Math 5040 (CFD) Final Project

Due 4 PM on May 5, 2017. NO LATE SUBMISSIONS.

Let $\Omega \subset \mathbb{R}^2$ be the backward-facing step domain, as described in the project code on the course website. The top and bottom walls are denoted by Γ_1, the outflow boundary by Γ_2 and the inflow boundary by Γ_3. You will add to the provided code to solve the following problem:

$$
\begin{aligned}
-\nu\Delta u + u\cdot\nabla u + \nabla p &= f, \text{ in } \Omega, && (1)\\
\nabla\cdot u &= 0, \text{ in } \Omega, && (2)\\
u|_{\Gamma_1} &= g_1(x,y), && (3)\\
2\nu\hat{n}\cdot\mathbf{D}(u) - p\hat{n} &= 0, \text{ on } \Gamma_2 && (4)\\
u|_{\Gamma_3} &= g_3(x,y). && (5)
\end{aligned}
$$

Derive the weak problem as discussed in class. We will define a weak Galerkin-FEM problem with some additional features. First, the explicit skew-symmetry technique must be modified, since we do not have simply $u\cdot\hat{n} = 0$ at the outflow; instead, use the trilinear term

$$
b^*(u,v,w) \equiv \frac{1}{2}(u\cdot\nabla v, w) - \frac{1}{2}(u\cdot\nabla w, v) + \frac{1}{2}\int_{\Gamma_2}(u\cdot\hat{n})\, v\cdot w\, dy.
$$

For the problem we will solve, we will expect that

$$
\frac{1}{2}\int_{\Gamma_2}(u\cdot\hat{n})\,|u|^2\, dy \geq 0,
$$

since Γ_2 is the outflow boundary, thus

$$
b^*(u_h, u_h, u_h) \geq 0,
$$

even for the discrete approximation u_h, which may help the stability.

Also, you will implement "grad-div stabilization" to help reduce the size of the divergence of the discrete velocity. Since the NSE solution is divergence-free, we have $-\gamma\nabla(\nabla\cdot u) = 0$. The value $\gamma > 0$ is a constant, called the *grad-div parameter*. Add this into the NSE (for free). The corresponding weak terms are:

$$
\gamma(\nabla\cdot u, \nabla\cdot v),
$$

which is NOT just zero when we discretize;

$$\gamma(\nabla \cdot u_h, \nabla \cdot v_h) \neq 0.$$

However, as $\gamma \to \infty$, $\nabla \cdot u_h \to 0$.

The first steps for you are as follows:

1. Add in the nonlinear terms in the code to perform a fixed-point iteration; find $u_h^{(n+1)}$ from $u_h^{(n)}$ with
$$b^*\left(u_h^{(n)}, u_h^{(n+1)}, v_h\right).$$
2. Add in the grad-div terms.
3. Add in code to calculate $\|\nabla \cdot u_h\|$ and print the value at the end.

You will then need to "verify" that the code works. We proceed via the method of manufactured solutions; one chooses a solution as follows. Define

$$\begin{aligned} C &= \frac{1}{8000} \\ u &= C\left\langle 2y - 1 + (20-x)^3, 3y(20-x)^2 - 2x \right\rangle, \\ \text{and } p &= C\,(20-x)(2y-1). \end{aligned}$$

This will be a unique solution if we simply define the data f, g_1 and g_3 by inserting our choices of u and p into (1), (3) and (5). This is already implemented in your code. One may easily verify that the remaining equations also hold, but you need to pay attention to the definition of the domain in the code to see why. Note that the velocity is a cubic polynomial and the pressure quadratic. In the code, $P3 - P2$ Taylor-Hood is implemented.

If you completed the code correctly, then upon running it with initial guess $u_h^{(0)} = u$ (true solution), you should see the errors for velocity are on the order of round-off, meaning around 10^{-16} or so, and the pressure only a bit larger (due to the penalization technique). The divergence measurement will be a bit larger still, due to taking a square-root of a small number. **Be careful not to edit anything that you do not need to, which could introduce bugs in your code.** The parameter values are already set, except set $\gamma = 1$. Then, take $u_h^{(0)} = 0$ and verify that the code converges. The errors should be on the order of 10^{-10} or smaller. This gives you an idea of the best accuracy you can expect to achieve with the parameters as set in the code.

After verifying your code, create a copy with a new name. I will want to see both copies, ultimately. You will perform the following tests, modifying the new code accordingly. In all cases, set $f = 0$, $g_1 = 0$ and

$$g_3(x, y) \equiv \langle 12(1-y)(2y-1), 0 \rangle.$$

This is a problem for flow over a backward-facing step, neglecting gravity. Perform all of the following tests and answer all questions as indicated. Lengthy answers are not necessary. Send me an e-mail with both versions of your code and the answers to the questions.

Test 1

Keep all parameters as set in the code, except compare the results with $\gamma = 0$ and with $\gamma = 10^j$ for $j = 0, 1, 2, 3, 4$. The plots will look roughly the same, but how does $\|\nabla \cdot u_h\|$ scale with γ?

Test 2

Now decrease the viscosity to $\nu = 2 \cdot 10^{-3}$ and set $\gamma = 10^4$. How does the solution change with the smaller viscosity? What about the required number of iterations for convergence?