φ_h^k (to be defined)

Remark

Other contact algorithms has been used in FSI, for example:

Stadler-Holzapfel-Korelc, 2003, Tezduyar-Sathe, 2007

Multi-body contact

The algorithm (O. Pantz, 2007)

- **The idea:** replace this non-convex optimization problem with a sequence of convex ones.
- It manages directly self-contacts and thin-structures

Main Steps

- Initial guess: φ_0
- Solve for $k \geq 0$

$$J(\varphi_h^{k+1}) = \inf_{\psi_h \in T(\varphi_h^k)} J(\psi_h)$$

where $T(\varphi_h^k)$ is a convex neighborhood of φ_h^k (to be defined)

- Iterate on k until $\varphi_h^{k+1} \approx \varphi_h^k$

Remark

Other contact algorithms has been used in FSI, for example:

Stadler-Holzapfel-Korelc, 2003, Tezduyar-Sathe, 2007

Multi-body contact

Definition of the convex neighborhood

$$T(\psi_h) = \left\{ \varphi_h \in X_h, \min_{x_e \in e} \mathbf{n}_{e,x}(\psi_h) \cdot (\varphi_h(x_e) - \varphi_h(x)) \geq \varepsilon, \right. \\ \left. \text{for all edges } e \text{ and all nodes } x \notin e \right\}$$

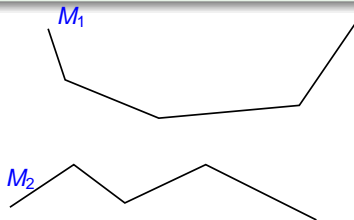
Multi-body contact

Definition of the convex neighborhood

$$T(\psi_h) = \left\{ \varphi_h \in X_h, \min_{x_e \in e} \mathbf{n}_{e,x}(\psi_h) \cdot (\varphi_h(x_e) - \varphi_h(x)) \geq \varepsilon, \right. \\ \left. \text{for all edges } e \text{ and all nodes } x \notin e \right\}$$

where $\mathbf{n}_{e,x}(\psi_h)$ is defined by

$$\min_{x_e \in e} \mathbf{n}_{e,x}(\psi_h) \cdot (\psi_h(x_e) - \psi_h(x)) = \text{dist}(\psi_h(e), \psi_h(x))$$



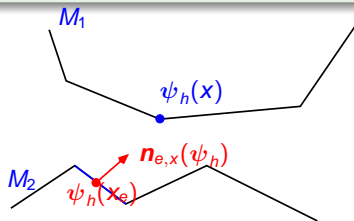
Multi-body contact

Definition of the convex neighborhood

$$T(\psi_h) = \left\{ \varphi_h \in X_h, \min_{x_e \in e} \mathbf{n}_{e,x}(\psi_h) \cdot (\varphi_h(x_e) - \varphi_h(x)) \geq \varepsilon, \right. \\ \left. \text{for all edges } e \text{ and all nodes } x \notin e \right\}$$

where $\mathbf{n}_{e,x}(\psi_h)$ is defined by

$$\min_{x_e \in e} \mathbf{n}_{e,x}(\psi_h) \cdot (\psi_h(x_e) - \psi_h(x)) = \text{dist}(\psi_h(e), \psi_h(x))$$



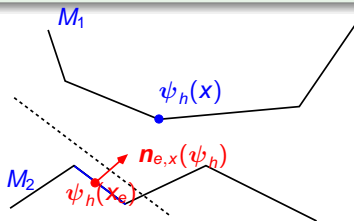
Multi-body contact

Definition of the convex neighborhood

$$T(\psi_h) = \left\{ \varphi_h \in X_h, \min_{x_e \in e} \mathbf{n}_{e,x}(\psi_h) \cdot (\varphi_h(x_e) - \varphi_h(x)) \geq \varepsilon, \right. \\ \left. \text{for all edges } e \text{ and all nodes } x \notin e \right\}$$

where $\mathbf{n}_{e,x}(\psi_h)$ is defined by

$$\min_{x_e \in e} \mathbf{n}_{e,x}(\psi_h) \cdot (\psi_h(x_e) - \psi_h(x)) = \text{dist}(\psi_h(e), \psi_h(x))$$



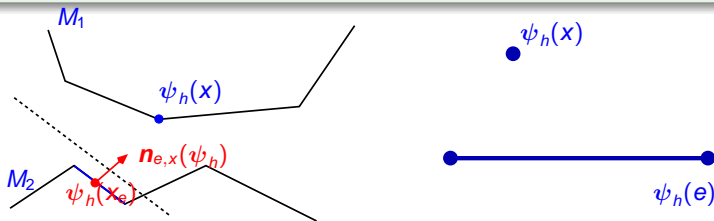
Multi-body contact

Definition of the convex neighborhood

$$T(\psi_h) = \left\{ \varphi_h \in X_h, \min_{x_e \in e} \mathbf{n}_{e,x}(\psi_h) \cdot (\varphi_h(x_e) - \varphi_h(x)) \geq \varepsilon, \right. \\ \left. \text{for all edges } e \text{ and all nodes } x \notin e \right\}$$

where $\mathbf{n}_{e,x}(\psi_h)$ is defined by

$$\min_{x_e \in e} \mathbf{n}_{e,x}(\psi_h) \cdot (\psi_h(x_e) - \psi_h(x)) = \text{dist}(\psi_h(e), \psi_h(x))$$



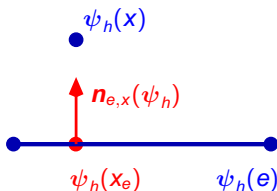
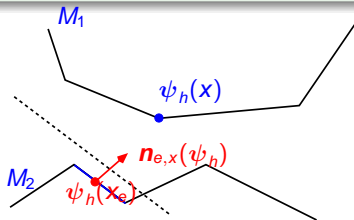
Multi-body contact

Definition of the convex neighborhood

$$T(\psi_h) = \left\{ \varphi_h \in X_h, \min_{x_e \in e} \mathbf{n}_{e,x}(\psi_h) \cdot (\varphi_h(x_e) - \varphi_h(x)) \geq \varepsilon, \right. \\ \left. \text{for all edges } e \text{ and all nodes } x \notin e \right\}$$

where $\mathbf{n}_{e,x}(\psi_h)$ is defined by

$$\min_{x_e \in e} \mathbf{n}_{e,x}(\psi_h) \cdot (\psi_h(x_e) - \psi_h(x)) = \text{dist}(\psi_h(e), \psi_h(x))$$



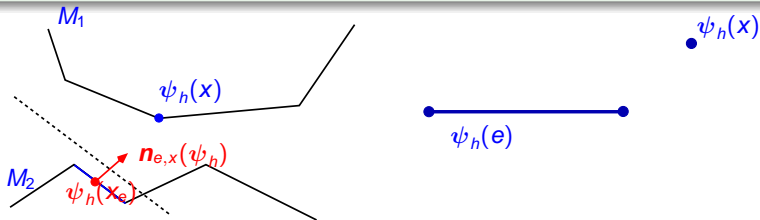
Multi-body contact

Definition of the convex neighborhood

$$T(\psi_h) = \left\{ \varphi_h \in X_h, \min_{x_e \in e} \mathbf{n}_{e,x}(\psi_h) \cdot (\varphi_h(x_e) - \varphi_h(x)) \geq \varepsilon, \right. \\ \left. \text{for all edges } e \text{ and all nodes } x \notin e \right\}$$

where $\mathbf{n}_{e,x}(\psi_h)$ is defined by

$$\min_{x_e \in e} \mathbf{n}_{e,x}(\psi_h) \cdot (\psi_h(x_e) - \psi_h(x)) = \text{dist}(\psi_h(e), \psi_h(x))$$



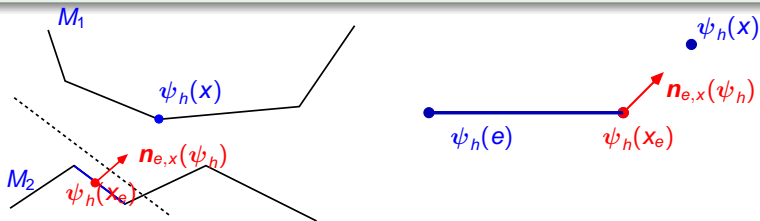
Multi-body contact

Definition of the convex neighborhood

$$T(\psi_h) = \left\{ \varphi_h \in X_h, \min_{x_e \in e} \mathbf{n}_{e,x}(\psi_h) \cdot (\varphi_h(x_e) - \varphi_h(x)) \geq \varepsilon, \right. \\ \left. \text{for all edges } e \text{ and all nodes } x \notin e \right\}$$

where $\mathbf{n}_{e,x}(\psi_h)$ is defined by

$$\min_{x_e \in e} \mathbf{n}_{e,x}(\psi_h) \cdot (\psi_h(x_e) - \psi_h(x)) = \text{dist}(\psi_h(e), \psi_h(x))$$



Remarks

- The generalization to the three dimensional case is direct but two kinds of contacts have to be considered: **vertex-element** and **edge-edge**

Remarks

- The generalization to the three dimensional case is direct but two kinds of contacts have to be considered: **vertex-element** and **edge-edge**
- The convex neighborhood can also be rewritten as:

$$T(\psi_h) = \left\{ \varphi_h \in X_h, F_{e,x}^0(\varphi_h) \leq 0, F_{e,x}^1(\varphi_h) \leq 0, \right. \\ \left. \text{for all edges } e \text{ and all nodes } x \notin e \right\}$$

$$F_{e,x}^j(\varphi_h) = \varepsilon - \mathbf{n}_{e,x}(\psi_h) \cdot (\varphi_h(e_j) - \varphi_h(x))$$

Remarks

- The generalization to the three dimensional case is direct but two kinds of contacts have to be considered: **vertex-element** and **edge-edge**
- The convex neighborhood can also be rewritten as:

$$T(\psi_h) = \left\{ \varphi_h \in X_h, F_{e,x}^0(\varphi_h) \leq 0, F_{e,x}^1(\varphi_h) \leq 0, \right. \\ \left. \text{for all edges } e \text{ and all nodes } x \notin e \right\}$$

$$F_{e,x}^j(\varphi_h) = \varepsilon - \mathbf{n}_{e,x}(\psi_h) \cdot (\varphi_h(e_j) - \varphi_h(x))$$

the same kind of convex constraints as in the first example

Remarks

- The generalization to the three dimensional case is direct but two kinds of contacts have to be considered: **vertex-element** and **edge-edge**
- The convex neighborhood can also be rewritten as:

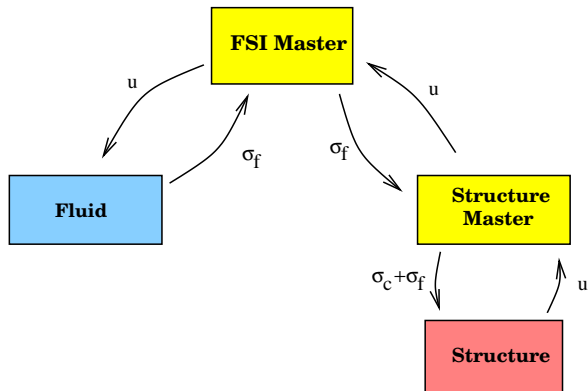
$$T(\psi_h) = \left\{ \varphi_h \in X_h, F_{e,x}^0(\varphi_h) \leq 0, F_{e,x}^1(\varphi_h) \leq 0, \right. \\ \left. \text{for all edges } e \text{ and all nodes } x \notin e \right\}$$

$$F_{e,x}^j(\varphi_h) = \varepsilon - \mathbf{n}_{e,x}(\psi_h) \cdot (\varphi_h(e_j) - \varphi_h(x))$$

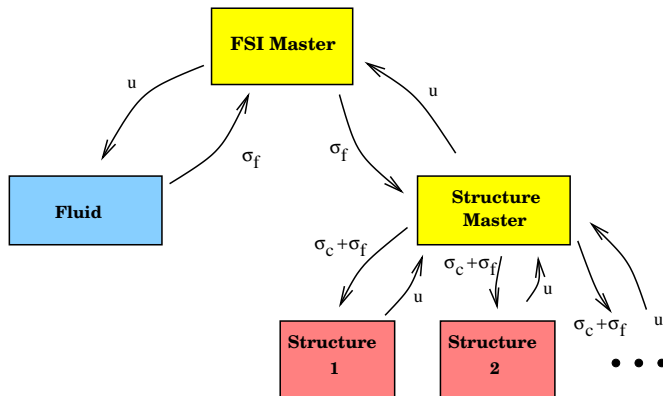
the same kind of convex constraints as in the first example

For the inner iterations of this new algorithm, one can reuse what has been done in the case of solid-wall contact.

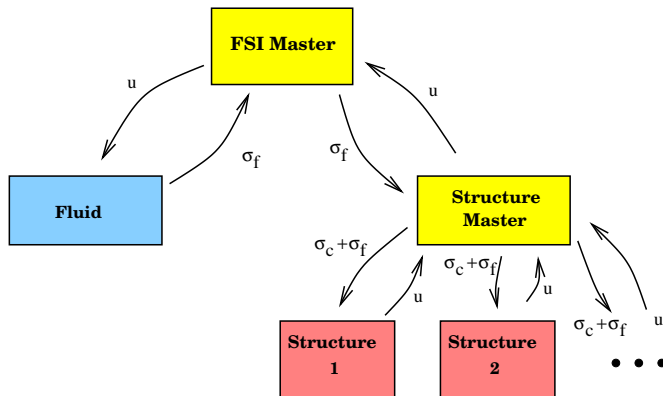
FSI & Multi-body contact



FSI & Multi-body contact



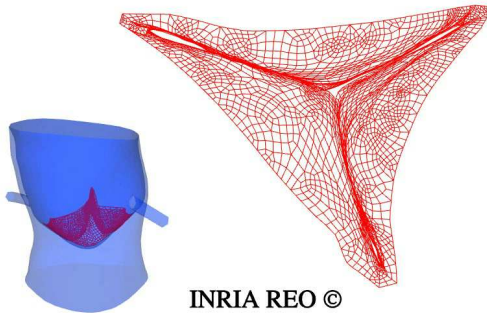
FSI & Multi-body contact



Nothing has been changed within the solvers

Contact among leaflets

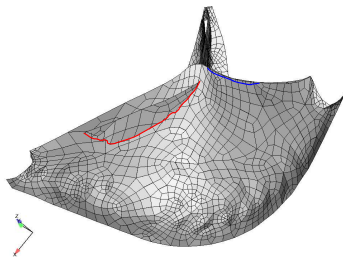
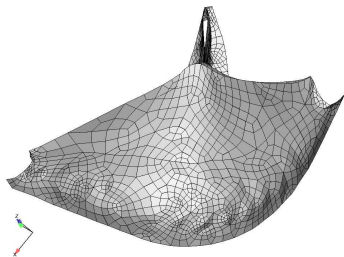
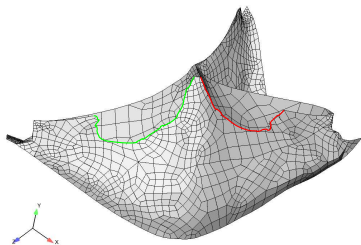
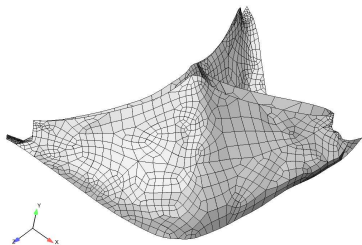
Time = 0.4800



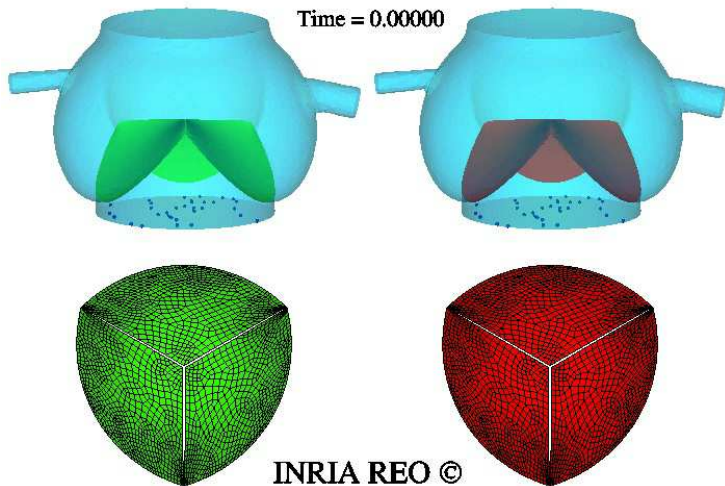
Aortic valve

(M.A., J.F. Gerbeau, O. Pantz and K.F. Traoré, 2008)

Contact among leaflets



Contact among leaflets

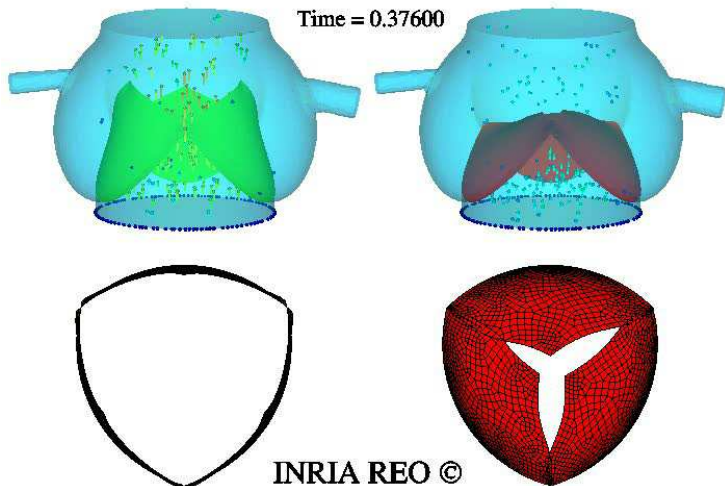


INRIA REO ©

Aortic valve

(M.A., J.F. Gerbeau, O. Pantz and K.F. Traoré, 2008)

Contact among leaflets



INRIA REO ©

(M.A., J.F. Gerbeau, O. Pantz and K.F. Traoré, 2008)

Conclusion and Future Works

Conclusions

The FSI semi-implicit projection algorithm

- is a performing scheme for FSI problems with strong added-mass effect;
- is conditionally stable...but generally stable in all practical situations.
- is at least $O(\sqrt{\delta t})$ in time.

The Fictitious Domain method

- is a robust method in case of very large displacements;
- could be efficiently implemented in a partitioned scheme and coupled to a contact algorithm.

The use of a contact algorithm shows remarkable differences in valve simulations.

Conclusion and Future Works

Remarks

The computational cost of the fluid-structure interaction procedure with contact handling is still quite expensive, this is mainly due to:

- the fixed-point iterations in the FSI problem;
- the gradient iteration with projection in the contact problem.

Current and Future works

For FSI semi-implicit projection algorithm:

- Stability condition and convergence in the case of the Newmark time discretization scheme.

In valves simulations:

- Extension of FSI semi-implicit projection method to valves simulations;
- Improvement of the optimization algorithm.

These works are partially supported by...

Cardiosense3D

CardioSense3D is a 4-year Large Initiative Action launched in 2005 and funded by the French national research center INRIA which focuses on the modeling and the estimation of the cardiac activity.



<http://www-sop.inria.fr/CardioSense3D/>

Research Themes

- Data assimilation
- Modeling and coupling of physiological phenomena:
 - ▶ Electrophysiology
 - ▶ Fluid dynamics
 - ▶ Solid mechanics
 - ▶ Metabolism and perfusion

Members

Four INRIA research teams: **Asclepios**, **Macs**, **Reo** and **Sisyphé** with multiple academic, clinical and industrial partnerships.