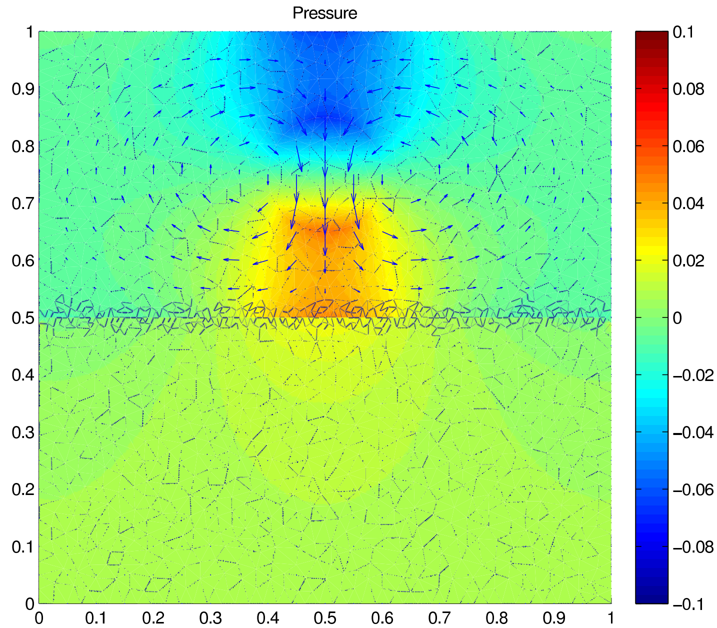

Figure 1: Pressure and velocity solution for a sinking, fluid slab impinging on viscosity contrast problem.

1 Exercise: Linear, incompressible Stokes flow with FE

Reading

- *Hughes* (2000), sec. 4.2-4.4
- *Dabrowski et al.* (2008), sec. 4.1.2, 4.3.1, 4.4-4.7

This FE exercise is again based on the MILAMIN package by *Dabrowski et al.* (2008). As for the heat and elasticity problems, we simplified their "mechanical", incompressible Stokes fluid solver to reduce the dependency on packages external to MATLAB.

Dabrowski et al. (2008) have a highly optimized version, which you can obtain from us or the original authors; it uses, *e.g.*, reordering of node numbers to improve matrix solutions which comes an important memory issue for larger problems. The notation here is close to *Dabrowski et al.* (2008), for simplicity, but *Hughes* (2000) has a somewhat clearer exposition.

1.1 Implementation of incompressible, Stokes flow

We are interested in the instantaneous solution of a fluid problem in the absence of inertia (infinite Prandtl number limit), as is appropriate for the Earth's mantle, for example (see secs. **??** and **??**). These approximations transform the general, Navier-Stokes equation for fluids into the Stokes equation, which is easier to solve, on the one hand, because there is no turbulence. On the other hand, it is more complicated numerically as Stokes requires implicit solution methods, whereas turbulent equations can often be solved in an explicit manner.

The static force-balance equations for body forces due to gravity are given by

$$\underline{\nabla} \cdot \underline{\underline{\sigma}} = \underline{f} = \rho \underline{g} \quad \text{or} \quad \frac{\partial \sigma_{ij}}{\partial x_j} = \rho g_i, \tag{1}$$

where $\underline{\underline{\sigma}}$ is the stress tensor, ρ density, and $\underline{g}$ gravitational acceleration ($g_i = g\delta_{iz}$).

We assume that the medium is incompressible and a linear (Newtonian) fluid constitutive law holds,

$$\sigma_{ij} = -p\delta_{ij} + 2\eta\dot{\varepsilon}'_{ij}, \tag{2}$$

where η is the viscosity, p pressure, and $\underline{\underline{\dot{\varepsilon}}}'$ the deviatoric strain-rate tensor,

$$\dot{\varepsilon}'_{ij} = v_{(i,j)} - \frac{1}{3}\frac{\partial v_k}{\partial x_k}\delta_{ij} = \frac{1}{2}\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\right) - \frac{1}{3}\frac{\partial v_k}{\partial x_k}\delta_{ij}, \quad \text{or} \quad \underline{\underline{\dot{\varepsilon}}}' = \underline{\underline{\dot{\varepsilon}}} - \frac{1}{3}tr(\underline{\underline{\dot{\varepsilon}}})\underline{\underline{I}}, \tag{3}$$

where $\underline{v}$ are the velocities, and $\underline{\underline{\dot{\varepsilon}}}'$ is the total strain-rate reduced by the isotropic part.

We can define

$$\dot{e} = \frac{1}{3}\sum_i \dot{\varepsilon}_{ii} = \frac{1}{3}\dot{\varepsilon}_{ii} = \frac{1}{3}tr(\underline{\underline{\dot{\varepsilon}}}) \tag{4}$$

in analogy to the pressure

$$p = -\frac{1}{3}\sum_i \sigma_{ii} \tag{5}$$

such that deviatoric stress and strain-rate are defined from the isotropic quantities as

$$\tau_{ij} = \sigma_{ij} + p\delta_{ij}, \quad \text{and} \tag{6}$$

$$\dot{\varepsilon}'_{ij} = \dot{\varepsilon}_{ij} - \dot{e}\delta_{ij}. \tag{7}$$

$$\tag{8}$$

Using the constitutive law, and assuming 2-D (x-z space), the Stokes equation can be written as (also see sec. **??**)

$$\frac{\partial}{\partial x}\left(\eta\left(\frac{4}{3}\frac{\partial v_x}{\partial x} - \frac{2}{3}\frac{\partial v_z}{\partial z}\right)\right) + \frac{\partial}{\partial z}\left(\eta\left(\frac{\partial v_x}{\partial z} + \frac{\partial v_z}{\partial x}\right)\right) - \frac{\partial p}{\partial x} = 0 \tag{9}$$

$$\frac{\partial}{\partial z}\left(\eta\left(\frac{4}{3}\frac{\partial v_z}{\partial z} - \frac{2}{3}\frac{\partial v_x}{\partial x}\right)\right) + \frac{\partial}{\partial x}\left(\eta\left(\frac{\partial v_x}{\partial z} + \frac{\partial v_z}{\partial x}\right)\right) - \frac{\partial p}{\partial z} = \rho g_z. \tag{10}$$

Often, we write the constitutive law for deviatoric quantities only,

$$\tau_{ij} = 2\eta\dot{\varepsilon}'_{ij} \quad \text{with} \quad \tau_{ij} = \sigma_{ij} + p = \sigma_{ij} - \sigma_{kk}/3. \tag{11}$$

Incompressibility translates to a constraint on the divergence of the velocity

$$\underline{\nabla} \cdot \underline{v} = 0 \quad \text{or} \quad \frac{\partial v_i}{\partial x_i} = 0, \tag{12}$$

which allows solving eq. (1) for the additional unknown, pressure. For $\underline{\nabla} \cdot \underline{v} = \underline{0}$,

$$tr(\dot{\varepsilon}) = 0 \quad \rightarrow \quad \dot{\varepsilon}' = \dot{\varepsilon}, \tag{13}$$

but we made the distinction between deviatoric and total strain-rate because we numerically only approximate the incompressible continuity equation, eq. (12), by requiring the divergence to be smaller than some tolerance.

There are several approaches to do this (*e.g.* penalty methods (sec. **??**) or Lagrange methods) which typically involve iterations to progressively introduce additional "stiffness" to the medium (sec. **??**). We shall allow for a finite, large bulk viscosity, κ, such that eq. (12) is approximated by

$$\frac{\partial v_x}{\partial x} + \frac{\partial v_z}{\partial z} = -\frac{p}{\kappa}, \tag{14}$$

the right hand side would $\rightarrow 0$ for $\kappa \rightarrow \infty$. Eq. (14) is valid for the incompressible and the compressible cases. However, for the compressible case, where the constitutive law, eq. (2), is replaced by

$$\sigma_{ij} = \kappa \frac{\partial v_k}{\partial x_k} \delta_{ij} + 2\eta \dot{\varepsilon}_{ij}, \tag{15}$$

p cannot be interpreted as the actual pressure, $P = -\sigma_{ii}/3$, rather it is a pressure parameter because

$$P = -(\kappa + 2\eta/3) \frac{\partial v_i}{\partial x_i} \tag{16}$$

and

$$p = -\kappa \frac{\partial v_i}{\partial x_i}. \tag{17}$$

Note: The general, compressible case is identical to the elastic formulation where $\underline{v} \rightarrow \underline{u}$ and the constitutive law is

$$\sigma_{ij} = \lambda \frac{\partial v_k}{\partial x_k} \delta_{ij} + 2\mu \dot{\varepsilon}_{ij}. \tag{18}$$

1.2 Problem in strong form

The (finite element) solution is to be found for the problem stated by eqs. (1) and (14),

$$\frac{\partial \sigma_{ij}}{\partial x_j} + f_i = 0 \tag{19}$$

$$\frac{\partial v_i}{\partial x_i} + \frac{p}{\kappa} = 0 \tag{20}$$

with boundary conditions

$$v_i = g_i \quad \text{on} \quad \Gamma_{g_i} \tag{21}$$

$$\sigma_{ij} n_j = h_i \quad \text{on} \quad \Gamma_{h_i} \tag{22}$$

for velocities and tractions, respectively.

1.2.1 Problem in weak form

The pressure equation modifies the stiffness matrix component such that

$$\int d\Omega\, w_{(i,j)} \sigma_{ij} - \int d\Omega\, q \left(\frac{\partial v_i}{\partial x_i} + \frac{p}{\kappa} \right) = \int d\Omega\, w_i f_i + \sum_i^{n_{sd}} \int_{\Gamma_{h_i}} d\Gamma\, w_i h_i, \tag{23}$$

with n_{sd} the number of spatial dimensions. We again use the Galerkin approach, which leads to the matrix equations.

1.2.2 Matrix assembly

In analogy to the elastic problem, we define a (total) strain-rate vector $\underline{\dot{e}} = \{\dot{\epsilon}_{xx}, \dot{\epsilon}_{zz}, \dot{\gamma}_{xz} = 2\dot{\epsilon}_{xz}\}$ such that strain-rates on an element level can be computed from

$$\underline{\dot{e}} = \underline{\underline{B}}\, \underline{v}, \tag{24}$$

where $\underline{v}$ are velocities given at the element-local nodes, and $\underline{\underline{B}}$ holds the derivatives, as before. When expressed for the local node a and shape functions N_a,

$$\underline{\underline{B}}_a = \begin{pmatrix} \frac{\partial N_a}{\partial x} & 0 \\ 0 & \frac{\partial N_a}{\partial z} \\ \frac{\partial N_a}{\partial z} & \frac{\partial N_a}{\partial x} \end{pmatrix}. \tag{25}$$

Likewise, deviatoric stresses can be computed from $\underline{t} = \underline{\underline{D}}\, \underline{\dot{e}}$, where the property matrix $\underline{\underline{D}}$ shall be given by

$$\underline{\underline{D}} = \eta \begin{pmatrix} \frac{4}{3} & -\frac{2}{3} & 0 \\ -\frac{2}{3} & \frac{4}{3} & 0 \\ 0 & 0 & 1 \end{pmatrix}, \tag{26}$$

for a plane-strain approximation (compare the elastic case). This allows to express the stress vector with pressure part as

$$\underline{s} = -p\underline{m} + \underline{\underline{D}}\, \underline{\dot{e}}, \tag{27}$$

where $\underline{m} = \{1, 1, 0\}$. The deviatoric-only version of $\underline{\underline{D}}$ is

$$\underline{\underline{D}}' = \eta \begin{pmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{28}$$

In analogy to the displacement, $\underline{u}$, representation for the elastic problem, interpolated velocities, $\underline{v}$, are assumed to be given by the summation over the nodal velocities times the shape functions within each element

$$\underline{v}(\underline{x}) \approx \sum_{a=1} N_a(\underline{x})\underline{v}_a. \tag{29}$$

Given the incompressibility constraint, special care has to be taken in the choice of shape functions, and we will use the seven-node, *Crouzeix and Raviart* (1973) triangle with quadratic shape functions N_a (*cf.* *Dabrowski et al.*, 2008).

As detailed in *Hughes* (2000), one can either choose "conforming" elements for the problem at hand and get a nice solution for the velocities and pressure right away (which is what we do here), or choose theoretically inappropriate shape functions and later correct the pressure (*e.g.* for so-called "checkerboard modes"). The latter, rough-and-ready approach may seem less appealing, but works just as well if done properly.

A departure from the elastic problem is that the pressure is treated differently from $\underline{v}$, and we use linear (constant) shape functions for

$$p(\underline{x}) = \sum_{a'} \tilde{N}_{a'}(\underline{x})p_{a'} = \tilde{N}_{a'}p_{a'}, \tag{30}$$

where a' indicates an element-local node, to be distinguished from a which we use for the velocity shape function, and the respective total node number per element may be different (*e.g.* seven for velocities, one for pressure). This approach is called the "mixed formulation". Correspondingly, we introduce an isotropic strain operator $\underline{\underline{B}}_v$, such that

$$\underline{\nabla} \cdot \underline{v} = \dot{\varepsilon}_v = \underline{\underline{B}}_v \underline{v}^e, \tag{31}$$

and $p^e = -\kappa \underline{\underline{B}}_v \underline{v}^e$.

The global system of equations for velocity, $\underline{V}$, and pressure, $\underline{P}$, at the nodes is given by

$$\begin{pmatrix} \underline{\underline{A}} & \underline{\underline{Q}}^T \\ \underline{\underline{Q}} & \underline{\underline{M}} \end{pmatrix} \begin{pmatrix} \underline{V} \\ \underline{P} \end{pmatrix} = \begin{pmatrix} \underline{F} \\ \underline{H} \end{pmatrix}, \tag{32}$$

where $\underline{F}$ are the load vectors, *e.g.* due to body forces, and $\underline{H}$ is due to the divergence that may be imposed traction loads for the compressible case ($\underline{H} = \underline{0}$ for incompressible case).

On an element-level, the stiffness matrix is given by

$$\begin{aligned} \underline{\underline{k}}^e_{ab} &= \int_{\Omega_e} d\Omega \begin{pmatrix} \underline{\underline{A}} & \underline{\underline{Q}}^T \\ \underline{\underline{Q}} & -\frac{1}{\kappa}\underline{\underline{M}} \end{pmatrix} \\ &= \int_{\Omega_e} d\Omega \begin{pmatrix} \underline{\underline{B}}_a^T \underline{\underline{D}}\,\underline{\underline{B}}_b & -\underline{\underline{B}}_v^T \tilde{N}^T \\ -\tilde{N}\underline{\underline{B}}_v & -\frac{1}{\kappa}\tilde{N}_a\tilde{N}_b^T \end{pmatrix}, \end{aligned} \tag{33}$$

i.e. $\underline{\underline{Q}} = -\tilde{N}\underline{\underline{B}}_v$, $\underline{\underline{M}} = \tilde{N}\tilde{N}^T$, and $\underline{\underline{A}}$ corresponds to the total stiffness matrix $\underline{\underline{k}}$ in the elastic case. We have omitted the dependence on the local node number in eq. (33). Note that all operations involving $\underline{\underline{Q}}$ and $\underline{\underline{M}}$ involve the pressure, and not the velocity, shape functions.

We avoid having to actually solve for the global p by using the "static condensation". This means that we locally (element by element) invert $\underline{\underline{M}}$ to obtain the pressure from

$$p \approx \tilde{N}_{a'} p_{a'} = \kappa \tilde{N}^T \left(\underline{\underline{M}}^{-1} \underline{\underline{Q}} \underline{v}^e \right) = -\kappa \underline{\underline{B}}_v \underline{v}^e. \tag{34}$$

(This is not a good idea if combined with iterative solvers.)

We can then simplify eq. (33) to the global, linear equation system

$$\underline{\underline{A}}' \underline{V} = \underline{f}, \tag{35}$$

which is to be solved for the nodal velocities $\underline{V}$. Here, $\underline{f} = \{\underline{f}^e\} = \{\rho^e \underline{g}^e\}$ and (the Schur complement)

$$\underline{\underline{A}}' = \underline{\underline{A}} + \kappa \underline{\underline{Q}}^T \underline{\underline{M}}^{-1} \underline{\underline{Q}}. \tag{36}$$

$\underline{\underline{A}}'$ is now symmetric and positive-definite, and the regular, efficient matrix solution methods can be applied. (Note that the $\underline{\underline{A}}$ is only symmetric if the Dirichlet boundary conditions are applied carefully. If implemented straightforwardly, $\underline{\underline{A}}$ is not symmetric.)

However, the matrix becomes ill-conditioned (hard to invert) for the desired large values of κ, which is why iterations for the velocity solution are needed in order to achieve the incompressibility constraint. Our example code applies "Powell and Hestenes" iterations for the global velocity and pressure vectors $\underline{V}$ and $\underline{P}$ (*cf.* *Dabrowski et al.*, 2008), as in

$$\begin{aligned} &\underline{P}^0 = 0, \quad i = 0 \\ &\text{while } \max(\Delta \underline{P}^i) > \text{tolerance} \\ &\quad \underline{V}^i = (\underline{\underline{A}}')^{-1} (\underline{f} - \underline{\underline{Q}}^T \underline{P}^i) \\ &\quad \Delta \underline{P}^i = \underline{\underline{M}}^{-1} \underline{\underline{Q}} \underline{V}^i \\ &\quad \underline{P}^{i+1} = \underline{P}^i + \Delta \underline{P}^i \\ &\quad i = i + 1 \\ &\text{end} \end{aligned} \tag{37}$$

If and when the algorithm converges, the pressure correction $\Delta \underline{P}^i = \underline{\underline{M}}^{-1} \underline{\underline{Q}} \underline{V}$, which depends on the divergence, $\underline{\underline{M}}^{-1} \underline{\underline{Q}} \underline{V}$, goes to zero. Above, all matrices are meant to be the global, not element-local representation.

1.3 Exercises

a) Make sure you have the common FE MATLAB subroutines from the earlier exercises (`ip_-triangle.m`, `shp_deriv_triangle.m`, `genereate_mesh.m`), and the `triangle` binary in your working directory.

b) Download the `mechanical2d_test.m` driver, and the `mechanical2d_std.m` solver. Inspect both and compare with above for implementation. You will have to fill in the blanks in the driver.

c) Compute the sinking velocity of a dense sphere (*i.e.* disk in 2-D) with radius 0.1 that is centered in the middle of the 1×1 box with free-slip boundary conditions (no shear stress tangentially to the boundary, no motion perpendicular to the boundary) on all sides.

Ensure that the sphere is well resolved by choosing ~ 50 points on its circumference and using a high quality mesh. Use the second order triangles (six nodes on the edges plus one added in the center), and six integration points.

(i) Note how boundary conditions are implemented in the MATLAB code, and comment on essential and natural types.

(ii) Compute the solution for the dense sphere with the same viscosity as the background. Plot the velocities on top of the pressure within the fluid. You may choose whichever absolute parameter values you like but will have to be consistent subsequently.

(iii) Change the number of integration points to three, and replot. Change the type of element to linear, replot. Comment on the velocity and pressure solution.

(iv) The solver applies a finite bulk viscosity (it should be ∞ for an incompressible fluid). For increasing sphere/medium viscosity contrasts upward of 10^3, experiment with increasing the pseudo-incompressibility and comment on the stability of the solution. After this experiment, reset to the starting value.

(v) The solver applies iterations to enforce the incompressibility constrain. Change the tolerance criterion and comment on the resulting velocity and pressure solutions.

(vi) Change back to seven node triangles with six integration points. Plot the vertical velocity, v_z, along a profile for $x \in [0;1]$ at $z = 0.5$.

(vii) Vary the radius of the sphere and comment on how the v_z profiles are affected by the size of the sinker relative to the box size. How small does the sphere have to be to not feel the effect of the boundaries?

(viii) Change the boundary conditions to no-slip ($\underline{v} = \underline{0}$ on all domain edges), replot the vertical velocity profile for a sphere of radius 0.1. Comment. Change back to free-slip subsequently.

(ix) Compute the sinking velocity of a dense sphere with radius 0.1 that is 0.001, 1, and 1,000 times the background viscosity. Define the sinking velocity as the maximum velocity at the sphere's origin at $\underline{x} = \{0.5, 0.5\}$.

(x) Provide an analytical estimate for the sinking velocities and compare with the numerical estimates.

d) Compute the sinking velocities of a highly elliptical (choose ellipticity 0.975, radius 0.25) body whose viscosity is 1,000 times the background viscosity. Investigate the case where this "needle" is oriented horizontally (*i.e.* perpendicular to the sinking velocity at its center) and when it is oriented vertically (*i.e.* aligned with the sinking velocity at its center). Comment on the difference in the maximum sinking velocity between the two elliptical and the spherical cases.

e) *Bonus (somewhat involved)*: Compute the sinking velocity for a non-Newtonian, power-law fluid with $\dot{\varepsilon}'_{II} \propto \tau_{II}^n$ where $n = 3$, and $_{II}$ indicated the second, shear invariants.

Hints: You will have to convert the constitutive law to a viscosity, for which you can assume constant strain-rates. Then, you will have to modify the code to compute the strain-rate tensor to obtain the second invariant, $\dot{\varepsilon}_{II}$. (You might want to check the elastic exercise for the use of $\underline{\underline{D}}$ and $\underline{\underline{B}}$ to obtain strain and stress.) This strain-rate will then enter the viscosity, and you will have to use a second iteration loop, starting with a Newtonian viscosity, then updating the viscosity from the first velocity solution, and repeat until velocities do not change by more than some tolerance.

Bibliography

Crouzeix, M., and P. A. Raviart (1973), Conforming and nonconforming finite elements methods for solving the stationary Stokes equation, *Rev. Franc. d'Automat. Informat. Rech. Opér.*, *3*, 33–76.

Dabrowski, M., M. Krotkiewski, and D. W. Schmid (2008), MILAMIN: MATLAB-based finite element method solver for large problems, *Geochem., Geophys., Geosys.*, *9*(Q04030), doi:10.1029/2007GC001719.

Hughes, T. J. R. (2000), *The finite element method*, Dover Publications.