Numerical Study of a Monolithic Fluid-Structure Formulation

Olivier Pironneau

▶ To cite this version:

Olivier Pironneau. Numerical Study of a Monolithic Fluid-Structure Formulation. Aldo Friedriani. Variational Analysis and Aerospace Engineering, 116, Springer Verlag, pp.401-420, 2016, Springer Optimization and Its Applications, 978-3-319-45679-9. 10.1007/978-3-319-45680-5 . hal-01270463

HAL Id: hal-01270463

https://hal.sorbonne-universite.fr/hal-01270463

Submitted on 8 Feb 2016

Numerical Study of a Monolithic Fluid-Structure Formulation

Olivier Pironneau

Abstract The conservation laws of continuum mechanic are naturally written in an Eulerian frame where the difference between a fluid and a solid is only in the expression of the stress tensors, usually with Newton's hypothesis for the fluids and Helmholtz potentials of energy for hyperelastic solids. There are currently two favored approaches to Fluid Structured Interactions (FSI) both working with the equations for the solid in the initial domain; one uses an ALE formulation for the fluid and the other matches the fluid-structure interfaces using Lagrange multipliers and the immersed boundary method. By contrast the proposed formulation works in the frame of physically deformed solids and proposes a discretization where the structures have large displacements computed in the deformed domain together with the fluid in the same; in such a monolithic formulation velocities of solids and fluids are computed all at once in a single variational formulation by a semi-implicit in time and the finite element method. Besides the simplicity of the formulation the advantage is a single algorithm for a variety of problems including multi-fluids, free boundaries and FSI. The idea is not new but the progress of mesh generators renders this approach feasible and even reasonably robust. In this article the method and its discretization are presented, stability is discussed showing in a loose fashion were are the difficulties and why one is able to show convergence of monolithic algorithms on fixed domains for fluids in compliant shell vessels restricted to small displacements. A numerical section discusses implementation issues and presents a few simple tests.

Sorbonne Universités, UPMC (Paris VI)
Laboratoire Jacques-Louis Lions e-mail: olivier.pironneau@upmc.fr

Submitted to "Variational Analysis and Aerospace Engineering: Mathematical Challenges for the Aerospace of the Future", Aldo Frediani, Bijan Mohammadi, Olivier Pironneau (eds). Springer 2016.

AMS classification

65M60 (74F10 74S30 76D05 76M25).

Introduction

In an earlier paper [6] the author and his coauthors proposed a method to compute a fluid in a vessel modeled as a shell with normal displacements as in Nobile and Vergara [23]. It was argued that since the model is valid for small displacements only, one may as well use a transpiration approximation for the fluid and do the full computation in a fixed domain. As we were able to prove convergence, an interesting question arose: what is so special about the model that one could prove existence and convergence of the numerical scheme?

This paper answers partially the question: what makes FSI really hard is the moving domain. The same is true of free boundary problems for the Navier-Stokes equations. So it was the transpiration approximation for the moving part which made the analysis possible in [6].

Turning to ALE to work on a fixed domain both for the fluid and the solid is a popular solution[26], but the difficulty is transferred to the mesh[19] and the matching conditions at the fluid-solid interface[17]. Even more so with immersed boundary methods (IBM)[24][11], although the convergence analysis is more advanced[3].

Furthermore, iterative solvers for FSI which rely on alternative solutions of the fluid and the structure parts are subject to the added mass effect and require special solvers[12][5].

Every so often it is not a bad idea, I guess, to rethink fundamentals and check that what is taken for granted in numerical analysis is still true in the face of hardware and software progress.

So, is there an alternative to ALE and IBM? One old method [1] has resurfaced recently, the so-called *actualized Lagrangian methods* for computing structures [16] [20] (see also [10] although different from the present study because it deals mostly with membranes).

Continuum mechanics doesn't distinguish between solids and fluids till it comes to the constitutive equations. This has been exploited in many studies but most often in the context of ALE[18][27].

In the present study we investigate what Stephan Turek [27] calls a monolithic formulation but here in an Eulerian framework, following the displaced geometry of the fluid and the solid.

To the specialist it may appear to be a back to square one idea and it is true: there is nothing new here from the modeling view point; everyone knows that it can be done. What is new is that the mesh generators are now robust and agile at following complex motions of objects, making feasible an Eulerian numerical method.

The first difficulty with Eulerian methods comes from the hyperbolic character of the equations for the displacement of solids while those for the fluid are parabolic in time for the velocity. So let us begin by showing that a wave equation for a displacement can be reformulated as a seemingly parabolic equation for its velocity.

Preliminaries on the Wave Equation

At the core of the numerical scheme proposed here for large displacements is the following rewriting of the first and second order finite difference in time schemes for the wave equation:

$$\begin{aligned}&\partial_{tt}d-\Delta d=f,\ \forall x\in\Omega,\ \ \forall t\in(0,T);\ \ d(x,t)=0\ \forall x\in\partial\Omega;\ \forall t\in(0,T)\\&d_{|t=0}=\partial_t d_{|t=0}=0,\ \forall x\in\Omega.\end{aligned}\qquad(1)$$

For $\alpha=0,\theta=1$ or $\alpha=1,\theta\in[\frac{1}{4},\frac{1}{2}]$ the following scheme is unconditionally stable on a uniform finite difference grid in 1D(see for instance [21]):

$$\frac{d_j^{n+1}-2d_j^n+d_j^{n-1}}{\delta t^2}-\Delta_j[\theta d^{n+1}+\alpha((1-2\theta)d^n+\theta d^{n-1})]=f^n,\ \ d^0=d^1=0,$$

where $\Delta_j d=(d_{j+1}-2d_j+d_{j-1})/\delta x^2$. With $\alpha=0$ it is first order in time; it is second order when $\alpha=1$.

By introducing $u_j^{n+1}=\frac{1}{\delta t}(d_j^{n+1}-d_j^n)$, these schemes can be rewritten as

$$\frac{u_j^{n+1}-u_j^n}{\delta t}-\Delta_j d^n-\theta\delta t\Delta_j(u^{n+1}-\alpha u^n)=f^n,\ \ d_j^{n+1}=d_j^n+\delta t u_j^{n+1},$$

and initialized by $u^0=d^0=0$. Evidently this is also unconditionally stable and first order when $\alpha=0,\ \theta=1$ and second order when $\alpha=1,\ \theta\in[\frac{1}{4},\frac{1}{2}]$.

1 General Laws of Continuum Mechanics

Consider a time dependent computational domain Ω_t made of a fluid region Ω_t^f and a solid region Ω_t^s: $\overline{\Omega}_t=\overline{\Omega}_t^f\cup\overline{\Omega}_t^s$, $\Omega_t^f\cap\Omega_t^s=\emptyset$ at all times. The fluid-structure interface is denoted $\Sigma_t=\overline{\Omega}_t^f\cap\overline{\Omega}_t^s$ and the boundary of Ω_t is $\partial\Omega_t$.

At initial time Ω_0^f and Ω_0^s are prescribed. The following notations are standard [8],[22],[1],[27],[18]:

- $\mathbf{X}:\Omega_0\times(0,T)\mapsto\Omega_t$: $\mathbf{X}(x^0,t)$, the Lagrangian position at t of x^0.
- $\mathbf{u}=\partial_t\mathbf{X}$, the velocity of the deformation,

- $\mathbf{F}_{ji} = \partial_{x_i^0}\mathbf{X}_j$, the transposed gradient of the deformation,
- $J = \det_{\mathbf{F}}$, the jacobian of the deformation.

Remark 1. We use a notation for the gradient which is the transposed of the one found in engineering anglo-saxon books (see [8] for instance). Here the gradient of a scalar being a column vector the gradient of a vector is the row of vectors of the derivative of its components.

We denote by tr_A and $\det_A$ the trace and determinant of A. As usual the following quantities are introduced:

- the density $\rho(x,t) = \mathbf{1}_{\Omega_t^f}\rho^f(x,t) + \mathbf{1}_{\Omega_t^s}\rho^s(x,t)$, at $x \in \Omega_t,\ t \in (0,T)$,
- the stress tensor $\sigma(x,t) = \mathbf{1}_{\Omega_t^f}\sigma^f(x,t) + \mathbf{1}_{\Omega_t^s}\sigma^s(x,t)$,
- $\mathbf{f}(x,t)$,$\mathbf{B}(x,t)$ the density of volumic and surfacic forces at x,t.
- $\mathbf{d} = \mathbf{X}(x^0,t) - x^0$, the displacement.

Finally and unless specified all spatial derivatives are with respect to $x \in \Omega_t$ and not with respect to $x^0 \in \Omega_0$. If ϕ is a function of $x = \mathbf{X}(x^0,t),\ x^0 \in \Omega_0$,

$$\nabla_{x^0}\phi = [\partial_{x_i^0}\phi] = [\partial_{x_i^0}\mathbf{X}_j\partial_{x_j}\phi] = \mathbf{F}^T\nabla\phi.$$

When $\mathbf{X}$ is one-to-one and invertible, $\mathbf{d}$ and $\mathbf{F}$ can be seen as functions of (x,t) instead of (x^0,t). They are related by

$$\mathbf{F}^T = \nabla_{x^0}\mathbf{X} = \nabla_{x^0}(\mathbf{d} + x^0) = \nabla_{x^0}\mathbf{d} + \mathbf{I} = \mathbf{F}^T\nabla\mathbf{d} + \mathbf{I}, \quad \Rightarrow \quad \mathbf{F} = (\mathbf{I} - \nabla\mathbf{d})^{-T}$$

Time derivatives are related by

$$D_t\phi := \frac{d}{dt}\phi(\mathbf{X}(x_0,t),t) = \partial_t\phi(x,t) + \mathbf{u}\cdot\nabla\phi(x,t).$$

It is convenient to introduce the notation

$$\mathrm{D}\mathbf{u} = \nabla\mathbf{u} + \nabla\mathbf{u}^T.$$

Conservation of momentum and conservation of mass take the same form for the fluid and the solid:

$$\rho D_t\mathbf{u} = \mathbf{f} + \nabla\cdot\sigma, \qquad \frac{d}{dt}(J\rho) = 0,$$

So $J\rho = \rho_0$ at all times and

$$J^{-1}\rho_0 D_t\mathbf{u} = f + \nabla\cdot\sigma \text{ in } \Omega_t,\ \forall t \in (0,T), \tag{2}$$

with continuity of $\mathbf{u}$ and of $\sigma\cdot\mathbf{n}$ at the fluid-structure interface Σ when $\mathbf{B} = 0$. There are also unwritten constraints pertaining to the realizability of the map $\mathbf{X}$ (see [8],[22]). Finally incompressibility implies $J = 1$ and so $\rho = \rho_0$ constant.

1.1 Constitutive Equations

.

- For a Newtonian incompressible fluid : $\sigma = -p^f\mathbf{I} + \mu^f \mathrm{D}\mathbf{u}$
- For an hyperelastic incompressible material : $\sigma = -p^s\mathbf{I} + \rho^s\partial_{\mathbf{F}}\Psi\mathbf{F}^T$
  where Ψ is the Helmholtz potential which, in the case of a Mooney-Rivlin two dimensional material, is [8]

$$\Psi(\mathbf{F}) = c_1\mathrm{tr}_{\mathbf{F}^T\mathbf{F}} + c_2(\mathrm{tr}_{(\mathbf{F}^T\mathbf{F})^2} - \mathrm{tr}^2_{\mathbf{F}^T\mathbf{F}}).$$

For a compressible Mooney-Rivlin material the same holds but without p^s. Here we will only consider incompressible material, but everything said below can be adapted easily.

1.2 Computation of the Mooney-Rivlin 2D Stress Tensor

It is readily seen that

$$\partial_{\mathbf{F}}\mathrm{tr}_{\mathbf{F}^T\mathbf{F}} = ((\partial_{\mathbf{F}_{ij}}\sum_{m,n} F^2_{m,n})) = 2\mathbf{F}$$

Similarly

$$\partial_{\mathbf{F}}\mathrm{tr}_{(\mathbf{F}^T\mathbf{F})^2} = ((\partial_{\mathbf{F}_{ij}}\sum_{n,m,p,k} F_{n,k}F_{n,m}F_{p,m}F_{p,k})) = 4\mathbf{F}\mathbf{F}^T\mathbf{F}$$

Hence

$$\Psi(\mathbf{F}) = c_1\mathrm{tr}_{\mathbf{F}^T\mathbf{F}} + c_2(\mathrm{tr}_{(\mathbf{F}^T\mathbf{F})^2} - \mathrm{tr}^2_{\mathbf{F}^T\mathbf{F}}) \Rightarrow \partial_{\mathbf{F}}\Psi = 2c_1\mathbf{F} + c_2(4\mathbf{F}\mathbf{F}^T\mathbf{F} - 4\mathrm{tr}_{\mathbf{F}^T\mathbf{F}}\mathbf{F})$$

Let $\mathbf{B} := \mathbf{F}\mathbf{F}^T = \big((\mathbf{I}-\nabla\mathbf{d})(\mathbf{I}-\nabla\mathbf{d})^T\big)^{-1}$, $b :=\mathrm{det}_{\mathbf{B}}$, $c := \mathrm{tr}_{\mathbf{B}} = \mathrm{tr}_{\mathbf{F}^T\mathbf{F}}$. Then

$$\partial_{\mathbf{F}}\Psi\mathbf{F}^T = (2c_1 - 4c_2c)\mathbf{B} + 4c_2\mathbf{B}^2.$$

Now by the Cayley-Hamilton theorem $\mathbf{B}^2 = c\mathbf{B} - b\mathbf{I}$ so

$$\partial_{\mathbf{F}}\Psi\mathbf{F}^T = 2c_1\mathbf{B} - 4c_2b\mathbf{I} = 2c_1\mathbf{F}\mathbf{F}^T - 4c_2\mathrm{det}_{\mathbf{F}\mathbf{F}^T}\mathbf{I}$$

By the Cayley-Hamilton theorem again

$$\mathbf{B} = c\mathbf{I} - b\mathbf{B}^{-1} = c\mathbf{I} - b(\mathbf{I} - \nabla\mathbf{d} - \nabla\mathbf{d}^T + \nabla\mathbf{d}\nabla\mathbf{d}^T).$$

So

$$\begin{aligned}\partial_{\mathbf{F}}\Psi\mathbf{F}^T &= (2c_1(c-b) - 4c_2b)\mathbf{I} + 2c_1b(\mathrm{D}\mathbf{d} - \nabla\mathbf{d}\nabla\mathbf{d}^T) \\ &= 2c_1\mathrm{det}_{\mathbf{F}\mathbf{F}^T}(\mathrm{D}\mathbf{d} - \nabla\mathbf{d}\nabla\mathbf{d}^T) + (2c_1\mathrm{tr}_{\mathbf{F}\mathbf{F}^T} - (2c_1 + 4c_2)\mathrm{det}_{\mathbf{F}\mathbf{F}^T})\mathbf{I}\end{aligned}$$

Hence an incompressible two dimensional Mooney-Rivlin material will have, for some α,

$$\partial_{\mathbf{F}}\Psi\mathbf{F}^T = 2c_1(\mathrm{D}\mathbf{d} - \nabla\mathbf{d}\nabla\mathbf{d}^T) + \alpha\mathbf{I}.$$

2 Monolithic Variational Formulation in 2D

Let $\Gamma \subset \partial\Omega$ the part of the boundary on which the solid is clamped or the fluid has a no-slip condition. For incompressible material the final fluid-structure formulation in two dimensions is:

Find $(\mathbf{u}, p)$ with $\mathbf{u}_{|\Gamma} = 0$, $\mathbf{d}$ and $\Omega_t^r,\ r = s, f$ solutions of

$$\begin{aligned}\int_{\Omega_t}\Big[\rho D_t\mathbf{u}\cdot\hat{\mathbf{u}}\ -p\nabla\cdot\hat{\mathbf{u}} - \hat{p}\nabla\cdot\mathbf{u} + \mathbf{1}_{\Omega_t^f}\frac{\nu}{2}\mathrm{D}\mathbf{u} : \mathrm{D}\hat{\mathbf{u}}\\ +\mathbf{1}_{\Omega_t^s}\tilde{c}_1(\mathrm{D}\mathbf{d} - \nabla\mathbf{d}\nabla\mathbf{d}^T) : \mathrm{D}\hat{\mathbf{u}}\Big] = \int_{\Omega_t} f\cdot\hat{\mathbf{u}}\\ \mathrm{D}_t\mathbf{d} = \mathbf{u},\end{aligned} \tag{3}$$

for all $(\hat{\mathbf{u}}, \hat{p})$ with $\hat{\mathbf{u}}_{|\Gamma} = 0$, where Ω_t^s and Ω_t^f are defined incrementally by

$$\frac{dX}{\mathrm{d}\tau} = \mathbf{u}(X(\tau),\tau),\ \ X(t) \in \Omega_t^r\ \Rightarrow\ X(\tau) \in \Omega_\tau^r\ \ \forall\tau \in (0,T),\ r = s, f$$

Initial conditions are: $\mathbf{u}$ given, $\mathbf{d} = 0$, Ω_0^r given, $r = s, f$.

We have used the notation $\mathbf{B} : \mathbf{C} = \mathrm{tr}_{\mathbf{B}^T\mathbf{C}}$ and $\tilde{c}_1 := \rho^s c_1$.

2.1 Conservation of Energy

Proposition 1.

$$\frac{d}{dt}\int_{\Omega_t}\frac{\rho}{2}|\mathbf{u}|^2 + \frac{\nu}{2}\int_{\Omega_t^f}|\mathrm{D}\mathbf{u}|^2 + \frac{d}{dt}\int_{\Omega_0^s}\Psi(\mathbf{I} + \nabla_{x^0}\mathbf{d}^T) = \int_{\Omega_t} f\cdot\mathbf{u}$$

When Ψ is convex, an existence of solution result can be gained from this equality (see [14] for example).

Proof. Choosing $\hat{\mathbf{u}} = \mathbf{u},\ \hat{p} = -p$ will give the proposition provided

$$\int_{\Omega_t^s}(\mathrm{D}\mathbf{d} - \nabla\mathbf{d}\nabla\mathbf{d}^T) : \mathrm{D}\partial_t\mathbf{d} = \frac{d}{dt}\int_{\Omega_0^s}\Psi(\nabla_{x^0}\mathbf{X}).$$

By construction

$$\int_{\Omega_t^s} c_0(\mathrm{D}\mathbf{d} - \nabla\mathbf{d}\nabla\mathbf{d}^T) : \mathrm{D}\hat{\mathbf{u}} = \int_{\Omega_t^s} (\partial_{\mathbf{F}}\Psi(\mathbf{F})\mathbf{F}^T - \alpha\mathbf{I}) : \mathrm{D}\hat{\mathbf{u}} = \int_{\Omega_0^s} \partial_{\mathbf{F}}\Psi(\mathbf{F}) : \mathrm{D}_{x^0}\hat{\mathbf{u}}$$

Now as $\frac{d}{dt}\Psi(\mathbf{F}) = \partial_{\mathbf{F}}\Psi(\mathbf{F}) : \partial_t\mathbf{F}$ and $\nabla_{x^0}\mathbf{u} = \partial_t\nabla_{x^0}\mathbf{d} = \partial_t\mathbf{F}^T$,

$$\int_{\Omega_t^s} c_0(\mathrm{D}\mathbf{d} - \nabla\mathbf{d}\nabla\mathbf{d}^T) : \mathrm{D}\mathbf{u} = \int_{\Omega_0^s} \frac{d}{dt}\Psi(\mathbf{F}) = \frac{d}{dt}\int_{\Omega_0^s} \Psi(\mathbf{I} + \nabla_{x^0}\mathbf{d}^T).$$

2.2 Discretization in Time

It is natural to use the following discretization

$$\int_{\Omega_t} (\mathrm{D}\mathbf{d} - \nabla\mathbf{d}\nabla\mathbf{d}^T) : \mathrm{D}\hat{\mathbf{u}} \approx \int_{\Omega_n} (\mathrm{D}\mathbf{d}^{n+1} - \nabla\mathbf{d}^{n+1}\nabla\mathbf{d}^{n+1^T}) : \mathrm{D}\hat{\mathbf{u}}$$

with $\mathbf{d}^{n+1} = \mathbf{d}^n + \delta t\mathbf{u}^{n+1}$. Hence

$$\mathrm{D}\mathbf{d} - \nabla\mathbf{d}\nabla\mathbf{d}^T \approx \mathrm{D}\mathbf{d}^n - \nabla\mathbf{d}^n\nabla\mathbf{d}^{nT} + \delta t(\mathrm{D}\mathbf{u}^{n+1} - \nabla\mathbf{u}^{n+1}\nabla\mathbf{d}^{nT} - \nabla\mathbf{d}^n\nabla\mathbf{u}^{n+1^T}) + o(\delta t).$$

So if X^n is a first order approximation of $X(t^{n+1} - \delta t)$ defined by

$$\dot{X} = \mathbf{u}(X(\tau), \tau),\ X(t^{n+1}) = x$$

such as $X^n(x) = x - \delta t\mathbf{u}^n(x)$, a consistent first order scheme is to find $\mathbf{u}^{n+1}, p^{n+1}$ such that $\mathbf{u}^{n+1} = 0$ on Γ and for all $\hat{\mathbf{u}}, \hat{p}$ with $\hat{\mathbf{u}}_{|\Gamma} = 0$,

$$\begin{aligned}
&\int_{\Omega_n} \Big[\rho^n \frac{\mathbf{u}^{n+1} - \mathbf{u}^n{\circ}X^n}{\delta t} \cdot \hat{\mathbf{u}} - p^{n+1}\nabla\cdot\hat{\mathbf{u}} - \hat{p}\nabla\cdot\mathbf{u}^{n+1} + \mathbf{1}_{\Omega_n^f}\frac{\nu}{2}\mathrm{D}\mathbf{u}^{n+1} : \mathrm{D}\hat{\mathbf{u}} \\
&+\tilde{c}_1\mathbf{1}_{\Omega_n^s}[\mathrm{D}\mathbf{d}^n - \nabla\mathbf{d}^{n^T}\nabla\mathbf{d}^n + \delta t(\mathrm{D}\mathbf{u}^{n+1} - \nabla\mathbf{u}^{n+1}\nabla\mathbf{d}^{nT} - \nabla\mathbf{d}^n\nabla\mathbf{u}^{n+1^T})] : \mathrm{D}\hat{\mathbf{u}}\Big] \\
&= \int_{\Omega_n} f\hat{\mathbf{u}}, \\
&\mathbf{d}^{n+1} = \mathbf{d}^n \circ X^n + \delta t\mathbf{u}^{n+1},\ \Omega_{n+1}^r = \{x + \delta t\mathbf{u}^{n+1}(x) : x \in \Omega_n^r\},\ r = s, f
\end{aligned} \tag{4}$$

2.3 Spatial Discretization with Finite Elements

Let $\mathscr{T}_h^0$ be a triangulation of the initial domain. Spatial discretization can be done with Lagrangian triangular elements of degree 2 for the space V_h of velocities and displacements and Lagrangian triangular elements of degree 1 for the pressure space

Q_h. A small penalization with parameter ε must be added to impose uniqueness of the pressure when a direct linear solver is used.

At each time step one must find $\mathbf{u}_h^{n+1}, p_h^{n+1} : \forall \hat{\mathbf{u}}_h \in V_{0h}, \forall \hat{p}_h \in Q_h$

$$\int_{\Omega_n} \Big[\rho^n \frac{\mathbf{u}_h^{n+1} - \mathbf{u}_h^n \circ X^n}{\delta t} \cdot \hat{\mathbf{u}}_h - p_h^{n+1} \nabla \cdot \hat{\mathbf{u}}_h - \hat{p}_h \nabla \cdot \mathbf{u}_h^{n+1} + \mathbf{1}_{\Omega_n^f} \frac{\nu}{2} \mathrm{D}\mathbf{u}_h^{n+1} : \mathrm{D}\hat{\mathbf{u}}_h$$
$$+ \tilde{c}_1 \mathbf{1}_{\Omega_n^s} [\mathrm{D}\mathbf{d}_h^n - \nabla \mathbf{d}^{n^T} \nabla \mathbf{d}_h^n + \delta t (\mathrm{D}\mathbf{u}_h^{n+1} - \nabla \mathbf{u}_h^{n+1} \nabla \mathbf{d}_h^{n^T} - \nabla \mathbf{d}_h^n \nabla \mathbf{u}_h^{n+1^T})] : \mathrm{D}\hat{\mathbf{u}}_h$$
$$+ \varepsilon p_h \hat{p}_h \Big] = \int_{\Omega_n} f \hat{\mathbf{u}}_h, \tag{5}$$

Then the triangulation must be updated by $\mathbf{u}_h^{n+1}$ by moving each vertex from q_i^n to $q_i^{n+1} := q_i^n + \delta t \mathbf{u}_h^{n+1}$.

Let $\mathbf{d}_i^n := \mathbf{d}^n(q_i)$ then

$$\mathbf{d}^n \circ X^n(q_i^{n+1}) = \mathbf{d}^n(q_i^n + \delta t \mathbf{u}_h^{n+1} - \delta t \mathbf{u}_h^{n+1}) = \mathbf{d}^n(q_i^n)$$

so to implement $\mathbf{d}_h^{n+1} = \mathbf{d}_h^n \circ X^n + \delta t \mathbf{u}_h^{n+1}$ it suffices to copy the array of values of $\mathbf{d}_h^n$ in the array of values of $\mathbf{d}_h^{n+1}$ and add $\delta t \mathbf{u}_h^{n+1}(q_i^n)$ to the array.

The vertices in the fluid are moved by $\tilde{\mathbf{u}}$ solution of $-\Delta \tilde{\mathbf{u}} = 0$ in the fluid and $\tilde{\mathbf{u}} = \mathbf{u}$ on Σ and zero on other boundaries. Moving the vertices of $\mathscr{T}_h^n$ gives a new triangulation $\mathscr{T}_h^{n+1}$.

2.4 Map Preserving Scheme

It is important to understand on which triangulation each variable is defined. One possibilities is to assume that Ω_{n+1} is constructed by successive iterations such that $\Omega_n = \{x - \delta t \mathbf{u}_h^{n+1}(x),\ x \in \Omega_{n+1}\}$. Then $\mathbf{X}^{n+1}(x_0)$ is such that

$$\mathbf{X}^{n+1}(x_0) = \mathbf{X}^n(x_0) + \delta t \mathbf{u}_h^{n+1}(\mathbf{X}^{n+1}(x_0))$$

So $\mathbf{u}_h^{n+1}$ and $\mathbf{d}_h^{n+1}$ live on $\mathscr{T}_h^{n+1}$ and $\mathbf{u}_h^n$ and $\mathbf{d}_h^n$ live on $\mathscr{T}_h^n$.

Recall the notation $X^{n+1}(x) = x - \mathbf{u}^{n+1}(x) + o(\delta t)$, not to be confused with $\mathbf{X}^{n+1}$. Let $x = \mathbf{X}^{n+1}(x_0) \in \Omega_{n+1}$, then $X^{n+1}(x) \in \Omega_n$; let $\mathbf{F}^n(x) = (\nabla_{x_0} \mathbf{X}^n(x_0))^T$. Then

$$\nabla_{x_0} \mathbf{X}^{n+1}(x_0) = \nabla_{x_0} \mathbf{X}^n(x_0) + \delta t \nabla_{x_0} \mathbf{X}^{n+1}(x_0) \nabla \mathbf{u}_h^{n+1}(x), \tag{6}$$

Hence $\mathbf{F}^{n+1} = [\mathbf{I} - \delta t \nabla \mathbf{u}_h^{n+1}]^{-T} \mathbf{F}^n \circ X^{n+1}$ and

$$\mathbf{F}^{n+1} \mathbf{F}^{n+1^T} = [\mathbf{I} - \delta t \nabla \mathbf{u}_h^{n+1}]^{-T} (\mathbf{F}^n \mathbf{F}^{n^T}) \circ X^{n+1} [\mathbf{I} - \delta t \nabla \mathbf{u}_h^{n+1}(x)]^{-1} \tag{7}$$

As above, the Cayley-Hamilton theorem and incompressibility imply that

$$[\mathbf{I}-\delta t\nabla\mathbf{u}_h^{n+1}]^{-1}=\mathbf{I}+\delta t\nabla\mathbf{u}_h^{n+1}+o(\delta t)$$

Hence $\mathbf{B}^n(x)=\mathbf{F}^n(x)\mathbf{F}^{nT}(x)$ satisfies

$$\mathbf{B}^{n+1}=[\mathbf{I}+\delta t\nabla{\mathbf{u}_h^{n+1}}^T]\mathbf{B}^n\circ X^{n+1}[\mathbf{I}+\delta t\nabla\mathbf{u}_h^{n+1}]+o(\delta t) \tag{8}$$

This formula can be used in the variational formulation instead of $\nabla\mathbf{d}$. Its numerical performance is similar to the one that uses $\mathbf{d}$ but it preserves energy... at the cost of an iterative adjustment of Ω_{n+1}.

2.5 Remark on the Construction of Second order accurate schemes

To build a second order scheme for (3) one would have to

- use a second order characteristic method [4] for $D_t\mathbf{u}$,
- approximate $\mathbf{u}$ by $(\mathbf{u}^{n+1}+\mathbf{u}^n)/2$ in the fluid viscous term
- and $\mathbf{d}$ by $(\mathbf{d}^{n+1}+\mathbf{d}^{n-1})/2=\mathbf{d}^n+\delta t(\mathbf{u}^{n+1}-\mathbf{u}^n)/2$.
- But one must also approximate Ω_t by $\Omega_{n+\frac{1}{2}}$.

The last item is difficult; the construction of second order schemes is discussed in particular in the thesis of P. Hauret [14] with the Newmark mid-point scheme[15]. It seems rather difficult to prove the same here, so we postpone it to a future study.

2.6 Perturbation about an Equilibrium

An equilibrium is reached when $p,\mathbf{d}$ and $\Omega^s=\{x^0+\mathbf{d}(x^0):\ x^0\in\Omega_0^s\}$ are such that

$$\int_\Omega\Big[-p\nabla\cdot\hat{\mathbf{u}}+\tilde{c}_1\mathbf{1}_{\Omega^s}[\mathrm{D}\mathbf{d}-\nabla\mathbf{d}\nabla\mathbf{d}^T]:\mathrm{D}\hat{\mathbf{u}}\Big]=\int_\Omega f\hat{\mathbf{u}},\ \forall\hat{\mathbf{u}}, \tag{9}$$

Assuming for clarity that $\rho^s=\rho^f$ and Ω is independent of t, if we prime all variations about that state, they much be such that $\partial_t\mathbf{d}'=\mathbf{u}'$ and

$$\begin{aligned}&\int_\Omega\Big[\rho D_t\mathbf{u}'\cdot\hat{\mathbf{u}}-p'\nabla\cdot\hat{\mathbf{v}}+\hat{p}\nabla\cdot\mathbf{u}'\\&+\mathbf{1}_{\Omega^f}\frac{\nu}{2}\mathrm{D}\mathbf{u}':\mathrm{D}\hat{\mathbf{u}}+\tilde{c}_1\mathbf{1}_{\Omega^s}(\mathrm{D}\mathbf{d}'-\nabla\mathbf{d}'\nabla\mathbf{d}^T-\nabla\mathbf{d}\nabla\mathbf{d}'^T):\mathrm{D}\hat{\mathbf{u}}\Big]\\&+\tilde{c}_1\int_\Sigma[\mathbf{u}'\cdot\mathbf{n}(\mathrm{D}\mathbf{d}-\nabla\mathbf{d}\nabla\mathbf{d}^T):\mathrm{D}\hat{\mathbf{u}}]=\int_\Omega f'\hat{\mathbf{u}},\end{aligned} \tag{10}$$

because, provided Σ is smooth, $\left(\int_{\Omega_t^s}\phi\right)'=\int_{\Omega^s}\phi'+\int_\Sigma\phi\mathbf{u}'\cdot\mathbf{n}$. On this formulation we believe that it is not difficult to prove existence, uniqueness and convergence of the time scheme as in [6].

3 On the Stability of the Scheme

For clarity we drop the subscript h.

Consider the scheme based on (8) and iterated so as to replace Ω_n by Ω_{n+1} Then

$$\int_\Omega f\hat{\mathbf{u}} = \int_\Omega \Big[[\rho^n \frac{\mathbf{u}^{n+1} - \mathbf{u}^n \circ X^n}{\delta t} - p^{n+1}\nabla\cdot\hat{\mathbf{u}} - \hat{p}\nabla\cdot\mathbf{u}^{n+1}] : \hat{\mathbf{u}} \\ + \mathbf{1}_{\Omega^f_{n+1}} \frac{\nu}{2}\mathrm{D}\mathbf{u}^{n+1} : \mathrm{D}\hat{\mathbf{u}} + \tilde{c}_1 \mathbf{1}_{\Omega^s_{n+1}} \mathbf{F}^{n+1}\mathbf{F}^{n+1^T} : \mathrm{D}\hat{\mathbf{u}}\Big] \tag{11}$$

Now suppose for clarity that $\rho^f = \rho^s$ and $\mathbf{f} = 0$; the general case will be analyzed later. Choosing $\hat{p} = -p^{n+1}$, $\hat{\mathbf{u}} = \mathbf{u}^{n+1}$ and assuming Ω independent of t,

$$0 = \int_\Omega \Big[\rho \frac{1}{\delta t}(|\mathbf{u}^{n+1}|^2 - \mathbf{u}^n \circ X^n \cdot \mathbf{u}^{n+1}) + \mathbf{1}_{\Omega^f_{n+1}} \frac{\nu}{2}|\mathrm{D}\mathbf{u}^{n+1}|^2 \\ + \frac{\tilde{c}_1}{\delta t}\mathbf{1}_{\Omega^s_{n+1}} \mathbf{F}^{n+1}\mathbf{F}^{n+1^T} : \mathrm{D}(\mathbf{d}^{n+1} - \mathbf{d}^n \circ X^n)\Big] \tag{12}$$

Lemma 1.

$$2(\mathbf{u}^{n+1} - \mathbf{u}^n \circ X^n)\cdot\mathbf{u}^{n+1} = |\mathbf{u}^{n+1}|^2 - |\mathbf{u}^n \circ X^n|^2 + |\mathbf{u}^{n+1} - \mathbf{u}^n \circ X^n|^2 \geq |\mathbf{u}^{n+1}|^2 - |\mathbf{u}^n|^2$$

Now let us analyze the last term in (12). As $\nabla\cdot\mathbf{d} = 0$,

$$\tilde{c}_1 \int_{\Omega^s} \mathbf{F}^{n+1}\mathbf{F}^{n+1^T} : \mathrm{D}(\mathbf{d}^{n+1} - \mathbf{d}^n) = \int_{\Omega^s_0} \partial_{\mathbf{F}}\Psi(\mathbf{F}^{n+1}) : (\mathbf{F}^{n+1} - \mathbf{F}^n) \tag{13}$$

By the convexity of Ψ

$$\int_{\Omega^s_0} \Psi(\mathbf{F}^{n+1}) - \int_{\Omega^s_0} \Psi(\mathbf{F}^n) \leq \int_{\Omega^s_0} \partial_{\mathbf{F}}\Psi(\mathbf{F}^{n+1}) : (\mathbf{F}^{n+1} - \mathbf{F}^n)$$

Finally

$$\frac{1}{2}\|\mathbf{u}^{n+1}\|^2_\Omega + \frac{\nu\delta t}{2}\|\mathrm{D}\mathbf{u}^{n+1}\|^2_{\Omega^f} + \int_{\Omega^s_0}\Psi(\mathbf{F}^{n+1}) \leq \frac{1}{2}\|\mathbf{u}^n\|^2_\Omega + \int_{\Omega^s_0}\Psi(\mathbf{F}^n) \tag{14}$$

Case : $\rho^s \neq \rho^f$

Consider Ω^r_t, $r = s$ or f and

$$\mathscr{N}(u,\hat{u}) := \int_{\Omega^r_t} \rho^r[\partial_t\mathbf{u} + \mathbf{u}\cdot\nabla\mathbf{u}]\cdot\hat{\mathbf{u}} \approx \rho^r \int_{\Omega^r_n}[\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\delta t} + \mathbf{u}^{n+1}\cdot\nabla\mathbf{u}^{n+1}]\cdot\hat{\mathbf{u}} \tag{15}$$

Let $\hat{\mathbf{u}} = \mathbf{u}^{n+1}$, recall that $\int_{\Omega_n^r}(|\mathbf{u}^{n+1}|^2 - \mathbf{u}^n\cdot\mathbf{u}^{n+1}) \geq \int_{\Omega_n^r}\frac{1}{2}(|\mathbf{u}^{n+1}|^2 - |\mathbf{u}^n|^2)$. Now when $\nabla\cdot\mathbf{u}^{n+1} = 0$,

$$2\int_{\Omega_n^r}\mathbf{u}^{n+1}\cdot\nabla\mathbf{u}^{n+1}\cdot\mathbf{u}^{n+1} = \int_{\Omega_n^r}\mathbf{u}^{n+1}\cdot\nabla|\mathbf{u}^{n+1}|^2 = -\int_{\Omega_n^r}\nabla\cdot(\mathbf{u}^{n+1})|\mathbf{u}^{n+1}|^2$$
$$+\int_{\partial\Omega_n^r}|\mathbf{u}^{n+1}|^2\mathbf{u}^{n+1}\cdot\mathbf{n} = \frac{1}{\delta t}\Big[\int_{\Omega_{n+1}^r}|\mathbf{u}^{n+1}|^2 - \int_{\Omega_n^r}|\mathbf{u}^{n+1}|^2\Big] + o(\delta t)$$

Hence $\mathscr{N}(\mathbf{u}^{n+1},\mathbf{u}^{n+1}) \geq \frac{1}{2\delta t}\Big[\int_{\Omega_{n+1}^r}\rho^r\mathbf{u}^{n+1}|^2 - \int_{\Omega_n^r}\rho^r|\mathbf{u}^n|^2\Big]$. Consequently

$$\int_{\Omega_n}\rho[\frac{\mathbf{u}^{n+1}-\mathbf{u}^n}{\delta t} + \mathbf{u}^{n+1}\cdot\nabla\mathbf{u}^{n+1}]\cdot\mathbf{u}^{n+1} \geq \frac{1}{2\delta t}\Big[\int_{\Omega_{n+1}}\rho^{n+1}\mathbf{u}^{n+1}|^2 - \int_{\Omega_n}\rho^n|\mathbf{u}^n|^2\Big]$$

4 Formulation in the initial domain for the Solid

We wish to compare the formulation in the moving domain with the formulation in the fixed domain for a single hyperelastic incompressible structure. Recall that the deformation map satisfies

$$\partial_{tt}\mathbf{X} - \nabla_{x^0}\cdot((-p^s\mathbf{I} + \partial_{\mathbf{F}}\Psi\mathbf{F}^T)J\mathbf{F}^{-T}) = \tilde{\mathbf{f}} := \frac{1}{\rho^s}\mathbf{f}, \quad J := \det\nabla_{x^0}\mathbf{X} = 1,$$
$$\partial_{\mathbf{F}}\Psi\mathbf{F}^T = 2\tilde{c}_1\mathbf{F}\mathbf{F}^T + \alpha\mathbf{I} = 2\tilde{c}_1(\mathrm{D}_{x^0}\mathbf{d} + \nabla_{x^0}\mathbf{d}^T\nabla_{x^0}\mathbf{d}) + \tilde{\alpha}\mathbf{I}, \;\; \partial_{\mathbf{F}}\Psi = 2\tilde{c}_1\nabla\mathbf{d}^T + \bar{\alpha}\mathbf{F}^{-T}$$

Integrated on the initial domain and assuming no additional surface constraint on $\partial\Omega_0$, the variational formulation is:

$$\int_{\Omega_0^s}\Big[(\partial_{tt}\mathbf{d})\cdot\hat{\mathbf{d}} + (-pJ\mathbf{I}\mathbf{F}^{-T} + 2\tilde{c}_1\nabla_{x^0}\mathbf{d} : \nabla_{x^0}\hat{\mathbf{d}} + (J-1)\hat{p}\Big] = \int_{\Omega_0^s}\tilde{\mathbf{f}}\cdot\hat{\mathbf{d}}.$$

for all $\hat{\mathbf{d}}$ zero on Γ. A fully implicit numerical scheme is[27][14]:

$$\int_{\Omega_0}\Big[\frac{\mathbf{d}^{n+1} - 2\mathbf{d}^n + \mathbf{d}^{n-1}}{\delta t^2}\cdot\hat{\mathbf{d}} + 2c_1\nabla_{x^0}\tilde{\mathbf{d}} : \nabla_{x^0}\hat{\mathbf{d}}$$
$$-p^{n+1}J^{n+1}\mathbf{F}^{n+1^{-T}} : \nabla_{x^0}\hat{\mathbf{d}} + (J^{n+1}-1)\hat{p}\Big] = \int_{\Omega_0}\tilde{\mathbf{f}}\cdot\hat{\mathbf{d}}$$

with $\tilde{\mathbf{d}} = \frac{1}{2}(\theta\mathbf{d}^{n+1} + (2-\theta)\mathbf{d}^{n-1})$ and $\theta = 1$ or 2 and with $\mathbf{d}^{n+1} = \mathbf{d}^n + \delta t\mathbf{u}^{n+1}$. Neglecting the terms in δt^2, leads to

$$\int_{\Omega_0}\Big[\frac{\mathbf{u}^{n+1}-\mathbf{u}^n}{\delta t}\cdot\hat{\mathbf{u}} + \delta t c_1\nabla_{x^0}(\theta\mathbf{u}^{n+1} - (2-\theta)\mathbf{u}^n) : \nabla_{x^0}\hat{\mathbf{u}} + 2c_1\nabla_{x^0}\mathbf{d}^n : \nabla_{x^0}\hat{\mathbf{u}}$$
$$-p^{n+1}[\mathscr{F}(1,\mathbf{d}^n) + \delta t\mathscr{F}(0,\mathbf{u}^{n+1})] : \nabla_{x^0}\hat{\mathbf{u}} + (\det_{\mathscr{F}(1,\mathbf{d}^n)} - 1)\hat{p}$$
$$+\delta t(\partial_{x^0}d_1\partial_{y^0}u_2 - \partial_{x^0}d_2\partial_{y^0}u_1 + \partial_{x^0}d_1\partial_{y^0}u_2 - \partial_{x^0}u_2\partial_{y^0}d_1 + \partial_{x^0}u_1 + \partial_{y^0}u_2)^{n+1}\hat{p}\Big]$$

$$= \int_{\Omega_0} \tilde{\mathbf{f}}^n \cdot \hat{\mathbf{u}} \qquad\qquad \text{where } \mathscr{F}(\alpha, \mathbf{d}) = \begin{pmatrix} \alpha + \partial_{x^0} d_1, -\partial_{x^0} d_2 \\ -\partial_{y^0} d_1, \alpha + \partial_{y^0} d_2 \end{pmatrix}$$

Remark 2. To be consistent with the formulation in the moving domain one ought to have written $\nabla \cdot u = 0$, instead of $J = 1$, i.e. $(F^{-T}\nabla_{x^0}) \cdot \mathbf{u} = 0$ or equivalently $J(F^{-T}\nabla_{x^0}) \cdot \mathbf{u} = 0$, i.e.

$$\int_{\Omega_0} \left[\hat{p}\mathscr{F}(1, \mathbf{d}^n) : \nabla_{x^0} \mathbf{u}^{n+1} \right] = 0, \ \ \forall \hat{p},$$

leading to the formulation

$$\int_{\Omega_0} \Big[\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\delta t} \cdot \hat{\mathbf{u}} + \delta t c_1 \nabla_{x^0} (\theta \mathbf{u}^{n+1} - (2-\theta)\mathbf{u}^n) : \nabla_{x^0} \hat{\mathbf{u}} + 2c_1 \nabla_{x^0} \mathbf{d}^n : \nabla_{x^0} \hat{\mathbf{u}}$$
$$-p^{n+1}\mathscr{F}(1, \mathbf{d}^n) : \nabla_{x^0} \mathbf{u}^{n+1} - \hat{p}\mathscr{F}(1, \mathbf{d}^n) : \nabla_{x^0} \mathbf{u}^{n+1} \Big] = \int_{\Omega_0} \tilde{\mathbf{f}}^n \cdot \hat{\mathbf{u}} \qquad (16)$$

Remark 3. The following identity is true for all vector valued functions in $H^1_0(\Omega)^2$ (see Costabel et al[9]):

$$\int_\Omega \nabla u : \nabla v = \int_\Omega [\frac{1}{2}(\nabla u + \nabla u^T) : (\nabla v + \nabla v^T) - \nabla \cdot u \nabla \cdot v] \qquad (17)$$

It changes the boundary condition on $\partial\Omega\backslash\Gamma$ to use it in (16), ii.e.

$$2c_1 \nabla_{x^0} \mathbf{d}^n : \nabla_{x^0} \hat{\mathbf{u}} \quad \text{changed to} \quad \mathrm{D}_{x^0} \mathbf{d}^n : \mathrm{D}_{x^0} \hat{\mathbf{u}} - 2c_1 \nabla_{x^0} \cdot \mathbf{d}^n \nabla_{x^0} \cdot \hat{\mathbf{u}}$$

leads to a new boundary condition. The Mooney-Rivlin model may be degenerate in 2D for incompressible material or the boundary condition implied by this formulation may be more appropriate. This point needs to be studied further.

5 Numerical Tests

5.1 Comparisons for an Incompressible Beam

The monolithic method set in the moving domain is compared with the more classical method (16) set in the initial domain in the case of a single rectangular elastic beam of length to width ratio equal to 10 and bent by its own weight from rest and clamped either n both side or on one side.

The spatial discretization is performed by using Lagrangian Triangular Finite Elements with polynomials of degree 2 for the velocities and displacements and degree 1 for the pressures. `FreeFem++`[13] has been used to implement the algorithms. All linear systems are solved by direct solvers of the library`MUMPS`.

The method in the fixed initial domain did not seem to work with (16) but it did with (17); we have used the second order approximation, $\theta = 1$.

The penalization parameter is $\varepsilon = 0.01$ for the formulation in the initial domain and 10^{-6} for the one in the moving domain.

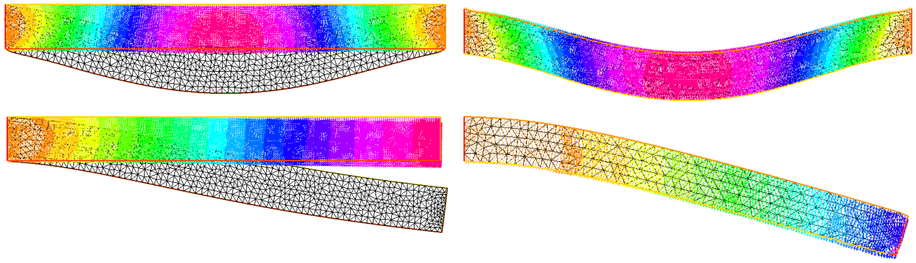


Fig. 1 *Top: Maximum bent under its own weight for the formulation in the initial domain (left) and for the formulation in the moving domain (right). Bottom: Free fall under its own weight of the same solid clamped on the left only; position at time 50 computed in the initial domain (left) and the moving domain (right). For the full swing see figure 4.*

The other parameters are

$$E = 2.15,\ \sigma = 0.29,\ \mu = \frac{E}{2(1+\sigma)},\ \rho^s = 1,\ c_1 = \frac{\mu}{2} = 0.417,\ T = 45,\ \delta t = 1.$$

The gravity differs :

1. when the beam is clamped at both sides, it is set to $f = -0.02$,
2. and when the beam is clamped on the left only $f = -0.002$.

The beam clamped at both side went through one and a half oscillation cycle during the 45 time steps while the one clamped on the left only went through a half cycle. With the method using the initial domain there is a slight change of surface area, going from 10 initially to 10.57 after 45 time iterations while the change is less that 1% for the other one.

5.2 *Fall of an Hyperelastic Beam Clamped on One Side*

The same beam clamped on one side only is allowed to fall under its own weight for 200 iterations. The results obtained by the method set in the moving domain are shown on figure 2. All parameters have the same value except the time step which is 0.5.

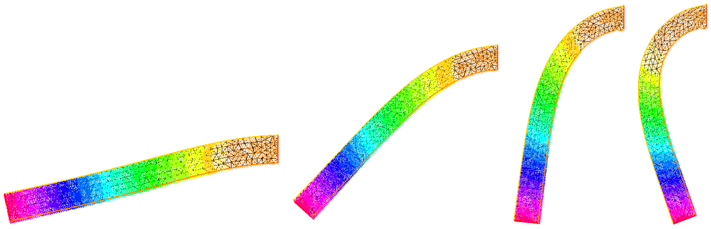


Fig. 2 *Free fall under its own weight of the same hyperelastic solid clamped on the left only; position at iteration 50, 100, 150, 200.*

6 Monolithic Fluid-Structure Interaction

Now the fall of a similar hyperelastic incompressible beam is studied in a rectangular box filled with a fluid. As the beam is clamped on the right and free to fall under its own weight in so doing it compels the fluid to move. The results are shown on figure 4. The beam is a rectangle 8×1 initially. Its Mooney-Rivlin coefficient is $c_1 = 0.2$. Its density is 1 and the gravity force is -0.2. The fluid has $\rho^f = 0.5$ and $\nu = 0.1$. The computation stops at$T = 30$ after 300 time steps. In the fluid part an auxiliary Laplace equation is solved to move the inner vertices at each time step:

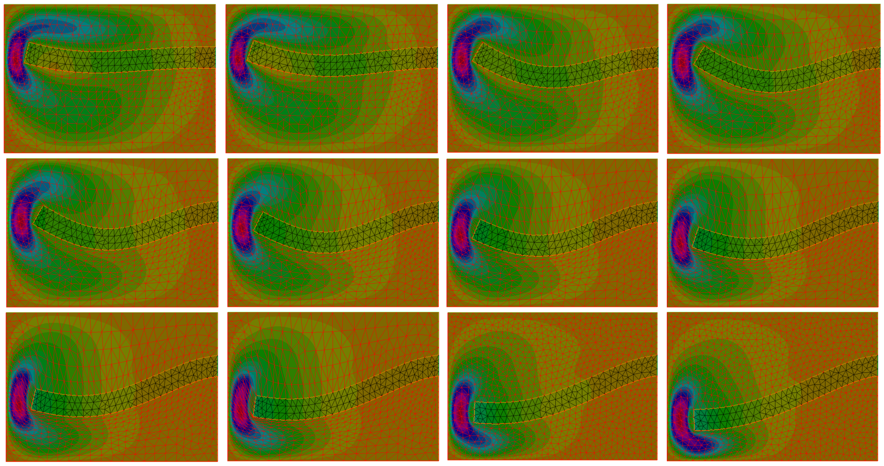


Fig. 3 *Free fall of the same beam, clamped on the right, in a fluid initially at rest. Every 10^{th} steps the geometry and the norm of velocities are shown. At the 110^{th} time steps the mesh is unusable, so `adaptmesh()` is called.*

$$-\Delta \mathbf{v} = 0, \text{ in } \Omega_t^f \;\; \mathbf{v}_{|\Sigma} = \mathbf{u}, \;\; \mathbf{v}_{|\partial\Omega\setminus\Sigma} = 0.$$

The vertices of the mesh in the solid part are moved by $\mathbf{u}^{n+1}\delta t$ by calling `movemesh()`, the mesh moving function of `FreeFem++`.

The mesh is moved by calling the `FreeFem++` function `movemesh()`. If it flips over a triangle the function `adaptmesh()` is called.

The second example is the free motion of an hyperelastic incompressible solid submitted to its own weight and to the force due to the rotation of the fluid induced by the sliding of the lower horizontal boundary at unit speed. Initially the solid is a disk. The fluid domain is a rectangle of size 10×7 and $\nu = 0.1$, $\rho^f = 0.5$ while $\rho^s = 1$. The gravity is -0.2.

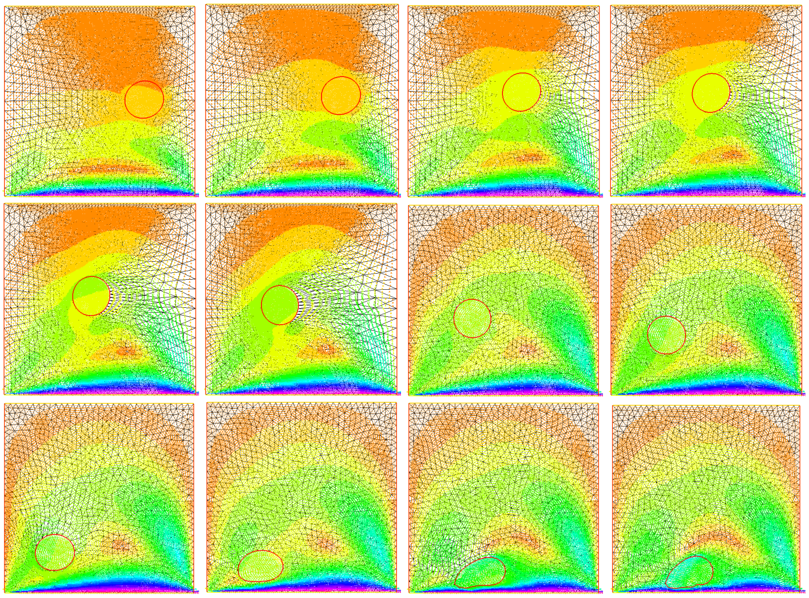
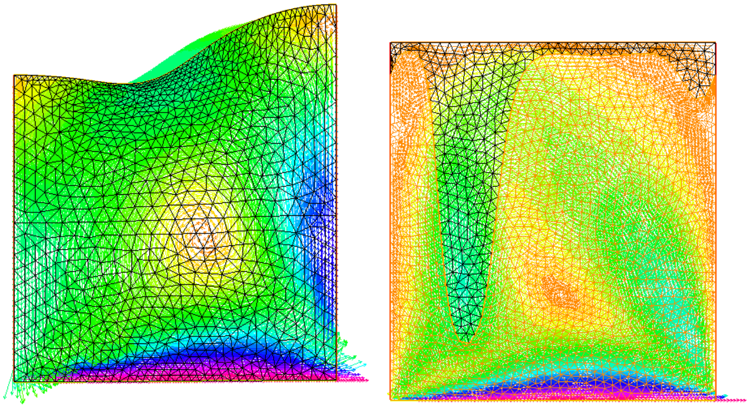


Fig. 4 *Free motion of an hyperelastic incompressible solid submitted to its own weight and to the force due to the rotation of the fluid induced by the sliding lower horizontal boundary.*

7 Free Surface Flow with the Same Code

A rotating fluid driven by the lower boundary has a free surface on top. It is subject to its own weight and it is allowed to slip on the two vertical boundaries. Results are shown on the left in figure 5. The following data have been used:



T=100, T=100

Fig. 5 *A free boundary problem (left) and a two fluid problem where the fluid above is twice heavier than the one below. Both problems are solved by the same Eulerian algorithm*

$$\rho^1 = 1,\ f = -0.1\rho,\ \delta t = 1.,\ \text{Bottom velocity } 1,\ \nu = 0.1.$$

With the same data two layers of fluids with different densities, the top one having $\rho^2 = 2$, rotate under the action of the sliding lower boundary. No-slip conditions is applied to the vertical walls. Here when `movemesh()` flip over a triangle, it is detected with `checkmovemesh()` and then a call to `adaptmesh(th,h,IsMetric=1)` rebuild the mesh `th` with mesh size average h.

Once more we stress the fact that all cases have been computed with the same computer program – given in appendix for the case of figure 4 – without modification of the core algorithm.

Conclusion

A fully Eulerian fluid-structure formulation has been presented and an attempt at deriving an implicit unconditionally stable monolithic finite element discretization has been proposed and studied. The method has been implemented with `FreeFem++`. It is reasonably robust but needs to be made unconditionally stable by implicit iterations on the moving domain, a path currently under investigation.

Acknowledgement

The author thanks Frédéric Hecht and Patrice Hauret for very valuable discussions and comments.

References

1. K.J. BATHE. Finite element procedures, Prentice-Hall, Englewood Cliffs, New-Jersey. 1996.
2. K.J. BATHE, E. RAMM AND E. L. WILSON. Finite Element Formulations for Large Deformation Dynamic Analysis. Int. J. Numer. Methods Eng., 9(2) :353-386, 1975.
3. D. BOFFI, N. CAVALLINI, L. GASTALDI, The finite element immersed boundary method with distributed lagrange multiplier, arXiv:1407.5184v2, 2015 (to appear)
4. K. BOUKIR, Y. MADAY, B. METIVET. A high order characteristics method for the incompressible Navier-Stokesequations. Comp. Methods in Applied Mathematics and Engineering 116 (1994), 211-218.
5. M. BUKACA, S. CANIC, R. GLOWINSKI, J. TAMBACAC, A. QUAINIA. Fluidstructure interaction in blood flow capturing non-zero longitudinal structure displacement. Journal of Computational Physics 235 (2013) 515541.
6. T. CHACON-REBOLLO, V. GIRAULT, F. MURAT AND O. PIRONNEAU. Analysis of a Coupled Fluid-Structure Model with Applications to Hemodynamics. SIAM J. Numer. Anal. (to appear).
7. D. CHAPELLE AND K. J. BATHE. The finite element analysis of shells-fundamentals, second edn, Springer-Verlag. 2011.
8. P.G. CIARLET. Mathematical Elasticity. North Holland, 1988.
9. M. COSTABEL, M. DAUGE. Singularities of electromagnetic fields in polyhedral domains. Preprint 97-19, Université de Rennes 1 (1997). http://www.maths.univ-rennes1.fr/∼dauge/
10. G.H. COTTET, E. MAITRE, T. MILCENT. Eulerian formulation and level set models for incompressible fluid-structure interaction. M2AN Math. Model. Numer. Anal. 42 (2008), no. 3, 471492.
11. TH. COUPEZ, L. SILVA, E. HACHEM, Implicit Boundary and Adaptive Anisotropic Meshes, S. Peretto, L. Formaggia eds. *New challenges in Grid Generation and Adaptivity for Scientific Computing*. SEMA-SIMAI Springer series, vol 5, 2015.
12. M. A. FERNANDEZ, J. MULLAERT, M. VIDRASCU. Explicit Robin-Neumann schemes for the coupling of incompressible fluids with thin-walled structures, Comp. Methods in Applied Mech. and Engg. 267, 566593, 2013.
13. F. HECHT New development in FreeFem++, J. Numer. Math., 20 (2012), pp. 251-265. www.FreeFem.org.
14. P. HAURET. Méthodes numériques pour la dynamique des structures non-linéaires incompressibles à deux échelles. Doctoral thesis, Ecole Polytechnique, 2004.
15. C. KANE, J.E. MARSDEN, M. ORTIZ, AND M. WEST. Variational integrators and the Newmark algorithm for conservative and dissipative mechanical systems. International Journal for Numerical Methods in Engineering, 49 :1295-1325, 2000.
16. S. LÉGER. Méthode lagrangienne actualisée pour des problèmes hyperélastiques en très grandes déformations. Thèse de doctorat, Université Laval, 2014.
17. P. LE TALLEC AND P. HAURET. Energy conservation in fluid-structure interactions. In P. Neittanmaki Y. Kuznetsov and O. Pironneau, editors, Numerical methods for scientific computing, variational problems and applications, CIMNE, Barcelona, 2003.
18. P. LE TALLEC AND J. MOURO. Fluid structure interaction with large structural displacements. Comp. Meth. Appl. Mech. Eng., 190(24-25) :3039-3068, 2001.

19. J. Liu. A second-order changing-connectivity ALE scheme and its application to FSI with large convection of fluids and near-contact of structures. Submitted to Journal of Computational Physics, (2015).
20. I-Shih Liu, R. Cipolatti and M. A. Rincon. Incremental linear approximation for finite elasticity. Proc ICNAAM 2006, WIley.
21. B. Lucquin and O. Pironneau. Introduction to Scientific Computing. Wiley, 1996.
22. J. Marsden and T.J.R. Hughes Mathematical Foundations of Elasticity. Dover publications 1993.
23. F. Nobile, C. Vergara. An effective fluid-structure interaction formulation for vascular dynamics by generalized robin conditions. *SIAM J. Sci. Comp.*, 30(2):731–763, 2008.
24. C.S. Peskin. The immersed boundary method. Acta Numerica, 11:479-517, 2002.
25. O. Pironneau. *Finite Element Methods for Fluids*. Wiley, 1989.
26. L. Formaggia, A. Quarteroni, A. Veneziani. *Cardiovasuclar Mathematics*. Springer MS&A Series. Springer-Verlag, 2009.
27. J. Hron and S. Turek. A monolithic fem solver for an ALE formulation of fluidstructure interaction with configuration for numerical benchmarking. European Conference on Computational Fluid Dynamics ECCOMAS CFD 2006 P. Wesseling, E. Onate and J. Periaux (Eds). TU Delft, The Netherlands (2006)
28. T. Yamada and F. Kikuchi. An arbitrary Lagrangian-Eulerian finite element method for incompressible hyperelasticity. Comput. Methods Appl. Mech. Engrg., 102(2):149-177, 1993.

8 Appendix: the FreeFem++ script

```
// FSI with same variable for fluid and structure
border a(t=10,3)  { x=0; y=t;};  //  left
border b(t=0,10) { x=t; y=3 ;};  //  bottom
border c(t=3,7)  { x=10; y=t ;}; //  right low
border d(t=10,1) { x=t; y=7; };  //  low beam
border e(t=7,8) { x=1; y=t;  };  // left beam
border f(t=1,10) { x=t; y=8; };  // top beam
border g(t=8,10) { x=10; y=t;};  // right up
border hh(t=10,0) { x=t; y=10 ;}; //  top
border ee(t=7,8) { x=10; y=t;  };  // left beam
int m=1;
mesh th = buildmesh( a(m*30)+b(m*20)+c(m*16)+d(m*30)+e(m*5)+f(m*30)
+g(m*5)+hh(m*20)+ee(m*5));
real h=0.3;
int fluid=th(1,4).region, beam=th(9,7.5).region;

fespace V2h(th,P2);
fespace Vh(th,P1);
fespace Wh(th,P1);
Vh p,ph;
V2h u=0,v=0,uh,vh,d1=0,d2=0, uold=0, vold=0,
               uu,vv, uuold=0,vvold=0;
real nu=0.1;
real E = 2.15;
real sigma = 0.29;
real mu = E/(2*(1+sigma));
real c1=2*mu/2;
real penal=1e-6;
//real lambda = E*sigma/((1+sigma)*(1-2*sigma));
real gravity = -0.2;
real rhof=0.5, rhos=1.;

macro div(u,v) ( dx(u)+dy(v) ) // EOM
macro DD(u,v)  [[2*dx(u),div(v,u)],[div(v,u),2*dy(v)]] // EOM
macro Grad(u,v)[[dx(u),dy(u)],[dx(v),dy(v)]] // EOM
Vh g11=1,g12=0,g21=0,g22=1, g11aux,g22aux,g12aux,g21aux,
    f11,f12,f21,f22;
macro G[[g11,g12],[g12,g22]]//EOM

int NN=100;
real T=300, dt=T/NN;

problem aa([u,v,p],[uh,vh,ph]) =
int2d(th,beam)( rhos*[u,v]’*[uh,vh]/dt - div(uh,vh)*p
- div(u,v)*ph+ penal*p*ph
+dt*c1*trace(DD(uh,vh)*(DD(u,v)-Grad(u,v)*Grad(d1,d2)’
- Grad(d1,d2)*Grad(u,v)’)))
   + int2d(th,beam) ( -rhos*gravity*vh +c1*trace(DD(uh,vh)*(DD(d1,d2)
- Grad(d1,d2)*Grad(d1,d2)’))
-  rhos*[uold,vold]’*[uh,vh]/dt)
  +  int2d(th,fluid)(rhof*[u,v]’*[uh,vh]/dt- div(uh,vh)*p -div(u,v)*ph
+ penal*p*ph + nu/2*trace(DD(uh,vh)'*DD(u,v)))
   -  int2d(th,fluid)(rhof*gravity*vh
+rhof*[convect([uuold,vvold],-dt,uuold),
convect([uuold,vvold],-dt,vvold)]'*[uh,vh]/dt)
 + on(1,3,7,8,9,u=0,v=0) + on(2,u=0,v=0) ;

// Computation time loop
for(int n=0;n<NN;n++){
aa;
solve bb([uu,vv],[uh,vh]) = int2d(th,fluid)(
trace(Grad(uu,vv)*Grad(uh,vh)') )
+ int2d(th,beam)(10000*[uu,vv]'*[uh,vh])
- int2d(th,beam)(10000*[u,v]'*[uh,vh])
+ on(1,2,3,7,8,uu=0,vv=0) + on(4,5,6,9,uu=u,vv=v);
  real mintcc = checkmovemesh(th,[x,y])/5.;
  real mint = checkmovemesh(th,[x+uu*dt,y+vv*dt]);
  uh=d1;
  vh=d2;
  if (mint<mintcc) {
    th=adaptmesh(th,h,IsMetric=1) ;// plot(th);
    }
  else {
th = movemesh(th,[x+uu*dt,y+vv*dt]);
  d1=0; d1[]=uh[]+dt*u[];
  d2=0; d2[]=vh[]+dt*v[];
  uold=0;  uold[]=u[];
  vold=0;  vold[]=v[];
uuold=u;
vvold=v;
  f11=1+dt*dx(uold); f12= dt*dx(vold); f21=dt*dy(uold);
  f22=1+dt*dy(vold);
     g11aux=g11*f11+g12*f21;
 g22aux=g12*f12+g22*f22;
 g12aux=g11*f12+g12*f22 ;
 g21aux=g12*f11+g12*f21 ;
     g11=f11*g11aux+f21*g21aux;
 g22=f12*g12aux+f22*g22aux;
 g12=f11*g12aux+f21*g22aux ;
  }
if((n/10)*10==n) plot(th,[uold,vold]);
vh=det(Grad(d1,d2));
cout<<n*dt<<" <- time, det d -> " << vh[].max<<
" pmax= "<<ph[].max<<" area= "<<int2d(th,beam)(1.)<<endl;
}
}
```