FEniCS Course

Lecture 9: Incompressible Navier–Stokes

Contributors
Anders Logg

The incompressible Navier–Stokes equations

$$
\begin{aligned}
\rho(\dot{u} + u \cdot \nabla u) - \nabla \cdot \sigma(u, p) &= f && \text{in } \Omega \times (0, T] \\
\nabla \cdot u &= 0 && \text{in } \Omega \times (0, T] \\
u &= g_{\mathrm{D}} && \text{on } \Gamma_{\mathrm{D}} \times (0, T] \\
\sigma \cdot n &= g_{\mathrm{N}} && \text{on } \Gamma_{\mathrm{N}} \times (0, T] \\
u(\cdot, 0) &= u_0 && \text{in } \Omega
\end{aligned}
$$

- u is the fluid velocity and p is the pressure
- ρ is the fluid density
- $\sigma(u, p) = 2\mu\epsilon(u) - pI$ is the Cauchy stress tensor
- $\epsilon(u) = \frac{1}{2}(\nabla u + (\nabla u)^{\top})$ is the symmetric gradient
- f is a given body force per unit volume
- g_{D} is a given boundary displacement
- g_{N} is a given boundary traction
- u_0 is a given initial velocity

Variational problem

Multiply the momentum equation by a test function v and integrate by parts:

$$\int_\Omega \rho(\dot{u}+u\cdot\nabla u)\cdot v \,\mathrm{d}x + \int_\Omega \sigma(u,p) : \epsilon(v)\,\mathrm{d}x = \int_\Omega f\cdot v\,\mathrm{d}x + \int_{\Gamma_{\mathrm{N}}} g_{\mathrm{N}}\cdot v\,\mathrm{d}s$$

Short-hand notation:

$$\langle \rho\dot{u}, v\rangle + \langle \rho u\cdot\nabla u, v\rangle + \langle \sigma(u,p), \epsilon(v)\rangle = \langle f, v\rangle + \langle g_{\mathrm{N}}, v\rangle_{\Gamma_{\mathrm{N}}}$$

Multiply the continuity equation by a test function q and sum up: find $(u,p)\in V$ such that

$$\langle \rho\dot{u}, v\rangle + \langle \rho u\cdot\nabla u, v\rangle + \langle \sigma(u,p), \epsilon(v)\rangle + \langle \nabla\cdot u, q\rangle = \langle f, v\rangle + \langle g_{\mathrm{N}}, v\rangle_{\Gamma_{\mathrm{N}}}$$

for all $(v,q)\in\hat{V}$

Discrete variational problem

Time-discretization leads to a *saddle-point* problem on each time step:

$$\begin{bmatrix} M + \Delta t A + \Delta t N(U) & \Delta t B \\ \Delta t B^\top & 0 \end{bmatrix} \begin{bmatrix} U \\ P \end{bmatrix} = \begin{bmatrix} b \\ 0 \end{bmatrix}$$

- Efficient solution of the saddle-point problem relies on the efficiency of special-purpose preconditioners (Uzawa iteration, Schur complement preconditioners, . . .)
- We will use another approach (simpler and often more efficient)

A splitting method

cG(1) / Crank-Nicolson approximation with explicit convection:

$$\rho D_t u^n + \rho u^{n-1} \cdot \nabla u^{n-1} - \nabla \cdot \sigma(u^{n-1/2}, p^{n-1/2}) = f^{n-1/2}$$

Compute the *tentative velocity* $u^\star$ using the approximation

$$\rho D_t u^\star + \rho u^{n-1} \cdot \nabla u^{n-1} - \nabla \cdot \sigma(u^{n-1/2}, {\color{red} p^{n-3/2}}) = f^{n-1/2}$$

Subtract:

$$\rho(D_t u^n - D_t u^\star) - \nabla \cdot \sigma(0, p^{n-1/2} - p^{n-3/2}) = 0$$

Expand and rearrange:

$$\rho(u^n - u^\star) + k_n \nabla(p^{n-1/2} - p^{n-3/2}) = 0$$

$D_t u = (u^n - u^{n-1})/k_n$ and $k_n = t_n - t_{n-1}$

A splitting method (contd.)

We have found that:

$$\rho(u^n - u^\star) + k_n \nabla(p^{n-1/2} - p^{n-3/2}) = 0$$

It follows that

$$\rho u^n = \rho u^\star - k_n \nabla(p^{n-1/2} - p^{n-3/2}) \tag{1}$$

Take the divergence and set $\nabla \cdot u^n = 0$:

$$-k_n \Delta p^{n-1/2} = -k_n \Delta p^{n-3/2} - \rho \nabla \cdot u^\star \tag{2}$$

- Compute $p^{n-1/2}$ by solving the Poisson problem (2)
- Compute u^n by solving the projection problem (1)
- To consider: *What about the boundary conditions for the Poisson problem (2)?*

Boundary conditions

- For outflow boundary conditions, corresponding to so-called "do-nothing" boundary conditions for the Laplacian formulation, we take $\partial_n u = 0$:

$$\begin{aligned}\sigma(u,p)\cdot n &= (2\mu\epsilon(u) - pI)\cdot n = \mu\nabla u\cdot n + \mu(\nabla u)^\top\cdot n - pn\\ &= \mu(\nabla u)^\top\cdot n - pn \approx \mu(\nabla u^{n-1/2})^\top\cdot n - p^{n-3/2}n\end{aligned}$$

- Boundary conditions for the pressure Poisson problem:

$$\partial_n \dot{p} = 0$$

on the pressure Neumann boundary

Incremental pressure correction scheme (IPCS)

1. Compute the tentative velocity $u^{\star}$ by

$$\langle \rho D_t^n u^{\star}, v \rangle + \langle \rho u^{n-1} \cdot \nabla u^{n-1}, v \rangle + \langle \sigma(u^{n-\frac{1}{2}}, p^{n-3/2}), \epsilon(v) \rangle$$
$$- \langle \mu n \cdot (\nabla u^{n-\frac{1}{2}})^{\top}, v \rangle_{\partial\Omega} + \langle p^{n-3/2} n, v \rangle_{\partial\Omega} = \langle f^{n-1/2}, v \rangle$$

2. Compute the corrected pressure $p^{n-1/2}$ by

$$k_n \langle \nabla p^{n-1/2}, \nabla q \rangle = k_n \langle \nabla p^{n-3/2}, \nabla q \rangle - \langle \rho \nabla \cdot u^{\star}, q \rangle$$

3. Compute the corrected velocity u^n by

$$\langle \rho u^n, v \rangle = \langle \rho u^{\star}, v \rangle - k_n \langle \nabla (p^{n-1/2} - p^{n-3/2}), v \rangle$$

Useful FEniCS tools (I)

Note ∇ vs. ∇:

```
dot(grad(u), u)
dot(u, nabla_grad(u))
```

Defining operators:

```
def sigma(u, p):
    return 2.0*mu*sym(grad(u)) - p*Identity(2)
```

The facet normal n:

```
n = FacetNormal(mesh)
```

Useful FEniCS tools (II)

Assembling matrices and vectors:

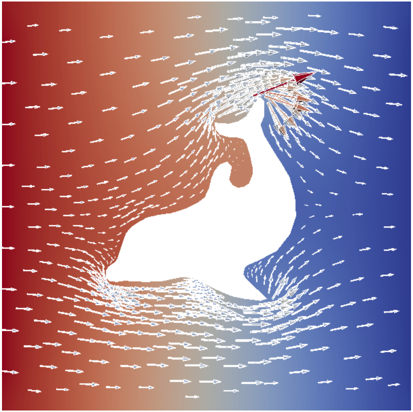
```
A = assemble(a)
b = assemble(L)
```

Solving linear systems:

```
solve(A, x, b)
solve(A, x, b, "gmres", "ilu")
solve(A, x, b, "cg", "amg")
```

The FEniCS challenge!

Solve the incompressible Navier–Stokes equations for the flow of water around a dolphin. The water is initially at rest and the flow is driven by a pressure gradient.



The FEniCS challenge!

- Compute the solution on the time interval $[0, 0.1]$ with time steps of size $k = 0.0005$
- Set $p = 1$ kPa at the inflow and $p = 0$ at the outflow
- The density of water is $\rho = 1000\ \mathrm{kg/m^3}$ and the viscosity is $\mu = 0.001002\ \mathrm{kg/(m \cdot s)}$
- To check your answer, compute the average velocity in the x-direction.

The student(s) who first produce the right anwswer will be rewarded with an exclusive FEniCS coffee mug!