# Bingham fluids using X-MESH

DEGROOFF Vincent

May 4, 2023

# Contents

# Chapter 1

# Governing equations

## 1.1 General notions of continuum mechanics

Let $\Omega \in \mathbb{R}^3$ denote the domain of interest and $\Gamma = \partial\Omega$ its boundary. Let $\mathbf{u}$ be the fluid velocity, and $\boldsymbol{\sigma} = -p\mathbf{I} + \boldsymbol{\tau}$ be the stress tensor with its deviatoric contribution $\boldsymbol{\tau}$.

Let us recall the mass and momentum conservation equations, with body forces $\mathbf{f}$, expressed in an Eulerian reference frame:

$$\nabla \cdot \mathbf{u} = 0 \tag{1.1}$$

$$\rho \left( \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\, \mathbf{u} \right) = -\nabla p + \nabla \cdot \boldsymbol{\tau} + \mathbf{f} \tag{1.2}$$

Throughout this study, we will consider low Reynolds numbers, i.e. flows that are dominated by viscous forces. We will thus neglect the inertia terms on the left of equation (1.3). The resulting equations are the well known Stokes equations, where we look for velocity and pressure fields $(\mathbf{u}, p)$ solution of

$$0 = -\nabla p + \nabla \cdot \boldsymbol{\tau}(\mathbf{u}) + \mathbf{f} \qquad \text{in } \Omega \tag{1.3}$$

$$0 = \nabla \cdot \mathbf{u} \qquad \text{in } \Omega \tag{1.4}$$

$$\mathbf{u} = \mathbf{u}_D \qquad \text{on } \Gamma_D \tag{1.5}$$

$$\boldsymbol{\sigma} \cdot \hat{\mathbf{n}} = \mathbf{g} \qquad \text{on } \Gamma_N \tag{1.6}$$

where $\Gamma_D$ and $\Gamma_N$ refer to the partitions of $\partial\Omega$ where Dirichlet and Neumann boundary conditions are applied. $\mathbf{u}_D$ and $\mathbf{g}$ are given beforehand. One could also decide to impose only the normal velocity along with the tangential stress, or the tangential velocity along with the normal stress.

Before looking for a closure equation for the shear stress tensor, let us first recall the velocity gradient decomposition,

$$\nabla \mathbf{u} = \mathbf{D} + \mathbf{W} \tag{1.7}$$

$$\mathbf{D} = \frac{1}{2} \left( \nabla \mathbf{u} + \nabla^T \mathbf{u} \right) = \frac{1}{2} \dot{\boldsymbol{\gamma}} \tag{1.8}$$

$$\mathbf{W} = \frac{1}{2} \left( \nabla \mathbf{u} - \nabla^T \mathbf{u} \right) \tag{1.9}$$

To be exhaustive:

- $\mathbf{D}$ is the strain-rate tensor, whose diagonal elements indicate the stretching of the fluid along the basis vectors and whose off-diagonal elements indicate the shearing of the fluid from one basis vector to another.
- $\mathbf{W}$ is the spin tensor that indicates the axis around which the flow is locally rotating.

## 1.2 Constitutive equations of Bingham fluids

Newtonian fluid behavior is the most simple one can find in fluid mechanics because it assumes that the fluid deformation, or strain rate, is is directly proportional to the shear stress it undergoes. It is very accurate for many liquids and gases like air, water, oil, alcohol, glycerine, etc. However, this model cannot characterize a variety of well known substances from blood to toothpaste to ketchup to name a few.

Such fluids can be described as Generalized Newtonian [1]. This model extends Newtonian fluids with a nonlinear constitutive law between the stress $\tau_{ij}$ and strain rate $\dot{\gamma}_{ij}$:

$$\tau_{ij} = \mu(\dot{\gamma}, T, \phi)\dot{\gamma}_{ij} \tag{1.10}$$

where $\mu$, $\dot{\gamma}$, $T$ and $\phi$ respectively refer to the fluid viscosity, the scalar strain rate, the temperature and the particle concentration (e.g. polymer chains in dilute solutions).

The simplest example are the Power-law Fluids, that assume $\mu(\dot{\gamma}) = K\dot{\gamma}^{n-1}$, with the consistency $K$ and index $n$. This can already describe

- *Shear thinning* fluids with viscosity inversely proportional to strain rate ($n < 1$). For example, Ketchup or blood flow much more easily with higher strain rates.
- *Shear thickening* or *Dilatant* fluids with viscosity proportional to strain rate ($n > 1$). A typical example is the mixture of cornstarch and water, called Oobleck, that appears liquid at rest, but that becomes viscous enough to walk on it if struck hard enough.
- *Newtonian* fluids of course when $n = 1$.

A little bit more general is the Herschel-Bulkley Model that extends the Power-law model with a yield stress $\tau_0$ under which the fluid does not deform: it is *unyielded*, $\dot{\gamma} = 0$. Above that threshold, the fluid becomes *yielded* and follows the Power-law model. Bingham fluids correspond to the special case of $n = 1$. The constitutive law of the Herschel-Bulkley Model reads as follows

$$\begin{aligned} &\dot{\gamma}_{ij} = 0 && \text{if} \quad \tau \leq \tau_0 \\ &\tau_{ij} = \left(K\dot{\gamma}^{n-1} + \frac{\tau_0}{\dot{\gamma}}\right)\dot{\gamma}_{ij} && \text{if} \quad \tau \geq \tau_0 \end{aligned} \tag{1.11}$$

where $\dot{\gamma}$ and $\tau$ are the matrix norms of $\dot{\boldsymbol{\gamma}}$ and $\boldsymbol{\tau}$, more precisely the square root of the second invariant of traceless tensors:

$$I_2(\mathbf{A}) := \frac{1}{2}\left[\mathrm{Tr}\big(\mathbf{A}^2\big) - \big(\mathrm{Tr}(\mathbf{A})\big)^2\right] \quad \implies \quad \begin{aligned} \dot{\gamma} &:= \sqrt{I_2(\dot{\boldsymbol{\gamma}})} = \sqrt{\frac{1}{2}\dot{\boldsymbol{\gamma}} : \dot{\boldsymbol{\gamma}}} = \sqrt{2\mathbf{D} : \mathbf{D}} \\ \tau &:= \sqrt{I_2(\boldsymbol{\tau})} = \sqrt{\frac{1}{2}\boldsymbol{\tau} : \boldsymbol{\tau}} \end{aligned} \tag{1.12}$$

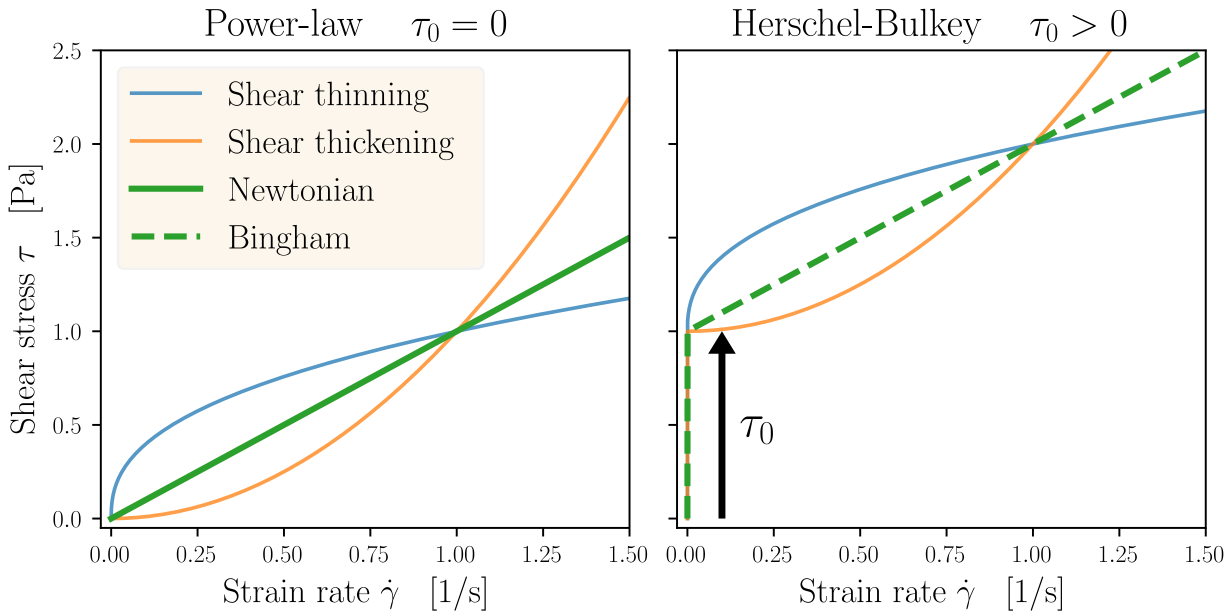


**Figure 1.1.** Non-Newtonian fluid models.

Since we will focus on Bingham fluids in this study, let us reformulate model (1.11) for $n = 1$, with a tensor notation:

$$\|\boldsymbol{\tau}\| < \tau_0 \qquad \text{if} \quad \dot{\boldsymbol{\gamma}} = \mathbf{0} \tag{1.13}$$

$$\boldsymbol{\tau} = \left(\mu + \frac{\tau_0}{\dot{\gamma}}\right)\dot{\boldsymbol{\gamma}} \qquad \text{if} \quad \dot{\boldsymbol{\gamma}} \neq \mathbf{0} \tag{1.14}$$

## 1.3 Weak formulation

In order to solve equations (1.3) and (1.4) with the finite element method, one can express them under its weak form. This is done by multiplying the equations of the strong form by velocity and pressure test functions.

$$\begin{aligned}
0 &= -\nabla p + \nabla \cdot \boldsymbol{\tau}(\mathbf{u}) + \mathbf{f} \\
&= \int_\Omega -\mathbf{v} \cdot \nabla p \,\mathrm{d}x + \int_\Omega \mathbf{v} \cdot (\nabla \cdot \boldsymbol{\tau}) \,\mathrm{d}x + \int_\Omega \mathbf{f} \cdot \mathbf{v} \,\mathrm{d}x \\
&= \int_\Omega p \nabla \cdot \mathbf{v} - \int_\Omega \nabla \cdot (p\mathbf{v}) + \int_\Omega \nabla \cdot (\boldsymbol{\tau} \cdot \mathbf{v}) - \int_\Omega \boldsymbol{\tau}(\mathbf{u}) : \nabla^\top \mathbf{v} + \int_\Omega \mathbf{f} \cdot \mathbf{v} \\
&= \int_\Omega p \nabla \cdot \mathbf{v} - \int_{\partial\Omega} p\mathbf{v} \cdot \hat{\mathbf{n}} + \int_{\partial\Omega} (\boldsymbol{\tau} \cdot \mathbf{v}) \cdot \hat{\mathbf{n}} - \int_\Omega \boldsymbol{\tau}(\mathbf{u}) : \left(\mathbf{D}(\mathbf{v}) + \boldsymbol{\Omega}(\mathbf{v})\right)^\top + \int_\Omega \mathbf{f} \cdot \mathbf{v} \\
&= \int_\Omega p \nabla \cdot \mathbf{v} \,\mathrm{d}x - \int_\Omega \boldsymbol{\tau}(\mathbf{u}) : \mathbf{D}(\mathbf{v}) \,\mathrm{d}x + \int_\Omega \mathbf{f} \cdot \mathbf{v} \,\mathrm{d}x + \int_{\Gamma_N} \mathbf{g} \cdot \mathbf{v} \,\mathrm{d}s \\
&= \int_\Omega p \nabla \cdot \mathbf{v} \,\mathrm{d}x - \int_\Omega \left(2\mu + \frac{\tau_0}{\|\mathbf{D}(\mathbf{u})\|}\right) \mathbf{D}(\mathbf{u}) : \mathbf{D}(\mathbf{v}) \,\mathrm{d}x + \int_\Omega \mathbf{f} \cdot \mathbf{v} \,\mathrm{d}x \int_{\Gamma_N} \mathbf{g} \cdot \mathbf{v} \,\mathrm{d}s
\end{aligned} \tag{1.15}$$

that must hold $\forall\ \mathbf{v} \in H^1(\Omega)^d$ such that $\mathbf{v} = \mathbf{0}$ on $\Gamma_D$.

In equation (1.15), we initially multiplied Equation 1.3 by a test function $\mathbf{v}$ that is zero on $\Gamma_D$, then and integrated it over $\Omega$. Afterwards, we integrated by parts, and finally used the Divergence theorem.

On the other hand, Equation 1.4 gets multiplied by a pressure test function $q$.

$$\int_\Omega q\nabla \cdot \mathbf{u} = 0 \quad \forall q \in L^2(\Omega) \tag{1.16}$$

## 1.4 Energy functional

The velocity field $\mathbf{u}$ solution of this weak formulation can also be found through an energy functional $\mathcal{J}(\mathbf{v})$ [3, 2]. As viscous forces dominate the physics of the problem, we can assume that viscous dissipation is the only source of internal energy. As in linear elasticity for solids, the infinitesimal work of internal stresses in the deformation is

$$\begin{aligned}
W_{\text{internal}}(D_{ij}) &= \int_0^{D_{ij}} \sigma_{ij}\, \mathrm{d}D'_{ij} \\
&= \int -p\delta_{ij}\, \mathrm{d}D'_{ij} + \int 2\mu D'_{ij}\, \mathrm{d}D'_{ij} + 2\int \frac{\tau_0 D'_{ij}}{2\sqrt{\frac{1}{2}D'_{kl}D'_{kl}}}\, \mathrm{d}D'_{ij} \\
&= -pD_{ii} + \mu D_{ij}D_{ij} + 2\tau_0\sqrt{\frac{1}{2}D_{ij}D_{ij}} \\
&= -p\nabla \cdot \mathbf{u} + 2\mu\|\mathbf{D}\|^2 + 2\tau_0\|\mathbf{D}\| \\
W_{\text{internal}}(\mathbf{u}) &= -p\nabla \cdot \mathbf{u} + \frac{\mu}{2}\|\dot{\boldsymbol{\gamma}}(\mathbf{u})\|^2 + \tau_0\|\dot{\boldsymbol{\gamma}}(\mathbf{u})\|
\end{aligned} \tag{1.17}$$

Notice that we were able to get rid of the twofold definition of the constitutive law of Bingham fluids. In fact, the stresses under the yield point $\tau_0$ do not generate any work due to the absence of deformation.

The total energy of the system $J$ is the work of internal forces diminished by the work of external forces $\int_\Omega \mathbf{f} \cdot \mathbf{u} + \int_{\Gamma_N} (\boldsymbol{\sigma} \cdot \hat{\mathbf{n}}) \cdot \mathbf{v}$. We can now reformulate equations (1.3) to (1.6) as the solution of the optimization problem (1.18).

$$\begin{aligned}
&\mathbf{u} = \arg\min_{\mathbf{v}\in\mathcal{V}} \mathcal{J}(\mathbf{v}) \\
&\mathcal{J}(\mathbf{v}) = \frac{\mu}{2}\int_\Omega \|\dot{\boldsymbol{\gamma}}(\mathbf{v})\|^2\, \mathrm{d}x + \tau_0\int_\Omega \|\dot{\boldsymbol{\gamma}}(\mathbf{v})\|\, \mathrm{d}x - \int_\Omega \mathbf{f}\cdot\mathbf{v}\, \mathrm{d}x - \int_{\Gamma_N} \mathbf{g}\cdot\mathbf{v}\, \mathrm{d}s \\
&\mathcal{V} = \{\mathbf{v} \mid \nabla\cdot\mathbf{v} = 0 \text{ in } \Omega,\ \mathbf{v} = \mathbf{u}_D \text{ on } \Gamma_D\}
\end{aligned} \tag{1.18}$$

In equation (1.17), we observe that the first term is the incompressibility constraint multiplied by its Lagrange multiplier: the pressure $p$, with a minus sign. It is therefore equivalent to to set it as a constraint of problem (1.18). Furthermore, in this study, only homogeneous Neumann boundary conditions will be considered. The boundary integral will therefore be neglected from now on.

We can show that the weak formulation and the energy minimization problem are equivalent. This can be done with a Gâteau derivative of the energy function around its minimum $(\mathbf{u}, p)$.

$$\begin{aligned}\mathcal{J}(\mathbf{u}+\epsilon\mathbf{v},p+\epsilon q) &= \int_\Omega \frac{\mu}{2}\|2\mathbf{D}(\mathbf{u}+\epsilon\mathbf{v})\|^2+\tau_0\|2\mathbf{D}(\mathbf{u}+\epsilon\mathbf{v})\| && (1.19)\\ &\quad -(p+\epsilon q)\nabla\cdot(\mathbf{u}+\epsilon\mathbf{v})-\mathbf{f}\cdot(\mathbf{u}+\epsilon\mathbf{v})\,\mathrm{d}\mathbf{x} && (1.20)\end{aligned}$$

$$\begin{aligned}\|2\mathbf{D}(\mathbf{u}+\epsilon\mathbf{v})\|^2 &= \frac{1}{2}2\mathbf{D}(\mathbf{u}+\epsilon\mathbf{v}):2\mathbf{D}(\mathbf{u}+\epsilon\mathbf{v})\\ &= 2\mathbf{D}(\mathbf{u}):\mathbf{D}(\mathbf{u})+4\epsilon\mathbf{D}(\mathbf{u}):\mathbf{D}(\mathbf{v})+2\epsilon^2\mathbf{D}(\mathbf{v}):\mathbf{D}(\mathbf{v}) && (1.21)\end{aligned}$$

$$\begin{aligned}\left.\frac{\mathrm{d}\mathcal{J}(\mathbf{u}+\epsilon\mathbf{v},p)}{\mathrm{d}\epsilon}\right|_{\epsilon=0} &= \int_\Omega \frac{\mu}{2}4\mathbf{D}(\mathbf{u}):\mathbf{D}(\mathbf{v})+\tau_0\frac{4\mathbf{D}(\mathbf{u}):\mathbf{D}(\mathbf{v})}{2\sqrt{2\mathbf{D}(\mathbf{u}):\mathbf{D}(\mathbf{u})}}-p\nabla\cdot(\mathbf{v})-\mathbf{f}\cdot(\mathbf{v})\,\mathrm{d}\mathbf{x}\\ &= \int_\Omega\left(2\mu+\frac{\tau_0}{\|\mathbf{D}(\mathbf{u})\|}\right)\mathbf{D}(\mathbf{u}):\mathbf{D}(\mathbf{v})-p\nabla\cdot\mathbf{v}-\mathbf{f}\cdot\mathbf{v} && (1.22)\end{aligned}$$

$$\left.\frac{\mathrm{d}\mathcal{J}(\mathbf{u},p+\epsilon q)}{\mathrm{d}\epsilon}\right|_{\epsilon=0} = -\int_\Omega q\nabla\cdot\mathbf{u} \qquad (1.23)$$

It is clear that $\mathbf{u}$ is a minimum of $J$ when (1.22) and (1.23) cancel out for every perturbation fields $\mathbf{v}$ and $q$. Those are precisely the expressions of the weak formulation we derived in (1.15) and (1.16).

# Chapter 2

# 1D solution using conic optimization

## 2.1 Introduction

To start, we will study the well know Poiseuille flow: the fluid is pushed forward by a constant pressure gradient in a channel between two infinitely long plates. This flow is of course a bit boring since it is stationary and fully developed. But it has the merit of being simple to introduce the optimization method in a one-dimensional framework.

We consider $\Omega = [-h/2, h/2]$ that represents a transversal 1D slice of the channel whose width is $h$. We also have that the velocity field reduces to $\mathbf{u} = u(y)\,\hat{\mathbf{e}}_\mathbf{x}$, since the flow is $x$ independent and incompressible. The pressure gradient $-\Delta p/\Delta x$ is imposed and considered as a body force $f$. The fluid obviously does not slip at the wall. Furthermore, the symmetry of the problem implies that the shear stress profile satisfies $\tau_{xy}(-y) = -\tau_{xy}(y)$.

## 2.2 Analytic solution

We can now solve for $u(y)$. With our hypothesis, the conservation of momentum (1.3) has boiled down to a scalar equation:

$$0 = f + \frac{\partial \tau_{xy}}{\partial y} \qquad \Longrightarrow \qquad \tau_{xy} = -fy \qquad \text{since } \tau_{xy}(0) = 0 \tag{2.1}$$

The expression of $\tau_{xy}$ can be obtained from the general expression of the stress tensor in (1.14):

$$\boldsymbol{\tau} = \begin{pmatrix} 0 & \mu\partial_y u \\ \mu\partial_y u & 0 \end{pmatrix} + \tau_0 \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \operatorname{sign}(\partial_y u) \tag{2.2}$$

$$\Longrightarrow \tau_{xy} = \begin{cases} \mu\partial_y u - \tau_0 & y > y_0 \\ \mu\partial_y u + \tau_0 & y < -y_0 \end{cases} \tag{2.3}$$

The threshold shear-stress is reached at $y = \pm y_0$ when

$$\tau_{xy}(\pm y_0) = \mp f\, y_0 = \mp\tau_0 \iff y_0 = \frac{\tau_0}{f} \tag{2.4}$$

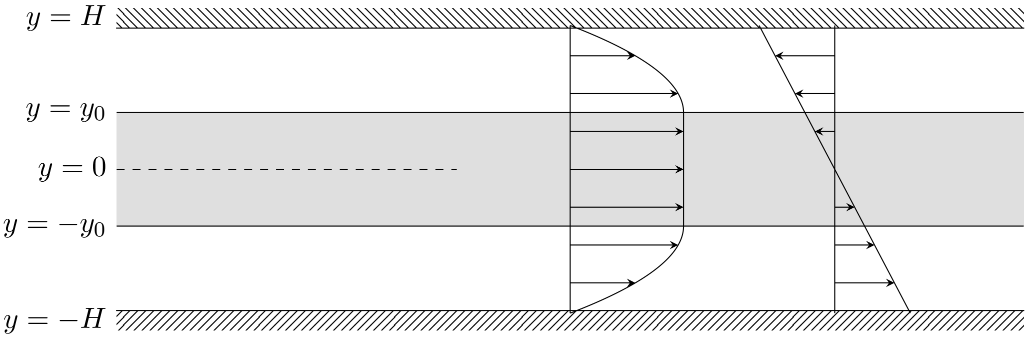


**Figure 2.1.** 1D problem channel situation, with the velocity and shear-stress profiles.

In view of (2.4), the fluid can be totally in the unyielded state when $\frac{\tau_0}{f} \geq \frac{h}{2}$. In that case, its velocity is zero everywhere to satisfy boundary conditions: the fluid is said to be in the *arrested state* [3].

In the lower zone $[-\frac{h}{2}, -y_0]$, we find the velocity profile $u_3(y)$ by integration equations (2.1) and (2.3):

$$\int_{-\frac{h}{2}}^{y} \mu \frac{\partial u}{\partial y} + \tau_0 = \int_{-\frac{h}{2}}^{y} -fy \tag{2.5}$$

$$u_3(y) = -\frac{\tau_0}{\mu}\left(\frac{h}{2} + y\right) + \frac{f}{2\mu}\left(\left(\frac{h}{2}\right)^2 - y^2\right) \tag{2.6}$$

The continuity of the velocity field allows us to find the velocity of the plug:

$$u_2 = u_3(-y_0) = \frac{f}{8\mu}\left(h - 2y_0\right)^2 \tag{2.7}$$

The procedure is very similar for the upper zone $[y_0, h/2]$.

Let us define the non-dimensional coordinate $\eta$, the reference velocity $U_\infty$ as the maximum velocity of a classical Poiseuille flow and the Bingham number $Bn$ that relates the yield and viscous stresses.

$$\eta = \frac{2y}{h} \qquad U_\infty = \frac{fh^2}{8\mu} \qquad Bn = \frac{\tau_0 h}{\mu U_\infty} = \frac{8\tau_0}{fh} \qquad \eta_0 = \frac{2y_0}{h} = \frac{Bn}{4} \tag{2.8}$$

The velocity profile can then be expressed as follows, in the presence of a yielded region, i.e. when $\eta_0 < 1 \iff Bn < 4$:

$$\frac{u(\eta)}{U_\infty} = -\frac{Bn}{2}(1-\eta) + (1-\eta^2) \qquad \eta_0 < \eta \leq 1 \tag{2.9}$$

$$\frac{u(\eta)}{U_\infty} = \left(1 - \frac{Bn}{4}\right)^2 \qquad -\eta_0 \leq \eta \leq \eta_0 \tag{2.10}$$

$$\frac{u(\eta)}{U_\infty} = -\frac{Bn}{2}(1+\eta) + (1-\eta^2) \qquad -1 \leq \eta < -\eta_0 \tag{2.11}$$

## 2.3 Finite element formulation

Let us first define the mesh $\Omega = [-h/2, h/2]$ that is made of $n$ elements $\Omega_i = [y_{i-1}, y_i]$, $i = 1, \dots, N$, and $N+1$ nodal values $(y_i)$. The velocity profile $u(y)$ is approximated by sum of shape functions $\phi_i$ weighted by their associated nodal value $U_i$. These shape functions have a compact support around their associated node, and satisfy $\phi_j(x_k) = \delta_{jk}$.

$$u^h(y) = \sum_{j=0}^{N} U_j \phi_j(y) \tag{2.12}$$

We will consider both linear and quadratic shape functions. They are expressed on a reference element $\hat{\Omega} = [-1, 1]$ and numbered with local node index $j = 1, \dots, n$ in Equation 2.13 and Figure 2.2.

$$\begin{aligned} \phi_1(\eta) &= \frac{1-\eta}{2} \\ \phi_2(\eta) &= \frac{1+\eta}{2} \end{aligned} \qquad \begin{aligned} \phi_1(\eta) &= \frac{1}{2}\,\eta\,(1-\eta) \\ \phi_2(\eta) &= \frac{1}{2}\,\eta\,(1+\eta) \\ \phi_2(\eta) &= 1-\eta^2 \end{aligned} \tag{2.13}$$

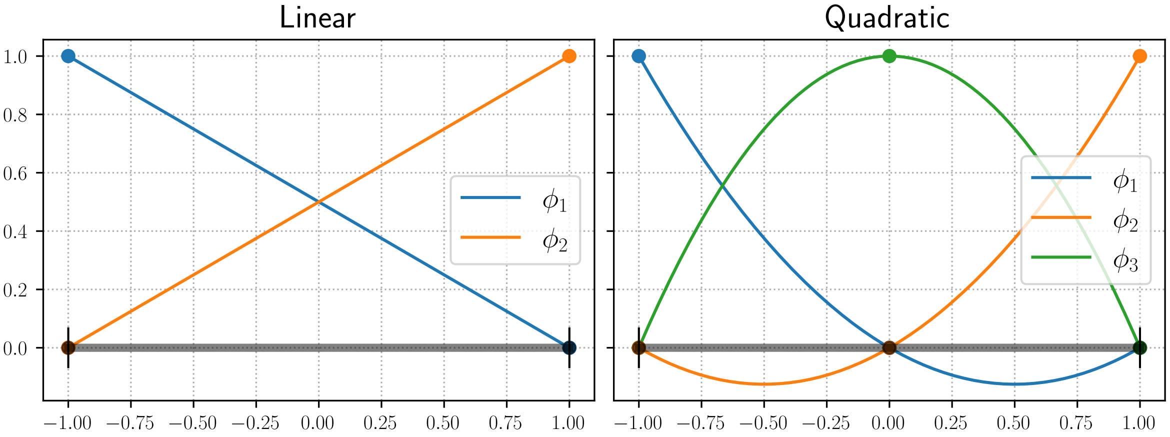


**Figure 2.2.** 1-dimensional shape functions $\phi$ on the reference element.

The strain-rate tensor with a unidirectional flow $u(y)$ only contains the off-diagonal component,

$$D_{12}^h(y) = \frac{1}{2}\frac{\partial u^h}{\partial y} \quad \text{where} \quad \left(\frac{\partial u^h}{\partial y}\right)_{\Omega_i} = \sum_{j=1}^{n} U_j \frac{\mathrm{d}\phi_j}{\mathrm{d}y} \tag{2.14}$$

The minimization function $J$ of Equation 1.18 becomes:

$$\begin{aligned} \mathcal{J}(u^h) &= \int_{\Omega} \left[ \frac{\mu}{2}\left(\frac{\partial u^h}{\partial y}\right)^2 + \tau_0 \left|\frac{\partial u^h}{\partial y}\right| - f u^h \right] \mathrm{d}y \\ &= \sum_{i=1}^{N} \int_{\Omega_i} \left[ \frac{\mu}{2}\left(\frac{\partial u^h}{\partial y}\right)^2_{\Omega_i} + \tau_0 \left|\frac{\partial u^h}{\partial y}\right|_{\Omega_i} - f u^h|_{\Omega_i} \right] \mathrm{d}y \\ &= \sum_{i=1}^{N} \sum_{g=1}^{ng} \omega_g \left[ \frac{\mu}{2}\left(\frac{\partial u^h}{\partial y}\right)^2_{y=y_g} + \tau_0 \left|\frac{\partial u^h}{\partial y}\right|_{y=y_g} - f u^h(y_g) \right] \frac{\Delta y_i}{2} \end{aligned} \tag{2.15}$$

where we used Gauss-Legendre quadrature of appropriate order with weights $\omega_g$ and coordinates $y_g$ in the element $\Omega_i$, mapped from coordinates $\eta_g$ in the reference element $[-1, 1]$.

In short, one should solve problem (2.16) hereunder

$$\begin{aligned} \text{minimize} \quad & \mathcal{J}(u^h) \\ \text{s.t.} \quad & u^h(y = -h) = u^h(y = h) = 0 \end{aligned} \tag{2.16}$$

However, this problem cannot be solve straightforwardly with classical optimization techniques such as gradient descent or Newton's method as it contains a non-differentiable term $|\frac{\partial u^h}{\partial y}|$. Next section will introduce a suitable method, known as conic optimization.

One could argue that absolute values can be transformed in multiple linear terms in this case. However, this is no longer true as we go to higher dimensions since the tensor norm will become the square root of multiple squared terms.

## 2.4 Conic optimization in a nutshell

This theory extends linear programming which is restricted to linear cost and linear constraints, by allowing specific nonlinear inequality constraints. For example, $\sqrt{x^2 + y^2} \leq z$ is expressed as $(x, y, z) \succeq_{L^3} 0$ or $(x, y, z) \in L^3$, where the Lorentz cone $L^3$ is precisely the set of points $(x, y, z)$ satisfying the nonlinear constraint.

A cone is a subset of a vector space that is closed under linear combinations with positive coefficients. Additional properties can upgrade a cone to a *proper* cone, the only ones we care about in conic optimization. These proper cones $K \subseteq \mathbb{R}^n$ satisfy the following properties:

1. $a \succeq_K 0 \implies \lambda a \succeq_K 0 \quad \forall \lambda \in \mathbb{R}^+$ (cone)
2. $a \succeq_K 0$ and $b \succeq_K 0 \implies a + b \succeq_K 0$ (closed under addition)
3. $x \succeq_K 0$ and $x \preceq_K 0 \implies x = 0$ (pointed)
4. $\text{int}(K) \neq \emptyset$ (solid)
5. if $\{x_i\}_{i \to \infty}$ with $x_i \succeq_K 0 \quad \forall i$, then $\lim_{i \to \infty} x_i = \bar{x} \implies \bar{x} \succeq_K 0$ (closed)

Cones with such properties will ensure us a global convergence of the Newton's iterations through functions known as *self-concordant barriers*. This is of course very appreciated since Newton's algorithm only converges locally in a general framework. A function $g : X \to \mathbb{R}$, where $X \subseteq \mathbb{R}^n$ is called self-concordant iff

- $g \in \mathcal{C}^3$, and
- $g$ is convex, and
- $\nabla^3 g(x)[h, h, h] \leq 2\big(\nabla^2 g(x)[h, h]\big)^{3/2} \quad \forall x \in X \quad \forall h \in \mathbb{R}^n$

where $\nabla^3 g(x)[h, h, h] = \sum_{i,j,k} \frac{\partial^3 g}{\partial x_i \partial x_j \partial x_k}(x)\ h_i h_j h_k$. With univariate functions $g(x) : \mathbb{R} \to \mathbb{R}$, this last property gets simplified to $|g'''(x)| \leq 2\big(g''(x)\big)^{3/2}$. Multivariate functions can then be verified to be self-concordant using the univariate test version on $G_{x,h}(t) : \mathbb{R} \to \mathbb{R} : t \mapsto g(x + th) \quad \forall x \in X \quad \forall h \in \mathbb{R}^n$.

Last but not least, self-concordance (s.c.) is preserved through

| Name | Definition and barrier |
|---|---|
| Half-space | $\mathbb{R}_+$<br>$g(x) = -\log(x)$ |
| Lorentz/quadratic cone | $L^{n+1} = \{(x,t) \in \mathbb{R}^n \times \mathbb{R} \quad \vert \quad \|x\|_2 \leq t\}$<br>$g(x,t) = -\log\big(t^2 - \|x\|^2\big)$ |
| Rotated Lorentz cone | $L_R^{n+2} = \{(x,s,t) \in \mathbb{R}^n \times \mathbb{R}_+ \times \mathbb{R}_+ \quad \vert \quad \|x\|_2^2 \leq 2st\}$<br>$g(x,s,t) = -\log\big(2st - \|x\|_2^2\big)$ |
| Exponential cone | $E = \text{closure}\{(x,y,z) \in \mathbb{R}^3 \quad \vert \quad z \geq y\exp(e/y), y > 0\}$<br>$g(x,y,z) = -\log(z - y\exp(e/y)) - \log(y) - \log(z)$ |
| Power cone | $P_\alpha = \{(x,y,z) \in \mathbb{R}^3 \quad \vert \quad x^\alpha y^{1-\alpha} \geq \vert z\vert, x > 0, y > 0, 0 < \alpha < 1\}$<br>$g(x,y,z) = -\log\big(x^{2\alpha}y^{2-2\alpha} - z^2\big) - \log(x) - \log(y)$ |

**Table 2.1.** Most frequent proper cones used in conic programming.

- sums: let $f$ and $g$ be s.c. functions, then $h = f + g$ is also a s.c. function.
- linear change of variables: let $x \mapsto f(x)$ be a s.c. functions, then $y \mapsto f(Ay + b)$ is also a s.c. function.

The most common proper cones, with their associated s.c. barrier. are listed in Table 2.1. It may not seem very useful to be limited to such a short list of cones, but a whole zoo of nonlinear constraints can be reformulated to fit in any of these five cones.

Once all the nonlinear constraints have been translated into conic constraints with s.c. barriers $g_i(x)$, the solution is found using an *Interior-point method* that minimizes

$$f_\mu(x) = \frac{c^\top x}{\mu} + \sum_i g_i(x) \tag{2.17}$$

where $c$ is the original linear cost and $\mu > 0$ is progressively brought to zero. For each $\mu$, there is an unique solution $x_\mu^*$. The set of solutions $x_\mu^*$ is called the *central path*. One can eventually retrieve the solution of the original problem since $x_\mu^* \to x^*$ as $\mu \to 0$.

Let us take a basic example in 2 dimensions.

$$\begin{aligned}
&\min_{x,y} && x \\
&\text{s.t.} && 2x + y \geq 3 \quad \text{and} \quad 2x - 4y \leq 1 \quad \text{and} \quad x + 5y \leq 10 \\
&\implies && f_\mu(x,y) = \frac{x}{\mu} - \log(2x + y - 3) - \log(1 - 2x + 4y) - \log(10 - x - 5y)
\end{aligned}$$

In practice, no one computes the $x_\mu^*$ on the *central path* because these points are only used as starting point of the next iteration with a lower $\mu$. Instead, we use an iterative algorithm where we alternate between a Newton step and a decrease of $\mu$ until we reach the required precision. Newton steps keep the current solution close enough to the central path, while the decrease of $\mu$ brings the objective function $f_\mu$ to the linear function $c^\top x$.

For a precision $\epsilon$ s.t. $c^\top x - c^\top x^* < \epsilon$, a solution $x$ is obtained in $\mathcal{O}(\sqrt{\nu}\log\frac{1}{\epsilon})$ iterations with the *short-step algorithm* briefly described here above. $\nu = \sum_i \nu_i$, with the barrier parameter $\nu_i$, in a sense related to the *steepness* of the constraint $i$. The initial value x must be close enough to the central path. This can be done with damped Newton steps from any admissible $x \in X$ CITE.

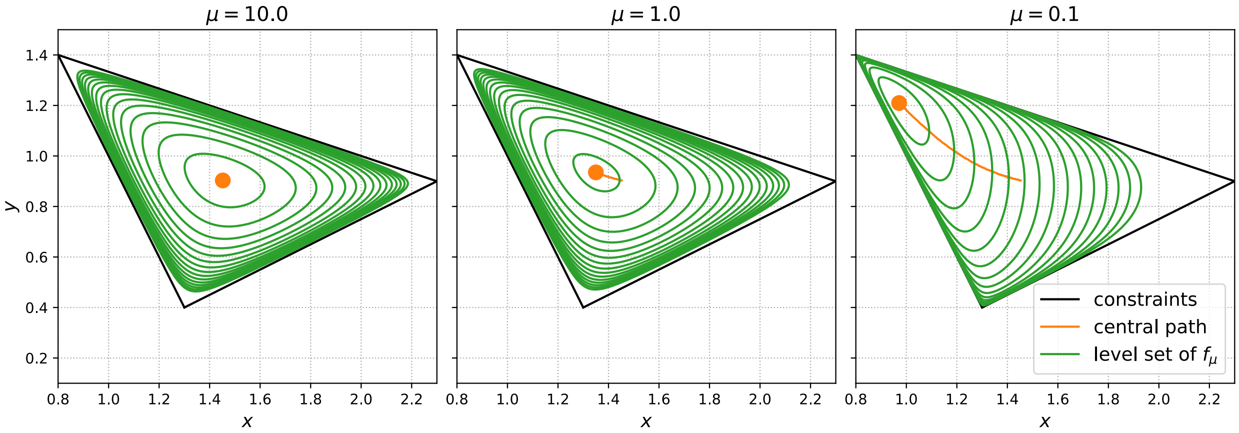


**Figure 2.3.** Solution of the basic example using the interior point method

## 2.5 Finite element solution

With our recent knowledge in conic programming, we can reformulate problem (2.16) in terms of second order cones only (SOCP).

$$
\begin{aligned}
\underset{U_j,S_{i,g},T_{i,g}}{\text{minimize}} \quad & \sum_{i=1}^{N}\sum_{g=1}^{ng}\omega_g\left[\frac{\mu}{2}\,S_{i,g}+\tau_0\,T_{i,g}-f u^h(y_g)\right]\frac{\Delta y_i}{2} \\
\text{s.t.} \quad & \left(\frac{\partial u^h}{\partial y}\right)^2_{y=y_g} \le S_{i,g} \quad \forall i\,\forall g \qquad \Longleftrightarrow \qquad \left[S_{i,g},\frac{1}{2},\left(\frac{\partial u^h}{\partial y}\right)^2_{y=y_g}\right]\in L_R^3 \quad \forall i\,\forall g \\
& \left|\frac{\partial u^h}{\partial y}\right|_{y=y_g} \le T_{i,g} \quad \forall i\,\forall g \qquad \Longleftrightarrow \qquad \left[T_{i,g},\left|\frac{\partial u^h}{\partial y}\right|_{y=y_g}\right]\in L^2 \quad \forall i\,\forall g \\
& U_0 = U_N = 0
\end{aligned}
\tag{2.18}
$$

We minimize this problem over the nodal velocities $U$, and the newly added $S$ and $T$ variables. Even if the inequalities may confuse at first glance, they are valid from a modelling standpoint. They are always verified as equalities at the optimum. We can show it by contradiction. Let us assume that $\left(\frac{\partial u^h}{\partial y}\right)^2 < S_{i,g}$ for a specific index $i, g$ at the optimum. Then the cost can be reduced by decreasing $S_{i,g}$ by $\epsilon > 0$, keeping all other variables unchanged: we are not at the optimum.

At this stage, the finite element solution will be relevant only when the interface is represented on the mesh. Therefore, the arbitrary mesh is manually modified such that two nodes are placed at $y = \pm y_0$. In the next chapter, we will develop an algorithm that moves the nodes without knowing the interface position beforehand.

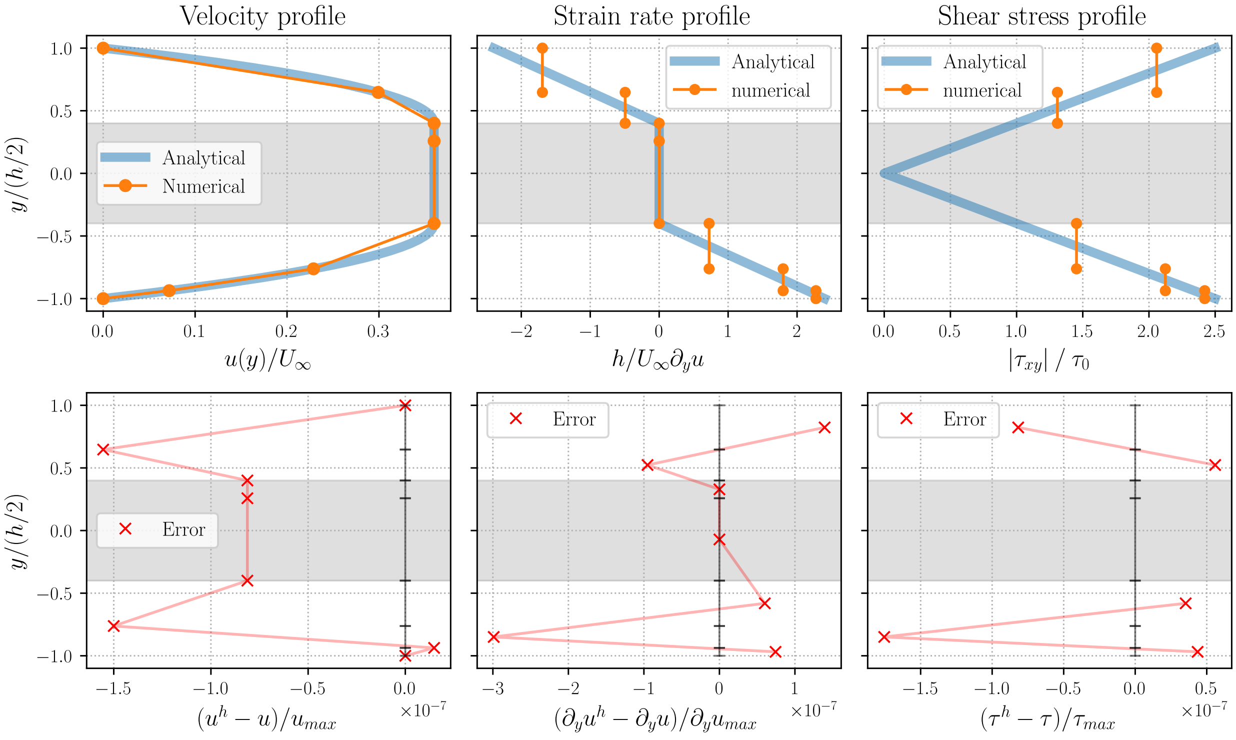


**Figure 2.4.** FE solution of the Poiseuille flow with $P1$ elements. The unyielded zone is shaded in grey. The error is shown at the nodes for $u(y)$ and at the element center for the strain and shear.

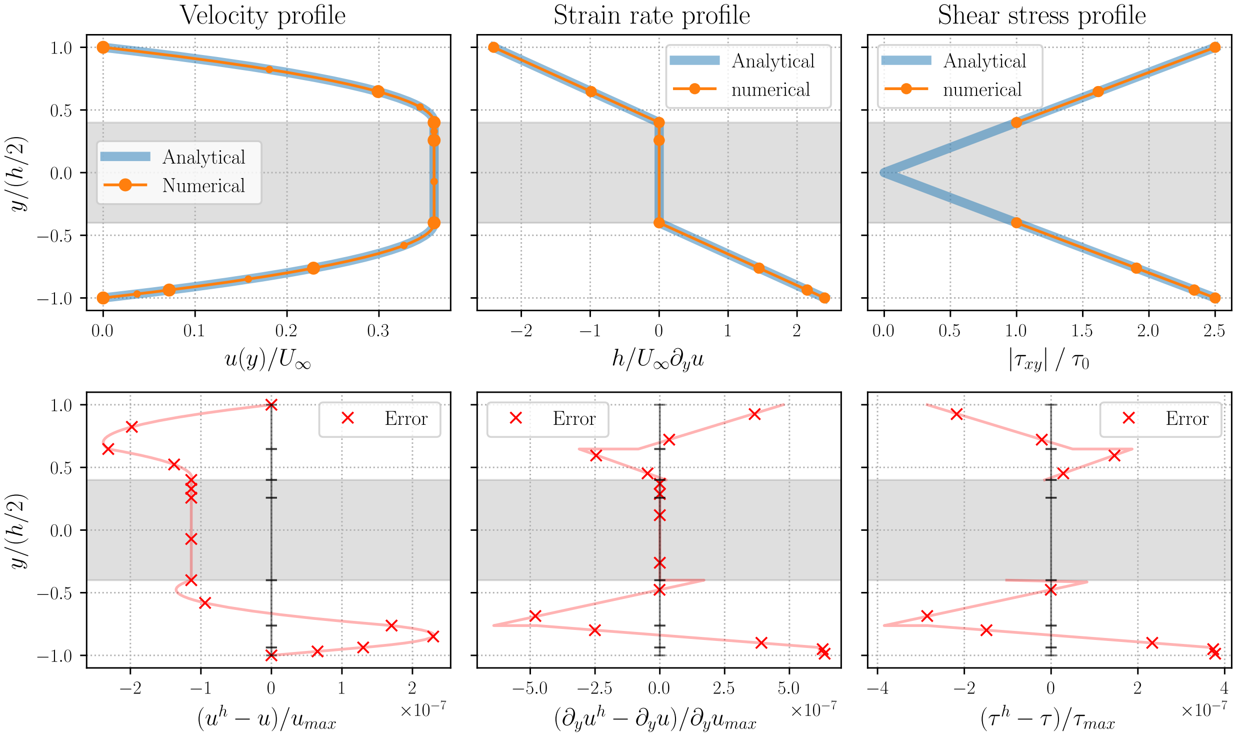


**Figure 2.5.** FE solution of the Poiseuille flow with $P2$ elements. The unyielded zone is shaded in grey. The error is computed on the whole domain $\Omega$.

# Chapter 3

# Interface tracking

INTRODUCE CONCEPT interface tracking

Going back to the constitutive law of bingham fluids, one can see that the strain rate at the interface is continuous and equal to 0. There is actually a discontinuity, but it only appears in the second derivative of the velocity:

— $\partial_{yy}u = 0$ in the solid plug as $\partial_y = 0$ over a whole nonempty interval.

— and $\partial_{yy}u > 0$ as for a Newtonian Poiseuille flow.

This is what we observed in figures 2.4 and 2.5.

We will therefore use this continuity of the strain field as a termination criteria for the tracking algorithm. Sadly, P1-elements for the velocity only provide piecewise discontinuous strains. Therefore, we reconstruct an continuous strain field as a linear approximation of the finite element solution at the Gauss points. This approximation is one-sided at each interface as we should only take the information from the yielded regions.

TALK ABOUT SUPERCONVERGENCE.

The procedure is described in algorithm 1.

The algorithm is illustrated in figures 3.2 and 3.4. The strain field $\partial_y u$ is shown at each iteration with with its corresponding mesh. The complete overview of the solution is provided for the first and last iterations, for both P1 and P2 elements, in figures 3.2 and 3.4.

Surprisingly the convergence is faster for P1 elements (piecewise constant strains). This can be easily explained by a flaw in algorithm 1: the reconstruction is made over all the Gauss points of elements near the current interface. However, one of these gauss point can be inside the unyielded region (as in chapter 3). In that case, the reconstruction is deteriorated because it uses information from the wrong side, leading to a slower convergence.

**Algorithm 1:** Interface-tracking algorithm in 1 dimension.

```
optimalMesh ← false;
$\mathcal{M}$ ← initial mesh with nodes $y_i$;
while not optimalMesh do
    Minimize the energy functional $\mathcal{J}$ with $\mathcal{M}$ and retrieve the solutions $U_j, T_{i,g}$;
    $\mathcal{I}$ ← set of current interfaces (node shared between yielded and unyielded elements);
    if $\mathcal{I} = \emptyset$ then
        break ;                    //Mesh should be finer to have unyielded elements
    end
    forall $k \in \mathcal{I}$ do
        $\ell_k$ ← one-sided linear approximation function of the strain close to $y_k$;
        if $\ell_k(y_k) \neq 0$ then
            optimalMesh ← false ;    //since $\partial_y u$ is not continuous at the interface
            $y_k$ ← root($\ell_k(y)$) ;                                      //update $\mathcal{M}$
        end
    end
end
return $U_j$
```

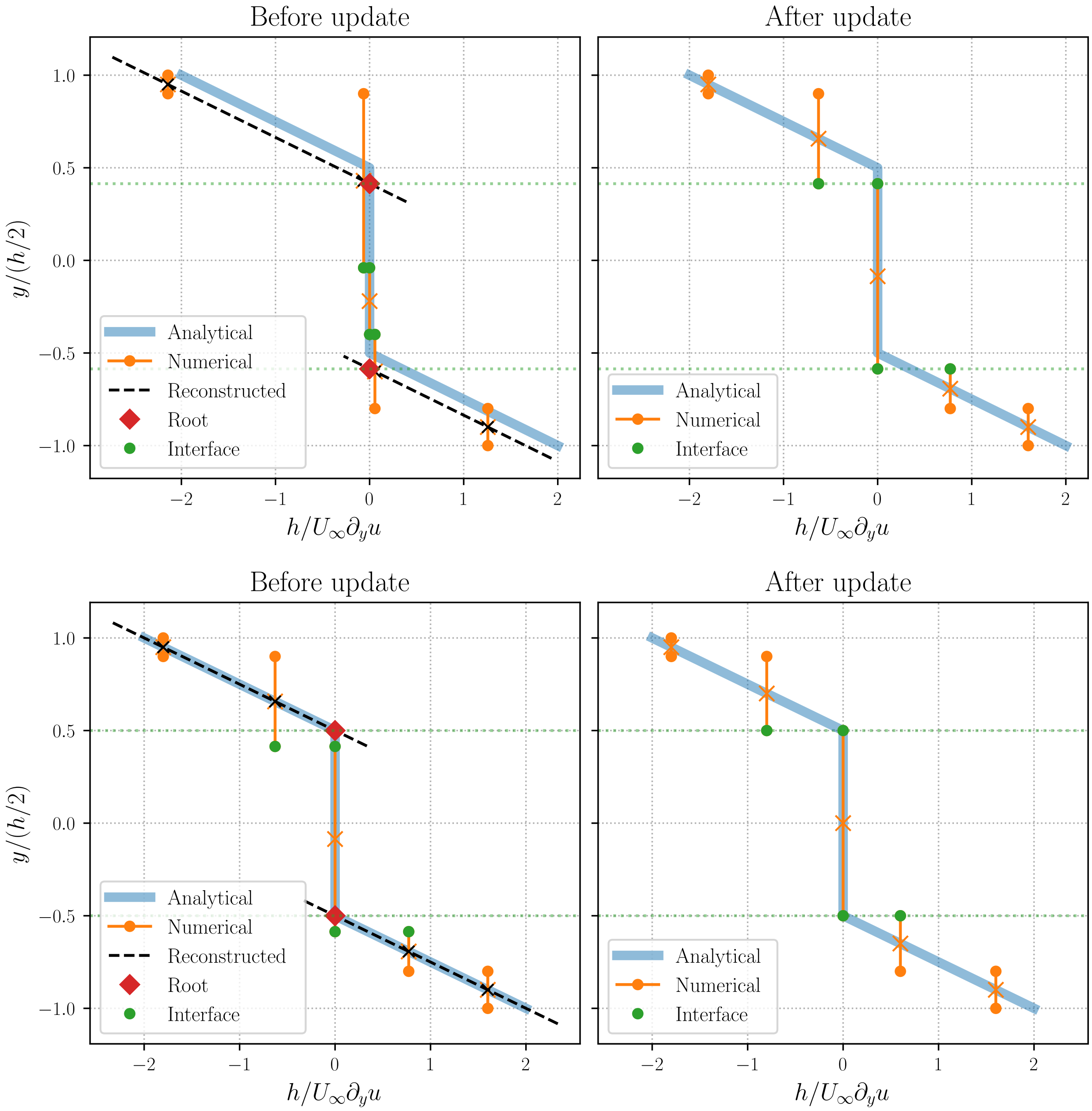
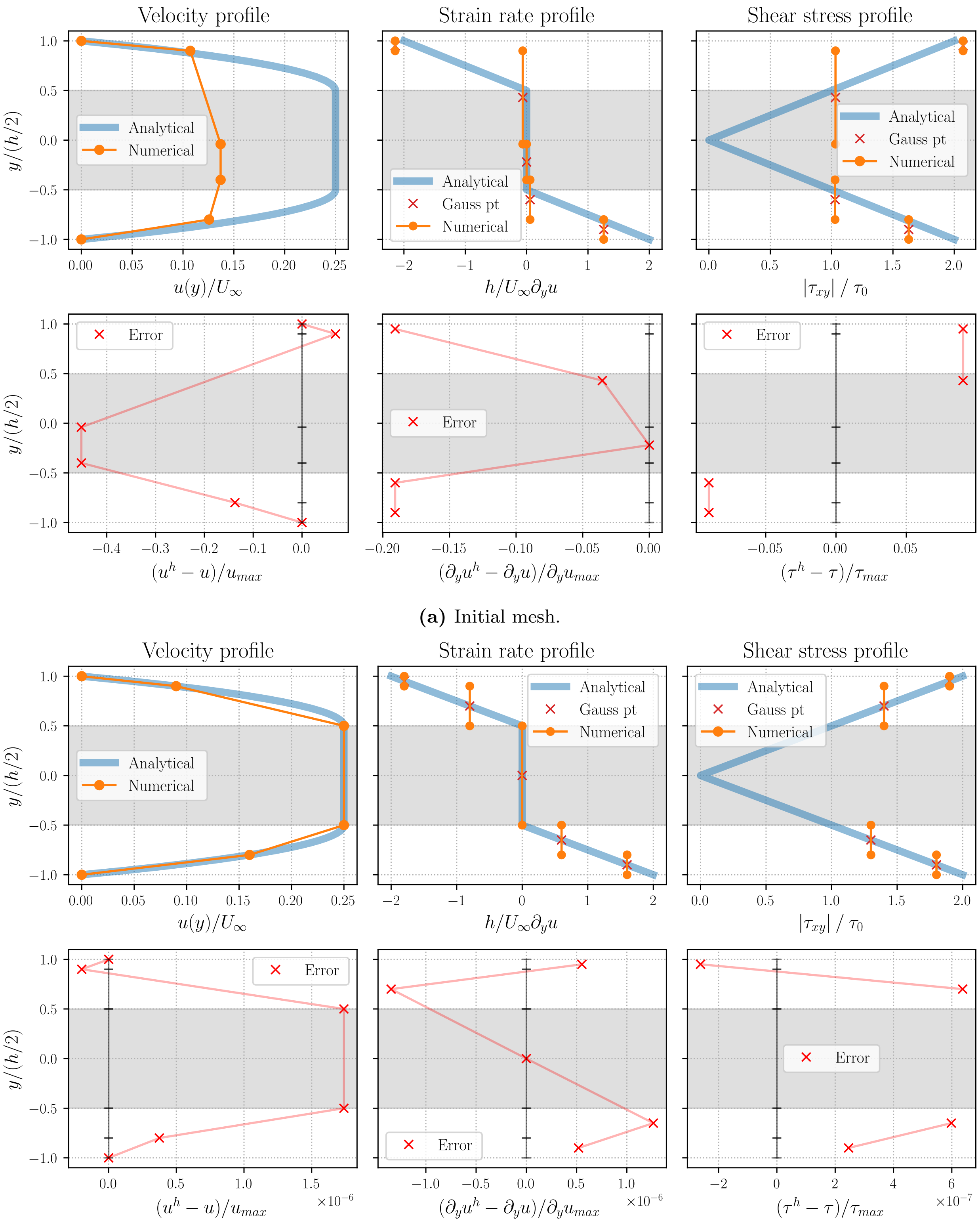


**Figure 3.1.** Strain field and its reconstruction during the interface-tracking algorithm. First to second iteration above, and second to third iteration below. P1 elements.



**(a)** Initial mesh.

**(b)** Final mesh.

**Figure 3.2.** Finite element solution of the Poiseuille flow at the start, and after convergence of the algorithm. P1 elements.

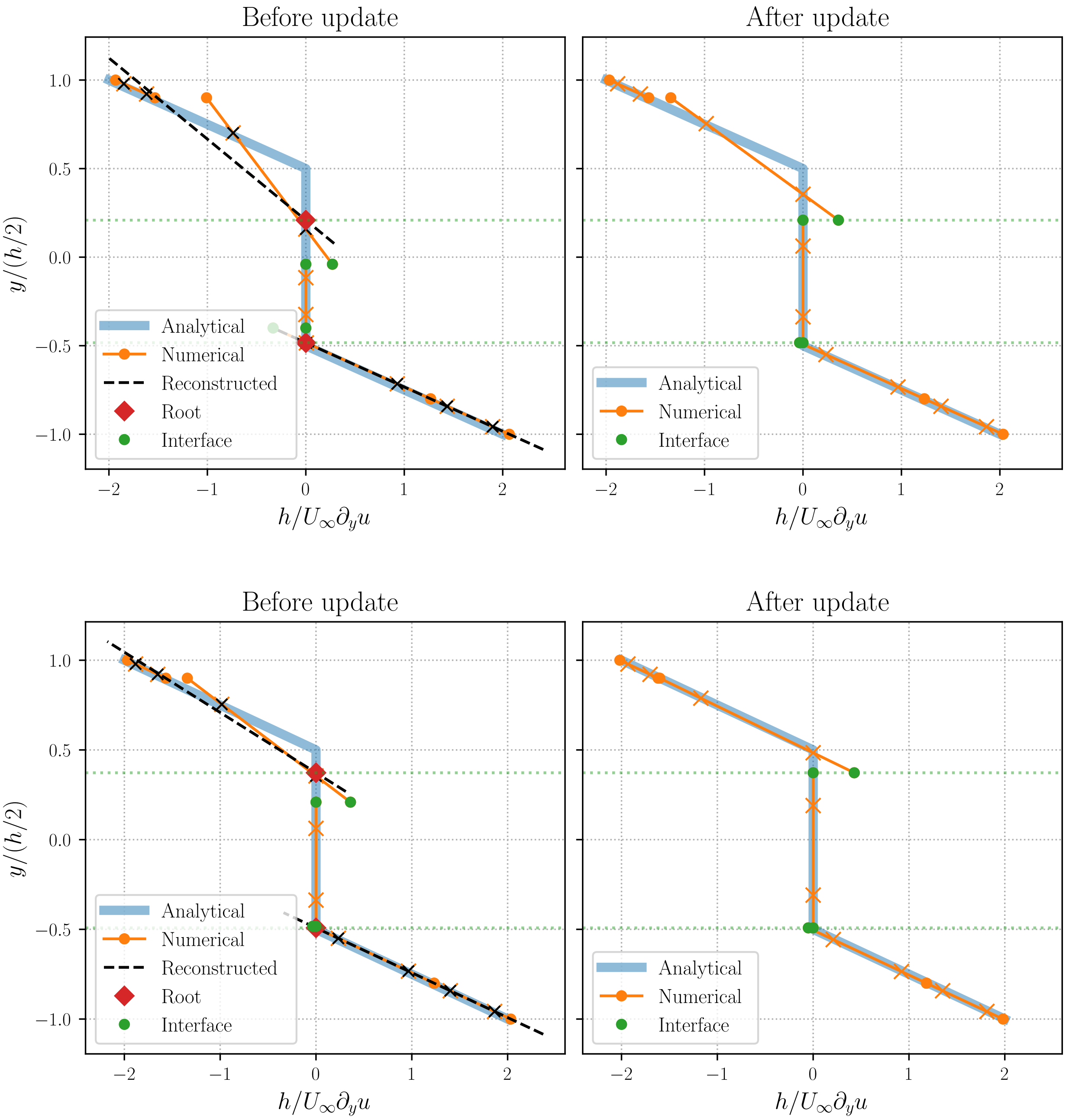
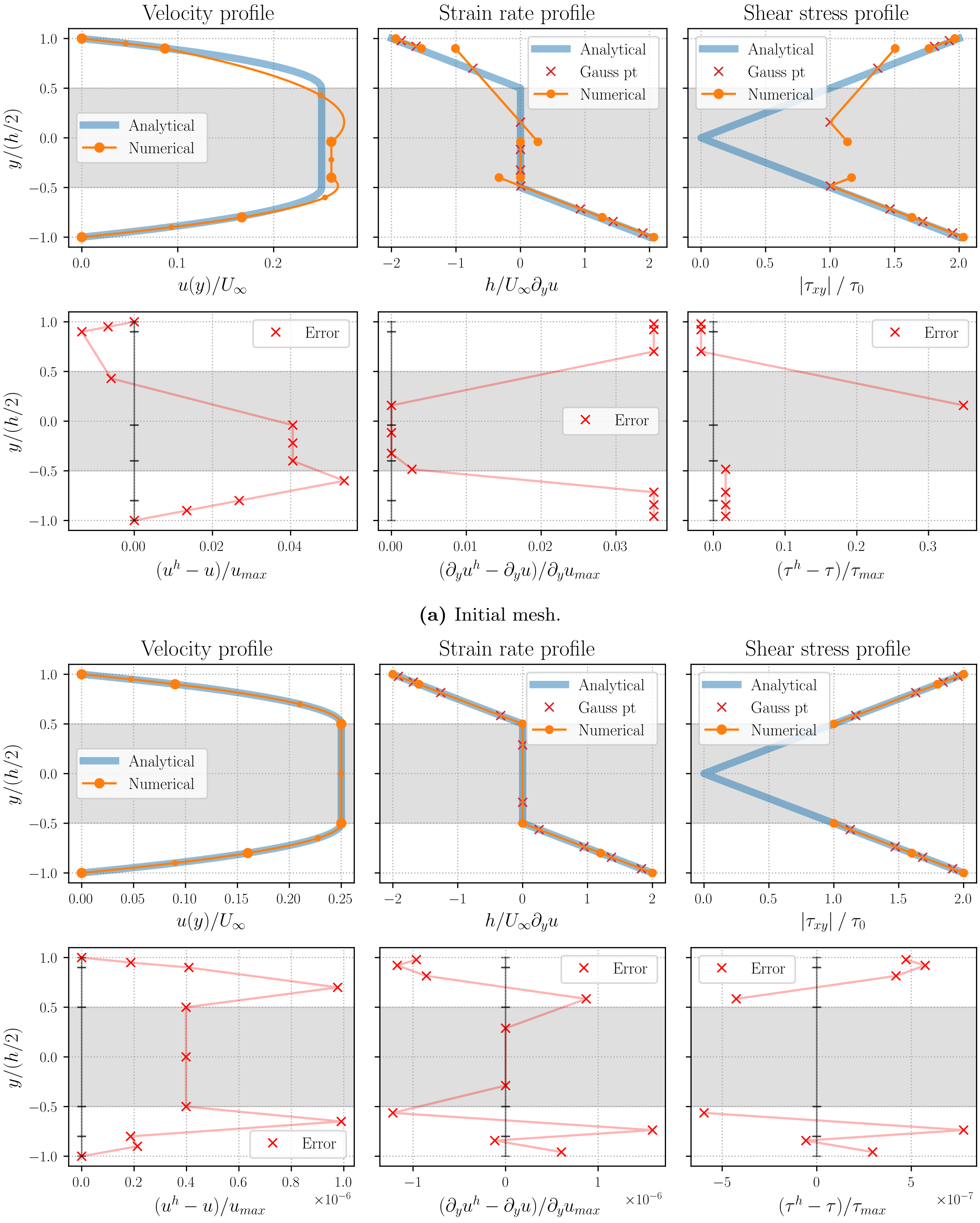


**Figure 3.3.** Strain field and its reconstruction during the interface-tracking algorithm. First to second iteration above, and second to third iteration below. P1 elements.



**(a)** Initial mesh.

**(b)** Final mesh.

**Figure 3.4.** Finite element solution of the Poiseuille flow at the start, and after convergence of the algorithm. P1 elements.

# Chapter 4

# 2D problem

## 4.1 Equations

It is well known that velocity-pressure fields discretized with P1-P0 and P1-P1 elements do not provide stable discretizations of the Stokes equations, or equivalently of incompressible linear elasticity REF. A first option is to enrich the velocity space with a bubble function: this is the MINI element. A second option is to use P2 elements for the velocity field. Both methods were implemented in our case.

The shape functions involved are represented graphically in Again, the finite element approximation velocity field $\mathbf{v}^h$ reads as

$$\mathbf{v}^h(x,y) = \sum_j \mathbf{V}_j \; \phi_j(x,y) \tag{4.1}$$

To express the deformation norm $\|\dot{\boldsymbol{\gamma}}\|$ needed in the functional, we need the velocity gradient:

$$\nabla \mathbf{v} = \begin{bmatrix} \frac{\partial u}{\partial x} & \frac{\partial u}{\partial y} \\ \frac{\partial v}{\partial x} & \frac{\partial v}{\partial y} \end{bmatrix} \tag{4.2}$$

$$\Longrightarrow \qquad \|\dot{\boldsymbol{\gamma}}(u,v)\|^2_{\text{Cart}} = 2\left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial u}{\partial y} + 2\frac{\partial v}{\partial x}\right)^2 + \left(\frac{\partial v}{\partial y}\right)^2 \tag{4.3}$$

Each of these partial derivatives are computed as a sum of shape functions derivatives. These spatial derivatives in the physical domain $(x,y)$ are themselves computed from derivatives in the parametric domain $(\xi,\eta)$ through the Jacobian of the transformation.

$$\frac{\partial u_i^h}{\partial x_j} = \sum_k U_k \frac{\partial \phi_k}{\partial x_j} \qquad i = 1,2 \quad j = 1,2 \tag{4.4}$$

$$(\nabla_{\mathbf{x}}\, \phi)_{\text{Cart}} = \nabla_{\boldsymbol{\xi}} \phi \; \cdot \frac{\mathrm{d}\boldsymbol{\xi}}{\mathrm{d}\mathbf{x}} \tag{4.5}$$

$$\frac{\mathrm{d}\boldsymbol{\xi}}{\mathrm{d}\mathbf{x}} = \left(\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}\boldsymbol{\xi}}\right)^{-1} = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} \end{bmatrix}^{-1} \tag{4.6}$$

The functional of the approximate field $\mathbf{v}^h$ is recalled below, along with the incompressibility

./svg-inkscape/shape_fcts_2d_P1_svg-tex.pdf
./svg-inkscape/shape_fcts_2d_bubble_svg-tex.pdf
./svg-inkscape/shape_fcts_2d_P2_svg-tex.pdf

**(a)** Lagrange P1

**(b)** Bubble enriched Lagrange

**(c)** Lagrange P2

**Figure 4.1.** Shape functions used for the pressure and velocity fields in 2-dimensions.

constraint, where $\psi$ denotes the pressure linear shape function, attached to every primary node.

$$\mathcal{J}(u^h, v^h) = \int_\Omega \frac{\mu}{2} \|\dot{\boldsymbol{\gamma}}(u^h, v^h)\|^2 + \tau_0 \|\dot{\boldsymbol{\gamma}}(u^h, v^h)\| - \mathbf{f} \cdot \mathbf{v}^h \, \mathrm{d}x \tag{4.7}$$

$$0 = \int_\Omega \psi_l \; \nabla \cdot \mathbf{v}^h \qquad \forall \quad \psi_l \tag{4.8}$$

The minimization problem can for Cartesian coordinate systems is given hereunder while cylindrical and polar coordinates are detailed in appendix A.

$$
\begin{aligned}
\underset{\mathbf{V}_j, S_{i,g}, T_{i,g}}{\text{minimize}} \quad & \sum_i \sum_g \omega_g \left[ \frac{\mu}{2} S_{i,g} + \tau_0 T_{i,g} - \mathbf{f} \cdot \mathbf{v}^h|_{\mathbf{x}_g} \right] \det \left( \frac{\mathrm{d}\mathbf{x}}{\mathrm{d}\boldsymbol{\xi}} \right)_{i,g} & & \\
\text{s.t.} \quad & \left( S_{i,g}, \frac{1}{2}, \sqrt{2}\frac{\partial u^h}{\partial x}, \sqrt{2}\frac{\partial v^h}{\partial y}, \frac{\partial u^h}{\partial y} + \frac{\partial v^h}{\partial x} \right) \in L^5_R & & \forall i, g \\
& \left( T_{i,g}, \sqrt{2}\frac{\partial u^h}{\partial x}, \sqrt{2}\frac{\partial v^h}{\partial y}, \frac{\partial u^h}{\partial y} + \frac{\partial v^h}{\partial x} \right) \in L^4 & & \forall i, g \\
& 0 = \sum_i \sum_g \omega_g \psi_l|_{\mathbf{x}_g} \left( \frac{\partial u^h}{\partial x} + \frac{\partial v^h}{\partial y} \right) & & \forall\ l \\
& \mathbf{V}_j = \mathbf{u}_D & & \forall\ \mathbf{V}_j \in \Gamma_D
\end{aligned}
\tag{4.9}
$$

# Bibliography

[1] Neil Balmforth and Richard Craster. *Geophysical Aspects of Non-Newtonian Fluid Mechanics*, volume 582, pages 34–51. Springer, 01 2008.

[2] Jeremy Bleyer, Mathilde Maillard, Patrick de Buhan, and Philippe Coussot. Efficient numerical computations of yield stress fluid flows using second-order cone programming. *Computer Methods in Applied Mechanics and Engineering*, 283:599–614, 2015.

[3] P. Saramito. *Complex fluids : modeling and algorithms.* Springer, 2016.

# Appendix A

# Curvilinear coordinates

In axisymmetric cylindrical coordinates, the solution only depends on the radial and axial coordinates $r, z$. We also assume that the velocity field has no tangential component.

$$\begin{bmatrix} x(r,\theta=0,z) \\ y(r,\theta=0,z) \\ z(r,\theta=0,z) \end{bmatrix} = \begin{bmatrix} r \\ 0 \\ z \end{bmatrix} \tag{A.1}$$

$$\mathbf{u}(r,z) = u_r(r,z)\mathbf{e}_r + u_z(r,z)\mathbf{e}_z \tag{A.2}$$

$$(\nabla\, \phi)_{\mathrm{Cyl}} = \nabla_{\boldsymbol{\xi}}\phi\ \cdot \frac{\mathrm{d}\boldsymbol{\xi}}{\mathrm{d}\mathbf{x}} \cdot \frac{\mathrm{d}\mathbf{x}}{\mathrm{d}(r,z)} \tag{A.3}$$

$$\nabla \cdot \mathbf{v} = \frac{\partial u_r}{\partial r} + \frac{u_r}{r} + \frac{\partial u_z}{\partial z} \tag{A.4}$$

$$\|\dot{\boldsymbol{\gamma}}(u,v)\|^2_{\mathrm{Cyl}} = 2\left(\frac{\partial u_r}{\partial r}\right)^2 + 2\left(\frac{u_r}{r}\right)^2 + 2\left(\frac{\partial u_z}{\partial r}\right)^2 + \left(\frac{\partial u_r}{\partial z} + \frac{\partial u_z}{\partial r}\right)^2 \tag{A.5}$$

Polar coordinates provide quite similar equations. Here again, we assume axisymmetry and the absence of azimuthal velocity.

$$\begin{bmatrix} x(r,\theta,\varphi=0) \\ y(r,\theta,\varphi=0) \\ z(r,\theta,\varphi=0) \end{bmatrix} = \begin{bmatrix} r\sin\theta \\ 0 \\ r\cos\theta \end{bmatrix} \tag{A.6}$$

$$\mathbf{u}(r,z) = u_r(r,\theta)\mathbf{e}_r + u_\theta(r,\theta)\mathbf{e}_\theta \tag{A.7}$$

$$(\nabla\, \phi)_{\mathrm{Pol}} = \nabla_{\boldsymbol{\xi}}\phi\ \cdot \frac{\mathrm{d}\boldsymbol{\xi}}{\mathrm{d}\mathbf{x}} \cdot \frac{\mathrm{d}\mathbf{x}}{\mathrm{d}(r,\theta)} \tag{A.8}$$

$$\nabla \cdot \mathbf{v} = \frac{1}{r^2}\frac{\partial(r^2 u_r)}{\partial r} + \frac{1}{r\sin\theta}\frac{\partial(u_\theta \sin\theta)}{\partial\theta} \tag{A.9}$$

$$\begin{aligned}\|\dot{\boldsymbol{\gamma}}(u,v)\|^2_{\mathrm{Cyl}} = {} & 2\left(\frac{\partial u_r}{\partial r}\right)^2 + 2\left(\frac{1}{r}\frac{\partial u_\theta}{\partial\theta} + \frac{u_r}{r}\right)^2 + 2\left(\frac{u_r}{r} + \cot\theta\frac{u_\theta}{r}\right)^2 \\ & + \left(\frac{1}{r}\frac{\partial u_r}{\partial\theta}\frac{\partial u_\theta}{\partial r} - \frac{u_\theta}{r}\right)^2\end{aligned} \tag{A.10}$$