Fluids modelling

Vladislav Pimanov
Anton Zhevnerchuk
`https://github.com/zhevnerchuk/Fluids-modelling`

Computational Science and Engineering II
Discretization
Skoltech, Spring 2018

Problem Formulation

Skoltech

Navier-Stokes Equation (NSE)

$$\begin{cases} \frac{\partial \mathbf{u}}{\partial t} + \nabla \mathbf{u} \cdot \mathbf{u} - \nu \Delta \mathbf{u} + \nabla p = f \\ \nabla \cdot \mathbf{u} = 0 \end{cases}$$

Stationary Stokes Equation (SSE)

$$\begin{cases} -\nu \Delta \mathbf{u} + \nabla p = f \\ \nabla \cdot \mathbf{u} = 0 \end{cases}$$

$\mathbf{u}$ – velocity, p – pressure, f – source term, ν – viscosity

Weak formulation

The abstract problem we investigate

$$\begin{cases} \text{Find } \mathbf{u} \in V, p \in Q \text{ such that:} \\ a(\mathbf{u}, \mathbf{v}) + b(\mathbf{v}, p) = f(\mathbf{v}), \ \forall \mathbf{v} \in V \\ b(\mathbf{u}, q) = 0, \ \forall q \in Q \end{cases}$$

Stokes settings

$$\begin{cases} V = [H_0^1(\Omega)]^d \text{– velocity space,} \\ Q = L^2_{\int=0}(\Omega) \text{– pressure space,} \\ a(\mathbf{u}, \mathbf{v}) = \int_\Omega \nabla \mathbf{u} : \nabla \mathbf{v}, \\ b(\mathbf{v}, p) = -\int_\Omega p \nabla \cdot \mathbf{v}, \\ f(\mathbf{v}) = \int_\Omega \mathbf{v} f. \end{cases}$$

Skoltech

$$\begin{cases} \text{Find } u \in W \text{ such that:} \\ c(u, v) = f(v), \ \forall v \in V \end{cases}$$

- V and W are Banach spaces, V – reflexive
- $c(\cdot, \cdot) \in \mathcal{L}(W \times V, \mathbb{R})$ – continious bilinear form on $W \times V$
- $f(\cdot) \in \mathcal{L}(V, \mathbb{R})$ – continious linear form on V

When the problem is well-posed?

In the case when V = W, we need only coercivity of $c(\cdot, \cdot)$:

$$\exists \alpha > 0,\ \forall u \in V,\ c(u,u) \geq \alpha ||u||_V. \qquad (LM)$$

In the general case we need the Banach-Necas-Babuska:

$$\exists \alpha > 0,\ \inf_{w \in W} \sup_{v \in V} \frac{c(w,v)}{||w||_W ||v||_V} \geq \alpha. \qquad (BNB_1)$$

$$\forall v \in V,\ (\forall w \in W,\ c(w,v) = 0) \Rightarrow (v = 0). \qquad (BNB_2)$$

Note: LM implies BNB_1 and BNB_2.

LBB Conditions

Back to the Stokes problem

$$\begin{cases} \text{Find } \mathbf{u} \in V, p \in Q \text{ such that:} \\ a(\mathbf{u}, \mathbf{v}) + b(\mathbf{v}, p) = f(\mathbf{v}), \ \forall \mathbf{v} \in V \\ b(\mathbf{u}, q) = 0, \ \forall q \in Q \end{cases}$$

Ladyzhenskaya-Babuska-Brezzi Conditions

Using *BNB* theorem the following condition can be obtained:

$$\exists \beta > 0, \ \inf_{q \in Q} \sup_{\mathbf{v} \in V} \frac{b(\mathbf{v}, q)}{||\mathbf{v}||_V ||q||_Q} \geq \beta. \qquad (LBB)$$

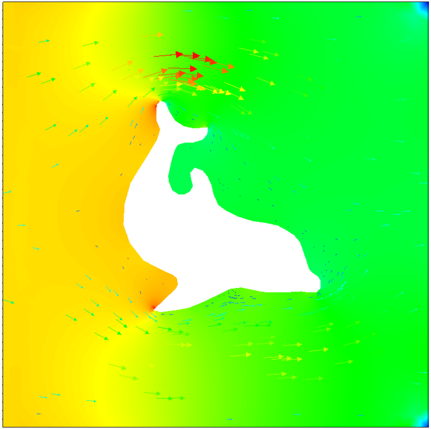
$$\begin{cases} V = [H_0^1(\Omega)]^d - \text{ velocity space,} \\ Q = L^2_{\int = 0}(\Omega) - \text{ pressure space.} \end{cases}$$

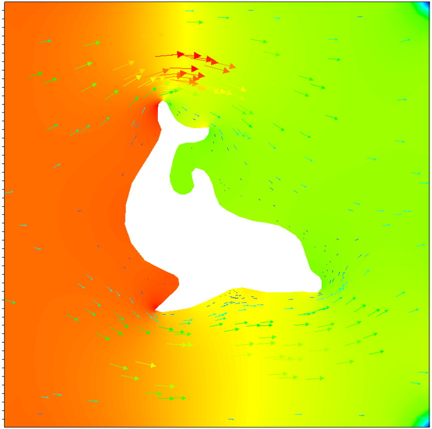
What about discrete case? How to choose FEM subspaces?

FEM Discretization

- $V \to V_h$ – discretize the velocity space
- $Q \to Q_h$ – discretize the pressure space
- LBB condition has to be satisfied
- Conformity does not imply LBB
- General issue for saddle-point problems
- Stokes problem – Taylor-Hood elements

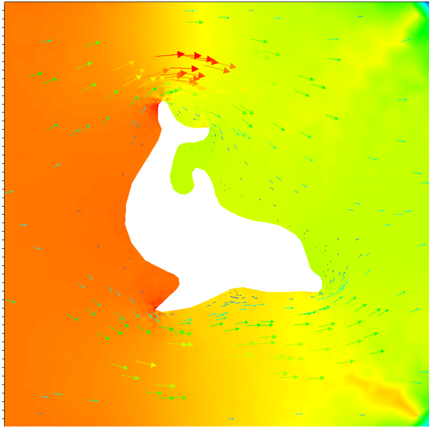
LBB for Stokes demo

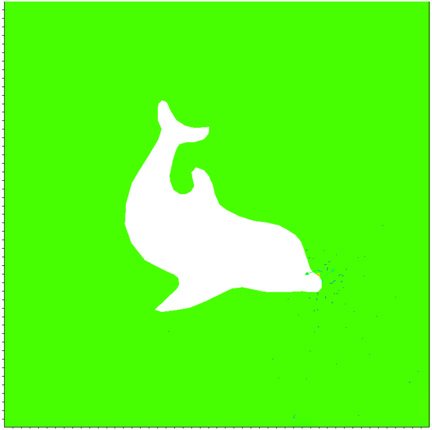




Velocity: 2-degree polynomials
Pressure: 1-degree polynomials

Velocity: 3-degree polynomials
Pressure: 2-degree polynomials





Velocity: 2-degree polynomials
Pressure: 2-degree polynomials

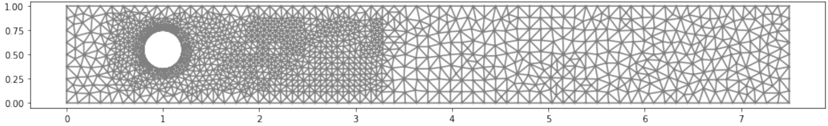
Velocity: 2-degree polynomials
Pressure: 3-degree polynomials

Problem formulation

$$\begin{cases} \frac{\partial \mathbf{u}}{\partial t} + \nabla \mathbf{u} \cdot \mathbf{u} - \nu \Delta \mathbf{u} + \nabla p = f \\ \nabla \cdot \mathbf{u} = 0 \end{cases}$$

Boundary conditions:

- $v|_{\Gamma_1} = (0, 0)$ (No-slip BC)
- $v|_{\Gamma_2} = v|_{\Gamma_3} = (cy(1-y), 0)$ (Poiseuille BC)
- $p|_{\Gamma_3} = 0$



Discretization Scheme

Chorin's Projection Method

Suppose we have $u^{(n)}$ and $p^{(n)}$.

- Find u^* from

$$\frac{u^* - u^{(n)}}{\Delta t} = \nu \Delta u^{(n)} - \nabla u^{(n)} \cdot u^{(n)}$$

- Find $p^{(n+1)}$ from

$$\Delta p^{(n+1)} = \frac{1}{\Delta t} \nabla \cdot u^*$$

- Find $u^{(n+1)}$ from

$$\frac{u^{(n+1)} - u^*}{\Delta t} = -\nabla p^{n+1}$$

Skoltech

Chorin's Projection Method

Finally we obtain

$$\frac{u^{(n+1)} - u^{(n)}}{\Delta t} = \nu \Delta u^{(n)} - \nabla u^{(n)} \cdot u^{(n)} - \nabla p^{n+1}$$

Chorin's projection method closely related to Helmholtz-Hodge decomposition.

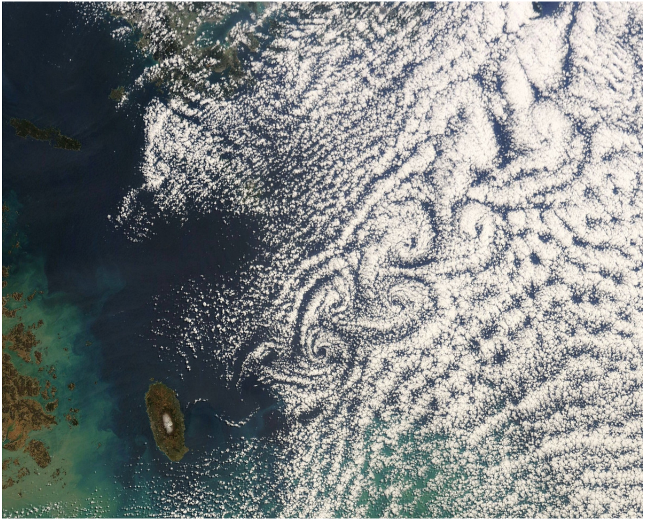
Helmholtz-Hodge decomposition

Vector field can be uniquely decomposed into a divergence-free part and an irrotational part:

$$u = u_{\mathrm{sol}} + u_{\mathrm{irrot}}$$

$$\nabla \cdot u_{\mathrm{sol}} = 0, \quad \nabla \times u_{\mathrm{irrot}} = 0$$

Von Karman Vortex Street



Implementation

- Python. Using DOLFIN, Python interface of FEniCS
- FEniCS: open-source package for solving PDEs
  1. Easy mesh-refinement
  2. Automatically setting up stiffness matrix from the weak form of equation and using elements
  3. Linear solvers with common preconditioners
- Visualization: VisIT