19th International Congress on Modelling and Simulation, Perth, Australia, 12–16 December 2011
http://mssanz.org.au/modsim2011

FEniCS Framework in Geoscientific Applications

L. Vynnytska [a] **and S.R. Clark** [a]

[a] *Simula Research Laboratory, PO BOX 134, Lysaker, Norway, 1325*
Email: lyudav@simula.no

Abstract: The FEniCS Project is an open source finite element framework for solving PDEs. The advantage of this framework is a tight connection between the mathematical model, the associated variational problem and the numerical algorithm. This connection is achieved by the form language UFL and the user interface DOLFIN.

Using the Python version of the DOLFIN interface, we implement numerical schemes for solving quasi-static Stokes equations with Newtonian and non-Newtonian viscosity accompanied by the heat and the compositional field transer equations in two dimensions. We use a mixed finite element formulation for the incompressible Stokes equations with an independent approximation of velocity and pressure. The advection-diffusion equations for the heat and the composition transfer, with similar mathematical properties, are discretized using an upwinded discontinuous Galerkin formulation, because of the discontinuities and sharp gradients found in these equations. To minimize numerical dispersion error for the compositional field, the discontinuous method is reinforced by a filtering algorithm.

We consider two numerical examples: firstly an isothermal and isoviscous gravitational instability; and secondly thermal convection with a non-Newtonian viscosity. Gravitational instability problem has applications in the modelling of the sinking of solidified magma leading to the formation of basalt layers, while a non-Newtonian power law viscosity is used in studies of mantle convection.

Results presented demonstrate functionality of the FEniCS project and its most impressive feature: comparatively easy way of experimenting with mathematical description of natural processes.

Keywords: FEniCS, finite element method, gravitational instability, non-Newtonian rheology

1 Introduction

Despite a quite long history of geoscience, there are still a lot of processes which are not well understood. One typical example is mantle. How is it structured? Are there significant phase changes in the mantle? What is the viscosity of the mantle and how is it radially distributed? Are compositional differences in the mantle significant enough to drive convection? Is mantle convection layered? Computational techniques were just beginning to allow simulations to tackle these questions (for example, see Hager and O'Connell, 1981; Christensen, 1984), but after three decades significant questions still remain: is there a compositionally different layer of mantle covering the core-mantle boundary? Are the majority of plumes from the core-mantle boundary or from shallower depths? Is there water-enriched mantle at the $670\,km$ discontinuity? What is the thickness and viscosity of the asthenosphere? Are there substantial lateral variations in the mantle? And so on (Ricard, 2009 for a fuller discussion).

Taking into consideration time and spatial scale of the geological processes, it is obvious that numerical modelling together with further comparison of obtained results with observation is an extremely important tool for solving these fundamental problems. However, it is also important to ensure flexibility of software in order to test different hypotheses.

The FEniCS Project is a tool incorporating such properties. It is a collection of open source software which aims to the automated solution of differential equations. FEniCS is based on the finite element method with highly efficient code, and allows a large amount of finite element discretizations. Moreover, its structure presumes that user specifies governing equations and discretization's scheme, while solving of error-prone tasks is automated and, consequently, hidden. User's interface is provided by a problem solving environment DOLFIN with both a C++ and a Python interfaces; it additionally ensures effective dealing with the auxiliary tasks (creating of meshes, specification of linear algebra solvers). The Unified Form Language (UFL), one of the components of FEniCS, is designed for the specification of finite element spaces and variational forms. An essential feature of the UFL is its close connection to mathematical representation. Further details about the FEniCS Project might be found in Logg et al. (2011).

Over long timescales, the mantle behaves like a viscous fluid, driven instantaneously by buoyancy differences caused by temperature or compositional variations. The negligible effect of inertia (Ricard, 2009) means the flow lacks a 'memory' of its previous state and is completely and instantly determined by the instantaneous buoyancy field. The mathematical description of such a fluid is given by the Stokes equations for the conservation of mass and momentum and by advection-diffusion equations for the transfer of heat and composition.

In the paper we consider two problems. The first one is an isothermal gravitational instability with a thin dense layer; it has applications in the modeling of the sinking of solidified magma leading to the formation of basalt layers (Sobouti et al., 2001). The second problem is that of mantle convection with a non-Newtonian power law viscosity, thought to be important in modelling the mantle's rheology (Ranalli, 1995).

2 Solution Procedure with FEniCS

In this section we describe work-flow of solving procedure with FEniCS. Starting from mathematical formulation, we proceed with discretization schemes and aspects of implementation.

2.1 Mathematical Statement

We consider two-dimensional equations in the Cartesian coordinate system Ox_1x_2. Taking into consideration the Boussinesq approximation, Stoke's flow of an incompressible fluid is governed by the following system of non-dimensional equations (Shubert et al., 2001):

$$-\nabla\cdot\sigma - \nabla p = (Rb\,\Phi - Ra\,T)\,e, \tag{1}$$

$$\nabla\cdot u = 0, \tag{2}$$

$$\frac{\partial T}{\partial t} + u\cdot\nabla T = \nabla\cdot\nabla T, \tag{3}$$

$$\frac{\partial \Phi}{\partial t} + u \cdot \nabla \Phi = \nabla \cdot (k_C \nabla \Phi) . \tag{4}$$

Here, equations (1), (2), (3) describe conservation of momentum, mass and energy, respectively. Instead of pure advective equation, transport of chemical compositional filed Φ is modeled by (4) which includes diffusive term with a very small Lewis number k_C. This substitution is convenient for numerical treatment and allows to take into consideration chemical diffusivity presented in real systems (van Keken et al., 1997). In the equations above, u is the velocity vector, p is the pressure, σ denotes the deviatoric stress tensor, T is the temperature field, e is the unit vector in the direction of gravity (opposite to x_2).

The Rayleigh numbers Ra and Rb are defined as

$$Ra = \frac{\alpha \Delta T \Delta \rho g b^3}{\kappa \eta_0}, \quad Rb = \frac{\Delta \rho g b^3}{\kappa \eta_0}, \tag{5}$$

where α is the thermal expansion, ΔT is the temperature contrast, $\Delta \rho$ is the density difference between layers, g is the acceleration of gravity, b is the height of the domain, κ is thermal diffusivity, and η_0 is a reference viscosity. In general, these numbers measure the proneness of the fluid to convect due to thermal (Ra) and compositional (Rb) variations. The choice of scaling parameter η_0 is strongly connected with the constitutive equations. In the case of Newtonian viscosity where stress depends linearly on the strain rate $\dot{\varepsilon}$ as

$$\sigma = 2\eta \dot{\varepsilon}(u), \quad \dot{\varepsilon}(u) = \frac{1}{2}\left(\nabla u + (\nabla u)^T\right), \tag{6}$$

η_0 is mostly equal to the viscosity on the surface. For the non-Newtonian rheology with power n, the reference viscosity reads as $\eta_0 = A^{1/n} b^{(1+n)/n} \kappa^{(1-n)/n}$, where A is a given constant, and non-dimensional viscosity is defined through the second strain rate invariant ε by the expression (Christensen and Yuen, 1989)

$$\eta = \varepsilon^{(1-n)/n}, \quad \varepsilon = \sqrt{\frac{1}{2} \dot{\varepsilon} : \dot{\varepsilon}}. \tag{7}$$

The problems are solved in the rectangular region Ω with the length L and the height H.

Isothermal gravitational (or Rayleigh-Taylor) instability is modeled by the equations (1), (2), (4) with $Ra = 0$, $Rb = 10^5$, $L/H = 1$ and $H_d = 0.2\,H$, where H_d is the height of the dense layer. For this problem we consider Newtonian viscosity which is equal for both layers, that is $\eta = 1$ in (6). Free-slip and impermeable boundary conditions are prescribed on all boundaries together with no influx for the composition. Initial disturbance between layers is caused by the adding of dense layer $0.08\,H \times 0.08\,H$ on the left side of the interface (**Figure 1a**).

Thermal convection problem is considered for the non-Newtonian viscosity (7) with $n = 3$ and is governed by the equations (1), (2), (3), where $Ra = 10^3$. Simulation is performed in the region with $L/H = 2.5$. No-slip boundary conditions are defined on the top and bottom boundaries, while free-slip and impermeability are specified on the sides. Temperature is equal to zero on the top and to one on the bottom, and no heat flux condition on the left and right boundaries. As the initial conditions, we used the solution of the steady-state problem with free-slip and impermeable boundary conditions for the whole region. This solution is obtained by successive iterations starting from initial guesses for the temperature field

$$T_s = (H - x_2)/H + 0.05 \cos(\pi x_1/L) \sin(\pi x_2/H),$$

and the temperature- and depth-dependent viscosity

$$\eta_s = \exp\left(-\ln(16384) T_s + \ln(64)(H - x_2)/H\right).$$

2.2 Discretization Schemes

Numerical treatment of the equations (1) - (4) includes both spatial and temporal discretizations. Spatial discretization of the equations is performed using continuous and discontinuous schemes of the finite element method applied to the correspondent variational forms, while for the solving of time-dependent problems the backward Euler method is used. Further we describe implemented numerical schemes for the incompressible Stokes equation and the steady-state advection-diffusion equation with dominated advective term.

First, let us assume that Ω is partitioned by the triangular mesh $\mathcal{T}_h = \{K\}$. For the discretization of equations (1) - (2), we implemented a mixed variational formulation with the lowest order Taylor-Hood elements (Taylor and Hood, 1973), that is quadratic vector fields for the velocities and continuous piecewise linears for pressure both over $\mathcal{T}_h$. Solving for the advection-dominated advection-diffusion equations (3), (4) in the steady-state case is performed by a discontinuous Galerkin method with an upwinded numerical flux di Pietro et al. (2006); Arnold (1982) on a finite element space of discontinuous piecewise linears over the tessellation $\mathcal{T}_h$. Further details might be found in Vynnytska et al. (2011).

2.3 Solving Algorithms and Some Implementation Details

The problems are solved using splitting scheme. We seek first the solution of the incompressible Stokes equation, and then solve for the solution of time-dependent equation using the backward Euler method. For the Rayleigh-Taylor instability, we apply additionally filtering algorithm for the composition Φ aimed for the correction of numerical dispersion (Lenardic and Kaula, 1993). New time step is computed to satisfy CFL condition, which relates spatial and temporal discretizations.

The solving algorithms are implemented using the Python user interface to DOLFIN version 0.9.10. The solutions are obtained on built-in meshes. They are formed by partition of the region on $n \times m$ rectangles and further dividing of each rectangle on two (using only one diagonal) of four (using both diagonals) triangles.

To demonstrate particularities and possibilities of FEniCS, let us consider the implemented Picard iteration algorithm for solving the equations (1)- (2) with the viscosity (7) which is presented in **Listing 1**. As input parameters, among other mesh, boundary conditions and solutions from the previous time step are given. The procedure starts from specification of finite elements spaces V, Q, W and definition of basis (both test and trial) and other functions. In the iteration loop, we define viscosity η using mid-point solution from two previous iterations for the velocity field u. A mixed variational formulation is defined by the form $F((u, p), (v, q))$. The last step is assembling and solving of the system of linear equations. The solution is obtained using the default solver.

Listing 1: Solving algorithm for the incompressible Stokes equation with non-Newtonian viscosity

```
def momentum_solve(nR, g, Ra, mesh, bcs, T_pr, u_pr):
    ”””
    Input parameters:
      nR - power in viscosity expression; g - acceleration of gravity
      Ra - Rayleigh number; bcs - boundary conditions; mesh
      T_pr, u_pr - temperature and velocities from previous time step
    ”””

    # definition of finite element spaces
    V = VectorFunctionSpace(mesh, ”CG”, 2)
    Q = FunctionSpace(mesh, ”CG”, 1)
    W = V * Q

    (u, p) = TrialFunctions(W); (v, q) = TestFunctions(W) # basis functions

    velocity_pressure = Function(W)
    u_mid = Function(V); u1 = Function(V)
    u1.assign(u_pr) # assign previous solution
    H = Ra*T_pr

    er = 1.0; eps = 1e-4 # quantities which are used in stopping criterion
    # start Picard's iteration
    while er > eps:
```

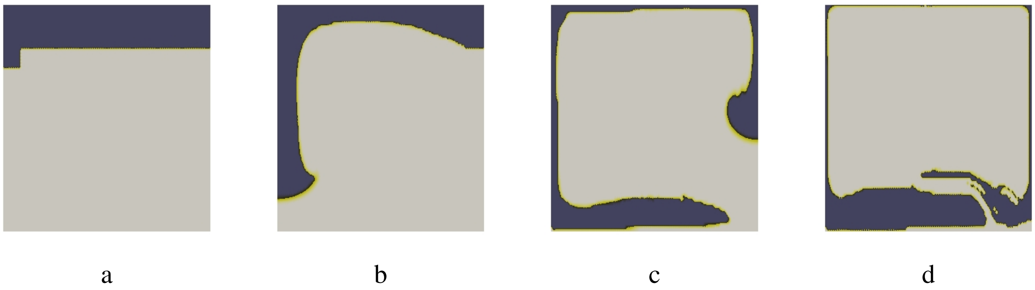


Figure 1: Gravitational instability: a) initial setup, t = 0; b) sinking plume on the left wall, $t = 0.00127$; c) sinking plume on the right wall, $t = 0.00369$; d) final stage, $t = 0.0531$

```
        u_mid = 0.5*(u_pr + u1) # define mid-point solution
        sri = sqrt(0.5*inner(strain(u_mid), strain(u_mid))) # II SR invariant
        eta =(sri**((1.0-nR)/nR)) # compute eta

        # Define equation F((u, p), (v, q)) = 0
        F = (2.0*eta*inner(strain(u), strain(v))*dx + div(v)*p*dx
          + div(u)*q*dx + H*inner(g, v)*dx)
        a = lhs(F); L = rhs(F) # geather left- and right-hand sides of F

        (A, b) = assemble_system(a, L, bcs) # assebble system
        solve(A, velocity_pressure.vector(), b) # solve system

        u_pr.assign(u1) # update previous solution
        u1.assign(velocity_pressure.split()[0]) # assign new solution
        er = abs(norm(u_pr) - norm(u1))/norm(u_pr) # compute relative error

    return u1 # return solution
```

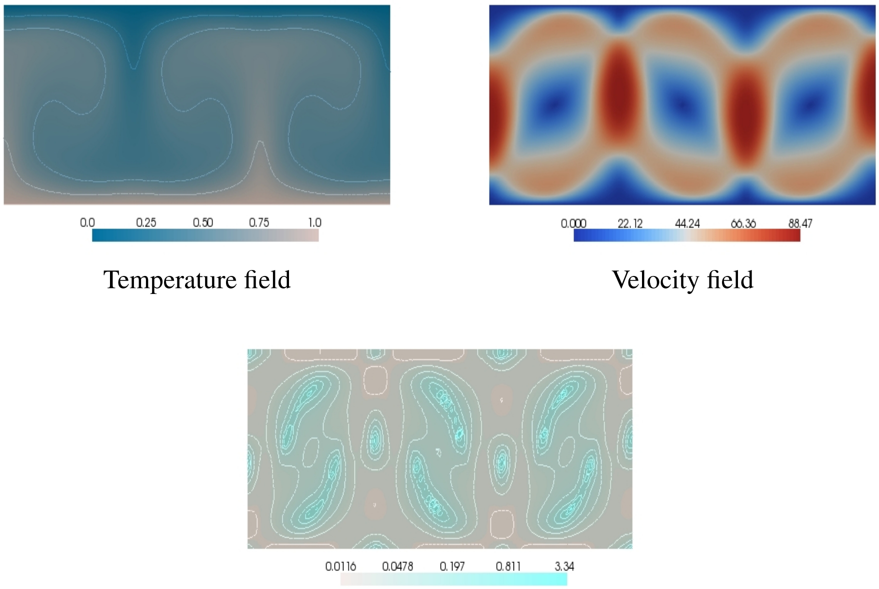
3 Numerical Results

3.1 Gravitational Instability

For this problem, we set $k_C = 10^{-8}$ to make influence of diffusion negligible, Φ is a step function which is equal to one for the dense layer and to zero for the light layer. The triangular mesh is constructed by the diagonal from the lower left corner to the upper right corner on the initial rectangular partition 100×70, that is the mesh is denser in x_1 direction. Simulation demonstrates the following scenario. Distortion of the interface causes gradients in pressure and formation of the plume on left side (**Figure 1b**) which moves through the light layer. On the further stages pressure gradient leads to the formation of smaller sinking plume structure near the right wall (**Figure 1c**); we stop the simulation when almost steady regime is reached (**Figure 1d**).

3.2 Thermal Convection

This problem is solved on the mesh 100×40 with crossed diagonals. We followed Christensen and Yuen (1989) in initial setup in order to compare results and verify numerical method. In particular, in Christensen and Yuen (1989) for this problem is obtained breakup of a long convection cell and subsequent transition to three similar and almost steady convective cells. Our scheme led us to the same conclusion. The simulation results for temperature, velocity, and viscosity fields are shown in **Figure 2**.



Temperature field

Velocity field

Viscosity field

Figure 2: Thermal convection with non-Newtonian viscosity: three convective cells, t = 0.32941

4 Conclusions

This paper demonstrates an applicability of general-purpose finite element library FEniCS for solving geoscientific problems. Using functionality of the FEniCS project, first of all the form language UFL, we implemented variational problems describing evolution of an incompressible Stokes flow induced by compositional or temperature variations. Numerical results show the ability of discontinuous Galerkin scheme to track interface between chemically distinct layers. However, in the case of very thin layer special-purpose tracer-based approaches should be applied. At the same time, DOLFIN allows to combine FEniCS functionality and additional code for specific purposes. Solving for the problem with non-Newtonian viscosity is implemented using the Picard iteration method, however FEniCS offers the possibility to use non-linear solvers. Despite problems considered are two-dimensional, FEniCS works in the same way in three dimensional case; it provides also parallel simulations.

Acknowledgements

The work of the authors is supported by a Statoil research grant.

References

Arnold, D. (1982). An interior penalty finite element method with discontinuous elements. *SIAM Journal on Numerical Analysis 19*(4).

Christensen, U. (1984). Heat-transport by variable viscosity convection and implications for the Earth's thermal evolution. *Physics of the Earth and Planetary Interiors 35*, 264–282.

Christensen, U. and D. Yuen (1989). Time-dependent convection with non-Newtonian viscosity. *Journal of Geophysical Research 94*(B1), 814–820.

di Pietro, D., S. L. Forte, and N. Parolini (2006). Mass preserving finite element implementations of the level set method. *Applied Numerical Mathematics 56*, 1179–1195.

Hager, B. and R. O'Connell (1981). A simple global model of plate dynamics and mantle convection. *Journal of Geophysical Research 86*(B6), 4843–4867.

Lenardic, A. and W. Kaula (1993). A numerical treatment of geodynamic viscous flow problems involving the advection of material interfaces. *Journal of Geophysical Research 98*(B5), 8243–8260.

Logg, A., K.-A. Mardal, and G. N. Wells (Eds.) (2011). *Automated solution of differential equation by the finite element method*. Springer. To appear.

Ranalli, G. (Ed.) (1995). *Rheology of the Earth*. Springer.

Ricard, Y. (2009). Physics of Mantle Convection. In D. Bercovici (Ed.), *Mantle Dynamics*, Treatise on Geophysics, pp. 1–31. Elsevier.

Shubert, G., D. Turcotte, and P. Olson (2001). *Mantle convection in the Earth and planets*. Cambridge University Press.

Sobouti, F., A. Ghods, and J. Arkani-Hamed (2001). On the advection of sharp material interfaces in geodynamic problems: Entrainment of the D" layer. *Journal of Geodynamics 31*, 459–479.

Taylor, C. and P. Hood (1973). A numerical solution of the Navier-Stokes equations using the finite element technique. *Computers & Fluids 1*(1), 73–100.

van Keken, P., S. King, H. Schmeling, et al. (1997). A comparison of methods for the modeling of thermochemical convection. *Journal of Geophysical Research 102*(B10), 22477–22495.

Vynnytska, L., S. Clark, and M. Rognes (2011). Dynamic simulations of convection in the Earth's mantle. In A. Logg, K.-A. Mardal, and G. N. Wells (Eds.), *Automated solution of differential equations by the finite element method*. Springer. To appear.