UiO : Department of Mathematics
University of Oslo

Monolithic solver for Computational Fluid-Structure Interaction

Sebastian Gjertsen

Master's Thesis, Spring 2017

This master's thesis is submitted under the master's programme *Computational Science and Engineering*, with programme option *Mechanics*, at the Department of Mathematics, University of Oslo. The scope of the thesis is 60 credits.

The front page depicts a section of the root system of the exceptional Lie group E_8, projected into the plane. Lie groups were invented by the Norwegian mathematician Sophus Lie (1842–1899) to express symmetries in differential equations and today they play a central role in various parts of mathematics.

Contents

Chapter 1

Fluid-Structure Interaction Problem

The following chapter will be devoted to introducing the full mathematical FSI problem. The equations and interface conditions will be introduced in a strong and weak form. From the weak form I will discretize the equations into a scheme which will be used in the FSI solver.

When computing FSI problems the computing domain is split into three parts, fluid, structure, and interface. Fluid and structure domains are separated, and different constitutive equations are solved in each domain. The spatial points in which fluid and structure join is called the interface. The treatment of the interface separates the two most commonly used methods for solving FSI problems [9]. The first method is called fully Eulerian. In a fully Eulerian framework, both the fluid and structure equations are defined and solved in a purely Eulerian description. The interface in a fully Eulerian framework is tracked across a fixed domain [21],which is a difficult task. The fully Eulerian description is suited for fluid problems but is problematic for structure problems because of the interface tracking.

The second approach is the *Arbitrary Lagrangian Eulerian* (ALE). The ALE method entails formulating the fluid equations in a type of Eulerian framework and the solid in a Lagrangian framework. The entire domain itself moves with the structural displacements and the fluid moves through these points. In the ALE framework we get the best of both worlds, in that fluid and solid are described in their most commonly stated mathematical forms.

The structure equation will remain as previously stated (**??**), and we need to take into account the convection arising from the mesh deformation.

Dealing with the movement of the domain is performed in two ways. One way is to move the domain itself in relation to the structural displacements, and use this new domain to calculate the equations for every iteration. This requires a specific function to move the mesh between each iteration. This process can be time consuming as problems get large. The structural deformation history can also give rise to problems as the points on the domain have changed location.

The second approach to ALE, which will be used in this thesis, is to calculate from **reference domain**. When solving equations from a reference frame we solve the equations on an initial, stress free domain, and use a series of mappings to account for the movements of the current time domain. It is the displacements in the in the domain that determines the value of the mappings between frames of reference. The solid equation is already stated in a Lagrangian formulation and does not need any mappings. It is the fluid velocities and fluid pressure that needs to be mapped from the reference domain into the time domain in which they are stated. Since the reference frame method does not need a function to move the mesh between each time iteration, it can be less time consuming. The interface is also located in the same position, making the interface easy to track. With the domain always remaining the same variational coupling of forces is easier when computing from a reference domain.

There are generally two different approaches when discretizing an FSI scheme. The first is a partitioned approach where fluid and structure are solved sequentially. The partitioned approach is appealing in that we have a wealth of knowledge and techniques on how to solve each of these kinds of problems in an efficient manner. The difficulty however is dealing with the interface. There are kinematic and dynamic conditions needed in FSI, and the coupling of these conditions is where problems arise (This is discussed further in discussion chapter). However, in this thesis the monolithic approach is used. In the monolithic approach all of the equations are solved at once. Artificial added mass can appear when a partitioned approach is used, arising from poorly coupling between fluid and solid velocities. The monolithic approach however, has the advantage of offering numerical stability for problems with strong added-mass effects [9], and are fully coupled. The disadvantage over

the partitioned approach is that we loose flexibility when solving many equations simultaneously, and the computing matrices can quickly become large, hence computationally costly and needing large memory. The overall the ease of implementation and numerical stability makes monolithic the preferred choice in this thesis.

The following chapter starts by introducing the mappings needed to change between current and reference domain. Lastly, the equations will be discretized following the notation and ideas from [15]:

1.1 Mapping between different frames of reference

Figure 1.1 depicts a simple Fluid-Structure Interaction domain. The fluid is surrounded by elastic walls, like for instance a blood vessel. $\hat{\mathcal{S}}$ and $\hat{\mathcal{F}}$ denotes the solid and fluid reference domain respectively. $\hat{\Sigma}$ denotes the interface in the reference domain. $\partial\hat{\mathcal{F}}_{in}$ and $\partial\hat{\mathcal{F}}_{out}$ denotes the fluid in and outflow. $\partial\hat{\mathcal{S}}$ is the outer solid wall. The reference domain is mapped using χ^s and χ^f to the time domain denoted as $\mathcal{S}$ and $\mathcal{F}$. While the interface in the time domain is denoted as Σ

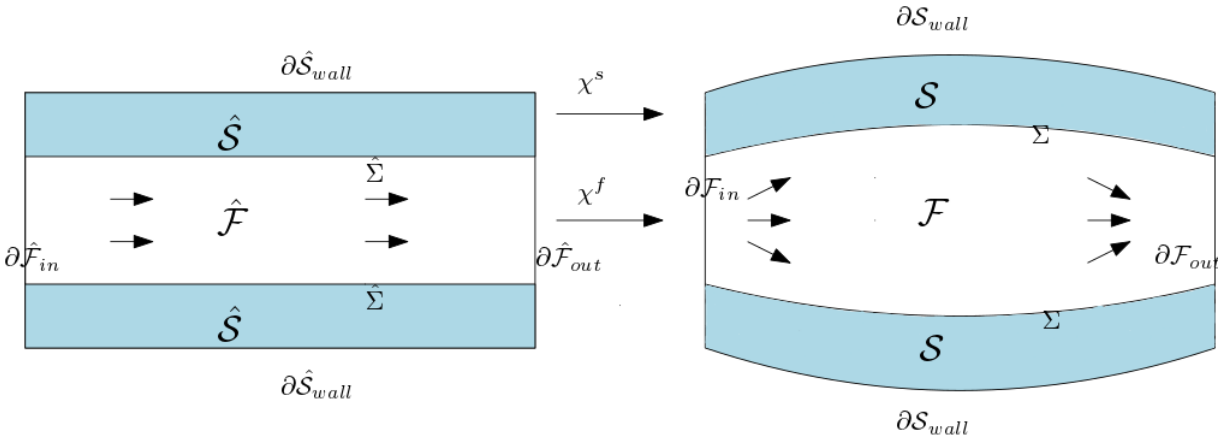


Figure 1.1: Mapping of domain from reference to current

Let $\hat{\mathcal{V}}$ be a reference domain and $\mathcal{V}(t)$ be the current time domain. Then using the deformation gradient (A.5) and the determinant of the deformation

gradient named the Jacobian (A.6), to define a mapping between the volumes from the current to the reference configurations.

The Jacobian is used to map the domain from the current to the reference domain, and the gradients acting on a vector $\mathbf{u}$ will be mapped with the deformation gradient. Lastly the divergence of a vector $\mathbf{u}$ will be mapped in a slightly different manner.

$$\int_{\mathcal{V}(t)} 1 dx = \int_{\hat{\mathcal{V}}} J dx \tag{1.1}$$

$$\int_{\mathcal{V}(t)} \nabla \mathbf{u} dx = \int_{\hat{\mathcal{V}}} J \nabla \mathbf{u} F^{-1} dx \tag{1.2}$$

$$\int_{\mathcal{V}(t)} \nabla \cdot \mathbf{u} dx = \int_{\hat{\mathcal{V}}} \nabla \cdot (J F^{-1} \mathbf{u}) dx \tag{1.3}$$

1.2 Governing equations for Fluid-Structure Interaction

The following section formulates the equations in the ALE description. The solid equation will remain in its Lagrangian description but the fluid equations will be changed in terms of the convective term and mapping between configurations. I start by briefly talking about time derivatives in the different configurations, to understand the need to change the the convective term in the fluid equation when stated in an ALE description [26]:

1.2.1 Derivatives in different frameworks

In the Lagrangian framework the total and partial derivatives have the following relation:

$$D_t f(x,t) = \partial_t f(x,t) \tag{1.4}$$

The total and partial derivatives have the following relation in the Eulerian framework, where $\mathbf{u}$ is the convecting velocity:

$$D_t f(x,t) = \mathbf{u} \cdot \nabla f + \partial_t f(x,t) \tag{1.5}$$

whilst in the ALE framework this concept is extended to take into account the motion of the domain:

$$D_t f(x,t) = \mathbf{w} \cdot \nabla f + \partial_t f(x,t) \tag{1.6}$$

Equation 1.6 shows that in the Lagrangian framework $\mathbf{w}$ is zero and for the Eulerian framework $\mathbf{w} = \mathbf{u}$.

1.2.2 Solid equation

Let $\Omega \in \hat{\mathcal{S}} \cup \hat{\mathcal{F}}$ be a global domain that consists of the fluid, structure, and the interface. The interface is defined as: $\hat{\Sigma} \in \hat{\mathcal{S}} \cap \hat{\mathcal{F}}$. Let $\mathbf{u}$ be a global velocity function that describes the fluid velocity in the fluid domain and the structure velocity in the structure domain, $\mathbf{u} = \mathbf{u_s} \cup \mathbf{u_f} \in \Omega$. The deformation function is also defined as a global function $\mathbf{d} = \mathbf{d_s} \cup \mathbf{d_f} \in \Omega$. Using global velocity and displacement functions adds the feature of the velocity and displacement being continuous across the entire domain. This is an important feature which will be apparent once the interface conditions are stated.

The solid equation will be written in terms of the solid velocity $\mathbf{u_s}$, in contrast to **??** which was defined w.r.t the deformation. Expressing the solid balance laws in the Lagrangian formulation from the initial configuration:

$$\rho_s \frac{\partial \mathbf{u}}{\partial t} = \nabla \cdot (P) + \rho_s f \quad in \quad \hat{\mathcal{S}} \tag{1.7}$$

1.2.3 Fluid equations

In the ALE description the fluid domain is moving, giving the need to redefine the velocity in the convective term in (**??**) to account for the moving domain:

$$\mathbf{u} \cdot \nabla \mathbf{u} \rightarrow (\mathbf{u} - \frac{\partial \mathbf{d}}{\partial t}) \cdot \nabla \mathbf{u} \tag{1.8}$$

Where $\mathbf{d}$ is the deformation in the fluid domain. The domain velocity $\mathbf{w} = \frac{\partial \mathbf{d}}{\partial t}$ is defined w.r.t deformation as the partial time derivative.

Applying the mappings from (1.1)(1.2) and (1.3) and including the mesh velocity in the convecting velocity as shown in (1.6), we end up with the fluid equations mapped from the time domain to the reference domain:

$$\int_{\mathcal{V}(t)} \rho_f \frac{\partial \mathbf{u}}{\partial t} = \int_{\hat{\mathcal{V}}} \rho_f J \frac{\partial \mathbf{u}}{\partial t} dx \tag{1.9}$$

$$\int_{\mathcal{V}(t)} \nabla \mathbf{u}(\mathbf{u} - \frac{\partial d}{\partial t}) dx = \int_{J\hat{\mathcal{V}}} ((\nabla \mathbf{u}) F^{-1}(\mathbf{u} - \frac{\partial d}{\partial t}) dx \tag{1.10}$$

$$\int_{\mathcal{V}(t)} \nabla \cdot \mathbf{u} dx = \int_{\hat{\mathcal{V}}} \nabla \cdot (JF^{-1}\mathbf{u}) dx \tag{1.11}$$

$$\int_{\mathcal{V}(t)} \nabla \cdot \sigma_f dx = \int_{\hat{\mathcal{V}}} \nabla \cdot (JF^{-1}\hat{\sigma_f}) dx \tag{1.12}$$

$$\hat{\sigma_f} = -pI + \mu(\nabla \mathbf{u} F^{-1} + F^{-T}\mathbf{u}^T) \tag{1.13}$$

Assembling all these terms together gives the fluid equations from a reference frame:

$$\rho_f J\big(\frac{\partial \mathbf{u}}{\partial t} dx + (\nabla \mathbf{u}) F^{-1}(\mathbf{u} - \frac{\partial \mathbf{d}}{\partial t})\big) = \nabla \cdot (JF^{-1}\hat{\sigma_f}) + J\rho_f f \tag{1.14}$$

$$\nabla \cdot (JF^{-1}\mathbf{u}) = 0 \tag{1.15}$$

1.3 Interface conditions at the boundary

The fluid‘s forces on the walls causes deformation in the solid domain and vice versa. The interface is where these energies are transferred and we therefore need conditions on the interface.

The three interface conditions comes from simple physical properties and consist of [15]:

- *Kinematic condition*: $\mathbf{u}_f = \mathbf{u}_s$ *on* $\hat{\Sigma}$. The fluid and structure velocities are equal on the interface. Meaning the fluid moves with the interface at all times. Since we use a global function for $\mathbf{u}$ in both fluid and structure domains, this condition is upheld. The fluid and solid velocities are usually in different coordinate systems, the solid velocity is then not available in Eulerian Coordinates. We instead link fluid velocity at the interface by using the fact that $\mathbf{u}_s = \frac{\partial \mathbf{d}}{\partial t}$. Setting $\mathbf{u}_f = \frac{\partial \mathbf{d}}{\partial t}$ at the interface.

- *Dynamic condition*: $\sigma_f n_f = \sigma_s n_s \ on \ \hat{\Sigma}$.
  The dynamic interface condition relates to Newtons third law of action and reaction. The forces on the interface area, here written as the normal forces are balanced on the interface. The forces from the fluid are defined in the time domain and is therefore written in terms of the reference domain:

  $$J\sigma_f F^{-T} n_f = P n_s \ on \ \hat{\Sigma}$$

  The dynamic condition is a Neumann condition that belongs to both subproblems.

- *Geometrical condition*: This condition says that the fluid and structure domains do not overlap, but rather that elements connect so the functions needing to transfer force are continuos across the entire domain.

1.4 Lifting operators for domain representation and mesh quality preservation

The kinematic interface conditions states that the fluid moves with the solid, and therefore the fluid domain needs to move in accordance with the solid deformations. The deformations from the structure are extrapolated through the interface into the fluid domain using different lifting operators. The lifting operators redistributes the interior node locations to uphold the mesh quality in the fluid domain. The choice of lifting operator is important for the overall FSI problem to be calculated [27]. When large deformations occur, we need a good lifting operator to uphold the integrity of the computing domain. A poor choice may make the mesh overlap and singularities may occur. When extrapolating deformation from the solid to the fluid domain, the fluid domain itself acts as a structure, deforming according to the deformations from the structure domain.

I will in this section present different lifting operators, that act differently on the computational domain. In **??** the techniques will be tested and investigated.

1.4.1 Harmonic extension

For small to moderate deformations an harmonic extension can be used. The harmonic extension uses the Laplace equation, transporting the deformations from the solid into the fluid domain. A variable $\alpha_u > 0$ can be multiplied to the Laplace equation, to control the amount of lifting of deformations to the fluid domain.

$$-\alpha_u \nabla^2 \mathbf{d} = 0 \quad in \ \hat{\mathcal{F}} \tag{1.16}$$

$$\mathbf{d} = 0 \quad on \ \partial\hat{\mathcal{F}}/\hat{\Sigma} \tag{1.17}$$

$$\mathbf{d}_f = \mathbf{d}_s \quad on \ \hat{\Sigma} \tag{1.18}$$

When using the harmonic lifting operator the variable α_u is very important when calculating moderate deformations. For small deformations a constant can be used for α_u. But for larger deformations we need to be a bit more clever. A good strategy for choosing α_u is proposed by Wick in [27], and further discussed in [19] and [12]. This alpha gets bigger when closer to the interface:

$$\alpha_u = \frac{1}{x^q} \tag{1.19}$$

where x^q is the distance from the interface. If $q = 0$ the laplacian i recovered. When the distance becomes larger, α_u gets smaller, and vice versa. Defining α_u in this manner is a smart choice since it upholds the cell structure closer to the interface where most of the cell distortion appears.

1.4.2 Biharmonic extension

The last extension is the biharmonic extension. The biharmonic extension provides more freedom than the harmonic and linear elastic in choosing boundary conditions and choice of parameter $\alpha_u > 0$. This is because the biharmonic extension, extends the deformation in a manner that upholds the integrity of the cells even in large deformations, even with α_u as a constant. In its simplest form it is written as:

$$-\alpha_u \nabla^4 \mathbf{d} = 0 \quad in \ \hat{\mathcal{F}} \tag{1.20}$$

The biharmonic extension is calculated with a mixed formulation where we introduce a new function ω (not to be confused with the domain velocity), the function is added to the system so that we solve for 4 functions:

$$\omega = \alpha_u \nabla^2 \mathbf{d} \quad and \quad -\alpha_u \nabla^2 \omega = 0 \ in \ \hat{\mathcal{F}} \tag{1.21}$$

with the two types of boundary conditions. The first boundary conditions being:

$$\mathbf{d} = \partial_n \mathbf{d} = 0 \quad on \ \partial\hat{\mathcal{F}} \setminus \hat{\Sigma} \tag{1.22}$$

$$\mathbf{d}_f = \mathbf{d}_s \quad on \ \hat{\Sigma} \tag{1.23}$$

The second type of boundary condition imposes conditions on $\mathbf{d}$ and ω, and are written in terms of single component functions $d^{(1)}, d^{(2)}$ and $\omega^{(1)}, \omega^{(2)}$

$$\mathbf{d}^{(1)} = \partial_n \mathbf{d}^{(1)} = 0, and \quad \omega^{(1)} = \partial_n \omega^{(1)} = 0 \quad on \ \partial\hat{\mathcal{F}}_{in,out} \tag{1.24}$$

$$\mathbf{d}^{(2)} = \partial_n \mathbf{d}^{(2)} = 0, and \quad \omega^{(2)} = \partial_n \omega^{(2)} = 0 \quad on \ \partial\hat{\mathcal{F}}_{walls} \tag{1.25}$$

$$\tag{1.26}$$

Since the biharmonic extension is of fourth order character it will have a higher computational cost [15] than the harmonic or linear elastic.

1.5 Discretization of monolithic FSI equations

After introducing all the equations and boundary conditions needed to solve a FSI problem. We are ready to discretize the equations into one monolithic scheme. The equations will be discretized and solved using finite difference and finite element methods. I will introduce a so called θ-scheme which will make it easy to implement different schemes, by choosing a value for θ. I will briefly introduce the spaces needed to discretize, following the ideas and notations of [26]:

Spaces

Let $X \in \mathbb{R}^d, d \in \{1, 2\}$ be a time dependent domain we define:

$$\hat{V}_X := H^1(X), \qquad \hat{V}_X^1 := H_0^1(X) \tag{1.27}$$

H^1 indicating a Hilbert space and

$$\hat{L}_X := L^2(X), \qquad \hat{L}^0_X := L^2(X)/\mathbb{R} \tag{1.28}$$

L^2 indicating a standard Lebesque space.

The trial and test spaces for the velocity variable in the fluid domain are defined as:

$$\hat{V}^0_{f,u} := \{\hat{u} \in H^1_0(\mathcal{F}) : \hat{u}_f = \hat{u}_s \ on \ \hat{\Sigma}\} \tag{1.29}$$

The artificial displacement spaces in the moving fluid domain are defined:

$$\hat{V}^0_{f,d} := \{\hat{d} \in H^1_0(\mathcal{F}) : \hat{d}_f = \hat{d}_s \ on \ \hat{\Sigma}\} \tag{1.30}$$

$$\hat{V}^0_{f,d} := \{\hat{d} \in H^1_0(\mathcal{F}) : \hat{\psi}_f = \hat{\psi}_s \ on \ \hat{\Sigma}\} \tag{1.31}$$

Discretization

The temporal discretization is done using finite difference schemes and the spatial is treated with the finite element method. Employing a so called $\theta-$ scheme that will enables us to easily switch between schemes.

In the domain Ω and time interval [0,T]:

Find $U = \{\mathbf{u}, \mathbf{d}, p\} \in \hat{X}^0_D$ where $\hat{X}^0_D := \{\mathbf{u}^D_f + \hat{V}^0_{f,\mathbf{u}}\} \times \hat{L}_f \times \{\mathbf{d}^D_f + \hat{V}^0_{f,\hat{f}}\} \times \{\mathbf{d}^D_f + \hat{V}^0_{f,\hat{f}}\} \times \hat{L}^0_f \times \hat{L}^0_s$ such that:

$$\int_0^T A(U)(\Psi)dt = \int_0^T \hat{F}(\Psi)dt \quad \forall \Psi \in \hat{X} \tag{1.32}$$

where $\Psi = \{\phi, \psi, \gamma\}$

$\hat{X} = \hat{V}^0_{f,\mathbf{u}} \times \hat{L}_f \times \hat{V}^0_{f,\mathbf{d},\hat{\Sigma}} \times \hat{V}^0_s \times \hat{L}^0_f \times \hat{L}^0_s$

I first introduce the scheme using, for simplicity, the harmonic mesh motion. Let $\mathbf{u}$ be a global function in the entire domain instead of $\mathbf{u}_f$ in the fluid and $\mathbf{u}_s$ in the solid. Same for the test functions. This is done for ease of reading and also for the ease of implementation later.

The full monolithic FSI variational form reads:

$$
\begin{aligned}
A(U) &= (J\rho_f \partial_t \mathbf{u}, \phi) - (J(\nabla \mathbf{u})F^{-1}(\mathbf{u} - \partial_t d), \phi)_{\hat{\mathcal{F}}} && (1.33)\\
&+ (J\sigma_f F^{-T}, \nabla\phi)_{\hat{\mathcal{F}}} && (1.34)\\
&+ (\rho_s \partial_t \mathbf{u}, \phi)_{\hat{\mathcal{S}}} + \big(P, \nabla\phi\big)_{\hat{\mathcal{S}}} && (1.35)\\
&+ (\alpha_u \nabla \mathbf{u}, \nabla\psi)_{\hat{\mathcal{F}}} + (\nabla \cdot (JF^{-1}\mathbf{u}), \gamma)_{\hat{\mathcal{F}}} && (1.36)\\
&+ \delta\big((\partial_t \mathbf{d}, \psi)_{\hat{\mathcal{S}}} - (\mathbf{u}, \psi)_{\hat{\mathcal{S}}}\big) && (1.37)\\
&+ \big(J\sigma_{f,p}F^{-T}, \nabla\phi\big) && (1.38)
\end{aligned}
$$

The condition (1.37), is weighted with a δ value. This is a critical detail for the program (detailed later) to run. The weighting says in a weak manner that this condition is important for the overall program.

Introducing the *One step-θ scheme* from [26], which has the advantage of easily being changed from a backward (implicit), forward(explicit), or a Crank-Nicholson (implicit) scheme, by changing the value of θ. The Crank-Nicholson scheme is of second order, but suffers from instabilities in this monolithic scheme for certain time step values [26]. A remedy for these instabilities is to chose a Crank-Nicholson scheme that is shifted towards the implicit side. How this is done will become evident once the scheme is defined.

The variational form is defined by dividing into four categories. This may seem strenuous at first, but the need for it will become evident when implementing the θ-scheme. The four divided categories consists of: a time group A_T(with time derivatives), implicit A_I, pressure A_P and the rest A_E (stress terms and convection):

$$
\begin{aligned}
A_T(U) &= (J\rho_f \partial_t \mathbf{u}, \phi) - (J(\nabla \mathbf{u})F^{-1}(\partial_t \mathbf{d}), \phi)_{\hat{\mathcal{F}}} && (1.39)\\
&+ (\rho_s \partial_t \mathbf{u}, \phi)_{\hat{\mathcal{S}}} + (\partial_t \mathbf{d}, \psi)_{\hat{\mathcal{S}}} && (1.40)\\
A_I(U) &= (\alpha_u \nabla \mathbf{u}, \nabla\psi)_{\hat{\mathcal{F}}} + (\nabla \cdot (JF^{-1}\mathbf{u}), \gamma)_{\hat{\mathcal{F}}} && (1.41)\\
A_E(U) &= (J(\nabla u)F^{-1}\mathbf{u}, \phi)_{\hat{\mathcal{F}}} + (J\sigma_{f,u}F^{-T}, \nabla\phi)_{\hat{\mathcal{F}}} && (1.42)\\
&+ \big(P, \nabla\phi\big)_{\hat{\mathcal{S}}} - (\mathbf{u}, \psi)_{\hat{\mathcal{S}}} && (1.43)\\
A_P(U) &= \big(J\sigma_{f,p}F^{-T}, \nabla\phi\big) && (1.44)
\end{aligned}
$$

Notice that the fluid stress tensors have been split into a velocity and pressure

part.

$$\sigma_{f,u} = \mu(\nabla u F^{-1} + F^{-T}\nabla u) \tag{1.45}$$
$$\sigma_{f,p} = -pI \tag{1.46}$$

For the time group, discretization is done in the following way:

$$A_T(U^{n,k}) \approx \frac{1}{k}\big(\rho_f J^{n,\theta}(\mathbf{u}^n - \mathbf{u}^{n-1}), \phi\big)_{\hat{\mathcal{F}}} - \frac{1}{k}\big(\rho_f(\nabla u)(\mathbf{d}^n - \mathbf{d}^{n-1}), \phi\big)_{\hat{\mathcal{F}}} \tag{1.47}$$
$$+\frac{1}{k}\big(\rho_s(\mathbf{u}^n - \mathbf{u}^{n-1}), \phi\big)_{\hat{\mathcal{S}}} + \frac{1}{k}\big((\mathbf{d}^n - \mathbf{d}^{n-1}), \psi\big)_{\hat{\mathcal{S}}} \tag{1.48}$$

The Jacobian is written with superscript θ as:

$$J^{n,\theta} = \theta J^n + (1-\theta)J^{n-1} \tag{1.49}$$

Finally we can introduce the *One step-θ scheme*: Find $U^n = \{\mathbf{u}^n, \mathbf{d}^n, p^n\}$

$$A_T(U^{n,k}) + \theta A_E(U^n) + A_P(U^n) + A_I(U^n) = \tag{1.50}$$
$$-(1-\theta)A_E(U^{n-1}) + \theta\hat{f}^n + (1-\theta)\hat{f}^{n-1} \tag{1.51}$$

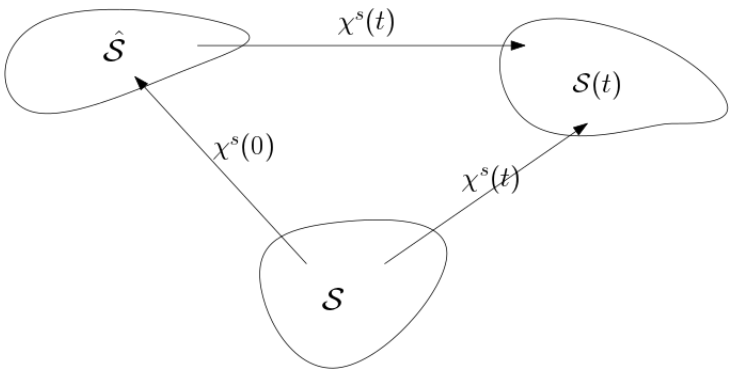
Notice that the scheme is selected by the choice of θ. By choosing $\theta = 1$ we get the back Euler scheme, for $\theta = \frac{1}{2}$ we get the Crank-Nicholson scheme and for the shifted Crank-Nicholson we set $\theta = \frac{1}{2} + k$, effectively shifting the scheme towards the implicit side. $\hat{f}$ is the body forces. The shifting of the scheme towards the implicit side is important for long term stability for certain time step values [26]. The shifting will be investigated in chaper **??**.

Appendices

Appendix A

Appendix

A.1 Lagrangian description of solid mechanics



Let $\hat{\mathcal{S}}$, $\mathcal{S}$, $\mathcal{S}(t)$ be the initial stress free configuration of a given body, the reference and the current configuration respectively. I define a smooth mapping from the reference configuration to the current configuration:

$$\chi^s(\mathbf{X}, t) : \hat{\mathcal{S}} \rightarrow \mathcal{S}(t) \tag{A.1}$$

Where $\mathbf{X}$ denotes a material point in the reference domain and χ^s denotes the mapping from the reference configuration to the current configuration. Let $d^s(\mathbf{X}, t)$ denote the displacement field which describes deformation on a body. The mapping χ^s can then be specified from a current position plus the displacement from that position:

$$\chi^s(\mathbf{X}, t) = \mathbf{X} + d^s(\mathbf{X}, t) \tag{A.2}$$

which can be written in terms of the displacement field:

$$d^s(\mathbf{X}, t) = \chi^s(\mathbf{X}, t) - \mathbf{X} \tag{A.3}$$

Let w($\mathbf{X}$,t) be the domain velocity which is the partial time derivative of the displacement:

$$w(\mathbf{X}, t) = \frac{\partial \chi^s(\mathbf{X}, t)}{\partial t} \tag{A.4}$$

A.1.1 Deformation gradient

The deformation gradient describes the rate at which a bode undergoes deformation. Let $d(\mathbf{X}, t)$ be a differentiable deformation field in a given body, the deformation gradient is then:

$$F = \frac{\partial \chi^s(\mathbf{X}, t)}{\partial \mathbf{X}} = \frac{\partial \mathbf{X} + d^s(\mathbf{X}, t)}{\partial \mathbf{X}} = I + \nabla d(\mathbf{X}, t) \tag{A.5}$$

which denotes relative change of position under deformation in a Lagrangian frame of reference. We can observe that when there is no deformation. The deformation gradient F is simply the identity matrix.

Let the Jacobian determinant, which is the determinant of the of the deformation gradient F, be defined as:

$$J = \det(F) \tag{A.6}$$

The Jacobian determinant is used to change between volumes, assuming infinitesimal line and area elements in the current ds, dx and reference dV, dX configurations. The Jacobian determinant is therefore known as a volume ratio.

A.1.2 Strain

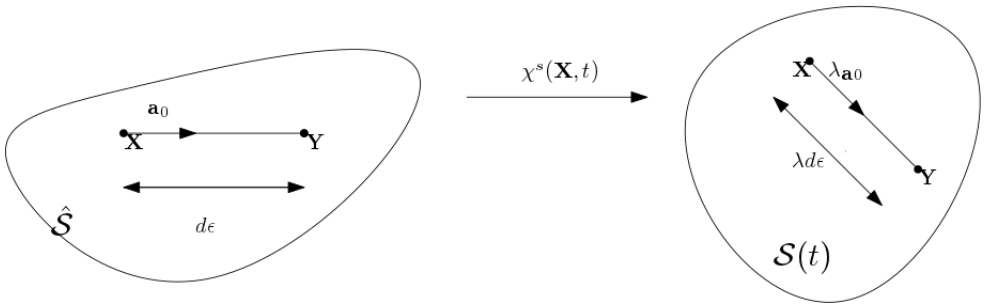


Figure A.1: Deformation of a line element with length $d\epsilon$ into a line element with length $\lambda d\epsilon$

Strain is the relative change of location between two particles. Strain, strain rate and deformation is used to describe the relative motion of particles in a continuum. This is the fundamental quality that causes stress [15].

Observing two neighboring points $\mathbf{X}$ and $\mathbf{Y}$. Let $\mathbf{Y}$ be described by adding and subtracting the point $\mathbf{X}$ and rewriting $\mathbf{Y}$ from the point $\mathbf{X}$ plus a distance $d\mathbf{X}$:

$$\mathbf{Y} = \mathbf{Y} + \mathbf{X} - \mathbf{X} = \mathbf{X} + |\mathbf{Y} - \mathbf{X}|\frac{\mathbf{Y} - \mathbf{X}}{|\mathbf{Y} - \mathbf{X}|} = \mathbf{X} + d\mathbf{X} \tag{A.7}$$

Let $d\mathbf{X}$ be denoted by:

$$d\mathbf{X} = d\epsilon\mathbf{a}_0 \tag{A.8}$$
$$d\epsilon = |\mathbf{Y} - \mathbf{X}| \tag{A.9}$$
$$\mathbf{a}_0 = \frac{\mathbf{Y} - \mathbf{X}}{|\mathbf{Y} - \mathbf{X}|} \tag{A.10}$$

where $d\epsilon$ is the distance between the two points and $\mathbf{a}_0$ is a unit vector

We see now that $d\mathbf{X}$ is the distance between the two points times the unit vector or direction from $\mathbf{X}$ to $\mathbf{Y}$.

A certain motion transform the points $\mathbf{Y}$ and $\mathbf{X}$ into the displaced positions $\mathbf{x} = \chi^s(\mathbf{X}, t)$ and $\mathbf{y} = \chi^s(\mathbf{Y}, t)$. Using Taylor?s expansion $\mathbf{y}$ can be expressed in terms of the deformation gradient:

$$\mathbf{y} = \chi^s(\mathbf{Y}, t) = \chi^s(\mathbf{X} + d\epsilon\mathbf{a}_0, t) \tag{A.11}$$
$$= \chi^s(\mathbf{X}, t) + d\epsilon F\mathbf{a}_0 + \mathcal{O}(\mathbf{Y} - \mathbf{X}) \tag{A.12}$$

where $\mathcal{O}(\mathbf{Y} - \mathbf{X})$ refers to the small error that tends to zero faster than $(\mathbf{X} - \mathbf{Y}) \to \mathcal{O}$.

Setting $\mathbf{x} = \chi^s(\mathbf{X}, t)$ It follows that:

$$\mathbf{y} - \mathbf{x} = d\epsilon F\mathbf{a}_0 + \mathcal{O}(\mathbf{Y} - \mathbf{X}) \tag{A.13}$$
$$= F(\mathbf{Y} - \mathbf{X}) + \mathcal{O}(\mathbf{Y} - \mathbf{X}) \tag{A.14}$$

Let the **stretch vector** be $\lambda_{\mathbf{a}0}$, which goes in the direction of $\mathbf{a}_0$:

$$\lambda_{\mathbf{a}0}(\mathbf{X}, t) = F(\mathbf{X}, t)\mathbf{a}_0 \tag{A.15}$$

Looking at the square of λ:

$$\lambda^2 = \lambda_{\mathbf{a}0}\lambda_{\mathbf{a}0} = F(\mathbf{X}, t)\mathbf{a}_0 F(\mathbf{X}, t)\mathbf{a}_0 \tag{A.16}$$
$$= \mathbf{a}_0 F^T F\mathbf{a}_0 = \mathbf{a}_0 C\mathbf{a}_0 \tag{A.17}$$

We have not introduced the important right Cauchy-Green tensor:

$$C = F^T F \tag{A.18}$$

Since $\mathbf{a}_0$ is just a unit vector, we see that C measures the squared length of change under deformation. We see that in order to determine the stretch one needs only the direction of $\mathbf{a}_0$ and the tensor C. C is also symmetric and positive definite $C = C^T$. I also introduce the Green-Lagrangian strain tensor E:

$$E = \frac{1}{2}(F^T F - I) \tag{A.19}$$

which is also symmetric since C and I are symmetric. The Green-Lagrangian strain tensor E has the advantage of having no contributions when there is no deformations. Where the Cauchy-Green tensor gives the identity matrix for zero deformation.

A.1.3 Stress

While strain, deformation and strain rate only describe the relative motion of particles in a given volume, stress give us the internal forces between neighboring particles. Stress is responsible for deformation and is therefore crucial in continuum mechanics. The unit of stress is force per area.

that completely define the state of stress at a point inside a material in the deformed state, placement, or configuration. The tensor relates a unit-length direction vector n to the stress vector T(n) across an imaginary surface perpendicular to n:

Introducing the Cauchy stress tensor σ_s, which define the state of stress inside a material. The version of Cauchy stress tensor is defined by the material model used. If we use this tensor on an area, taking the stress tensor times the normal vector $\sigma_s\mathbf{n}$ we get the forces acting on that area.

Stress tensor defined from the Cauchy by the constitutive law of St. Venant-Kirchhoff hyperelastic material model:

$$\sigma_s = \frac{1}{J}F(\lambda_s(trE)I + 2\mu_s E)F^T \tag{A.20}$$

Using the deformation gradient and the Jacobian determinant., I get the first Piola-Kirchhoff stress tensor P:

$$P = J\sigma F^{-T} \tag{A.21}$$

This is known as the *Piola Transformation* and maps the tensor into a Lagrangian formulation which will be used when stating the solid equation.

Introducing the second Piola-Kirchhoff stress tensor S:

$$S = JF^{-1}\sigma F^{-T} = F^{-1}P = S^T \tag{A.22}$$

from this relation the first Piola-Kirchhoff tensor can be expressed by the second:

$$P = FS \tag{A.23}$$

A.2 Results

Table 3 Results for unsteady benchmark FSI3.

	Unknowns	Δt	$u_1(A)$ [$\times 10^{-3}$]	$u_2(A)$ [$\times 10^{-3}$]	F_D	F_L
1	61318	1.0e−3	−2.54 ± 2.41	1.45 ± 32.80	450.3 ± 23.51	-0.10 ± 143.0
	237286	2.0e−3	−2.88 ± 2.73	1.53 ± 34.94	458.6 ± 27.18	2.08 ± 153.1
	237286	1.0e−3	−2.87 ± 2.73	1.54 ± 34.94	458.6 ± 27.31	2.00 ± 153.3
	237286	5.0e−4	−2.86 ± 2.72	1.53 ± 34.90	458.6 ± 27.27	2.01 ± 153.4
	941158	1.0e−3	−2.91 ± 2.77	1.47 ± 35.26	459.9 ± 27.92	1.84 ± 157.7
2a	11250	5.0e−3	−2.48 ± 2.24	1.27 ± 36.50	–	–
2b	7176	5.0e−3	−2.44 ± 2.32	1.02 ± 31.82	473.5 ± 56.97	8.08 ± 283.8
	7176	2.0e−3	−2.48 ± 2.39	0.92 ± 32.81	471.3 ± 62.28	6.11 ± 298.6
	7176	1.0e−3	−2.58 ± 2.49	0.94 ± 33.19	470.4 ± 64.02	4.65 ± 300.3
	27744	5.0e−3	−2.43 ± 2.27	1.41 ± 31.73	483.7 ± 22.31	2.21 ± 149.0
	27744	2.0e−3	−2.63 ± 2.61	1.46 ± 33.46	483.3 ± 24.48	2.08 ± 161.2
	27744	1.0e−3	−2.80 ± 2.64	1.45 ± 34.12	483.0 ± 25.67	2.21 ± 165.3
	42024	2.5e−3	−2.40 ± 2.26	1.39 ± 31.71	448.7 ± 21.16	1.84 ± 141.3
	42024	1.0e−3	−2.53 ± 2.38	1.40 ± 32.49	449.7 ± 22.24	1.61 ± 142.8
	42024	5.0e−4	−2.57 ± 2.42	1.42 ± 32.81	450.1 ± 22.49	1.49 ± 143.7
	72696	2.5e−3	−2.64 ± 2.48	1.38 ± 33.25	451.1 ± 24.57	2.04 ± 150.6
	72696	1.0e−3	−2.79 ± 2.62	1.28 ± 34.61	452.0 ± 25.78	1.91 ± 152.7
	72696	5.0e−4	−2.84 ± 2.67	1.28 ± 34.61	452.4 ± 26.19	2.36 ± 152.7
3	19488	1.0e−3	−3.02 ± 2.83	1.41 ± 35.47	458.2 ± 28.32	2.41 ± 145.6
	19488	5.0e−4	−3.02 ± 2.85	1.42 ± 35.63	458.7 ± 28.78	2.23 ± 146.0
	19488	2.5e−4	−3.02 ± 2.85	1.32 ± 35.73	458.7 ± 28.80	2.23 ± 146.0
	76672	1.0e−3	−2.78 ± 2.62	1.44 ± 34.36	459.1 ± 26.63	2.41 ± 151.3
	76672	5.0e−4	−2.78 ± 2.62	1.44 ± 34.35	459.1 ± 26.62	2.39 ± 150.7
	76672	2.5e−4	−2.77 ± 2.61	1.43 ± 34.43	459.1 ± 26.50	2.36 ± 149.9
	304128	1.0e−3	−2.86 ± 2.70	1.45 ± 34.93	460.2 ± 27.65	2.47 ± 154.9
	304128	5.0e−4	−2.86 ± 2.70	1.45 ± 34.90	460.2 ± 27.47	2.37 ± 153.8
	304128	2.5e−4	−2.88 ± 2.72	1.47 ± 34.99	460.5 ± 27.74	2.50 ± 153.9
4	81120	9.0e−5	−5.18 ± 5.04	1.12 ± 45.10	477.0 ± 48.00	7.00 ± 223.0
	324480	2.0e−5	−4.54 ± 4.34	1.50 ± 42.50	467.5 ± 39.50	16.20 ± 188.7
5	2480814	5.1e−5	−2.88 ± 2.71	1.48 ± 35.10	463.0 ± 31.30	1.81 ± 154.0
6	7059	5.0e−4	−1.60 ± 1.60	1.50 ± 25.90	525.0 ± 22.50	-0.55 ± 106.0
	27147	5.0e−4	−2.00 ± 1.89	1.45 ± 29.00	434.0 ± 17.50	2.53 ± 88.6
7	271740	5.0e−4	−3.04 ± 2.87	1.55 ± 36.63	474.9 ± 28.12	3.86 ± 165.9

Figure A.2: Results from different contributions in from the paper Turek et.al 2010 [20]

[20]

Bibliography

[1] K. Yusuf Billah. Resonance, Tacoma Narrows bridge failure, and undergraduate physics textbooks. *American Journal of Physics*, 59(2):118, 1991.

[2] David J Charlesworth. Solution of the Incompressible Navier- Stokes Equations on Unstructured Meshes by. (August), 2003.

[3] S. Étienne, A. Garon, and D. Pelletier. Some manufactured solutions for verification of fluid-structure interaction codes. *Computers and Structures*, 106-107:56–67, 2012.

[4] Stéphane Étienne, D Tremblay, and Dominique Pelletier. Code Verification and the Method of Manufactured Solutions for Fluid-Structure Interaction Problems. *36th AIAA Fluid Dynamics Conference and Exhibit*, (June):1–11, 2006.

[5] Charles L. Fefferman. Existence and smoothness of the Navier-Stokes equation. *The millennium prize problems*, (1):1–5, 2000.

[6] Miguel A. Fernández, Jimmy Mullaert, and Marina Vidrascu. Explicit robin-neumann schemes for the coupling of incompressible fluids with thin-walled structures. *Computer Methods in Applied Mechanics and Engineering*, 267:566–593, 2013.

[7] Miguel A. Fernández, Jimmy Mullaert, and Marina Vidrascu. Generalized Robin-Neumann explicit coupling schemes for incompressible fluid-structure interaction: Stability analysis and numerics. *International Journal for Numerical Methods in Engineering*, 101(3):199–229, 2015.

[8] Jaroslav Hron and Stefan Turek. Proposal for numerical benchmarking of fluid-structure interaction between an elastic object and laminar incompressible flow. *Fluid-Structure Interaction*, 53:371–385, 2006.

[9] Jie Liu, Rajeev K. Jaiman, and Pardha S. Gurugubelli. A stable second-order scheme for fluid-structure interaction with strong added-mass effects. *Journal of Computational Physics*, 270:687–710, 2014.

[10] Anders Logg, Harish Narayanan, Marie Rognes, Johannes Ring, Kristian B. Ølgaard, and Garth N. Wells. FEniCS Project, 2011.

[11] Cm Macal. Proceedings of the 2005 Winter Simulation Conference ME Kuhl, NM Steiger, FB Armstrong, and JA Joines, eds. *Simulation*, pages 1643–1649, 2005.

[12] Selim MM and Koomullil RP. Mesh Deformation Approaches – A Survey. *Journal of Physical Mathematics*, 7(2), 2016.

[13] William L. Oberkampf and Christopher J. Roy. *Verification and Validation in Scientific Computing*. Cambridge University Press, Cambridge, 2010.

[14] T Richter and T Wick. On time discretizations of Fluid-structure interactions. *Multiple Shooting and Time Domain Decomposition MEthods*, pages 377–400, 2013.

[15] Thomas Richter. Fluid Structure Interactions. 2016.

[16] Patrick J. Roache. Code Verification by the Method of Manufactured Solutions. *Journal of Fluids Engineering*, 124(1):4, 2002.

[17] Natural Sciences. A Newton ' s Method Finite Element Algorithm for Fluid-Structure Interaction. (October), 2012.

[18] Noelle Selin. Verification and Validation. (February), 2014.

[19] K. Stein, T. Tezduyar, and R. Benney. Mesh Moving Techniques for Fluid-Structure Interactions With Large Displacements. *Journal of Applied Mechanics*, 70(1):58, 2003.

[20] S Turek, J Hron, M Razzaq, H Wobker, and M Sch. Fluid Structure Interaction II. 73, 2010.

[21] Boris Valkov, Chris H Rycroft, and Ken Kamrin. Eulerian method for fluid – structure interaction and submerged solid – solid contact problems.

[22] E. H. van Brummelen. Added Mass Effects of Compressible and Incompressible Flows in Fluid-Structure Interaction. *Journal of Applied Mechanics*, 76(2):021206, 2009.

[23] V Vinje. Simulating Cerebrospinal Fluid Flow and Spinal Cord Movement Associated with Syringomyelia. 2016.

[24] Frank M White. Viscous Fluid Flow Viscous. *New York*, Second:413, 2000.

[25] Frank M. White. Chapter 3 - Solutions of the Newtonian viscous-flow equa- tions. *Viscous Fluid Flow*, (5), 2006.

[26] Thomas Wick. Adaptive Finite Element Simulation of Fluid-Structure Interaction with Application to Heart-Valve Dynamics. *Institute of Applied Mathematics, University of Heidelber*, page 157, 2011.

[27] Thomas Wick. Fluid-structure interactions using different mesh motion techniques. *Computers and Structures*, 89(13-14):1456–1467, 2011.