Project 1 - Flow in a driven cavity and non-conforming mesh coupling

Numerics for Fluids, Structures and Electromagnetics

Mathieu Grondin, Moritz Waldleben

January 2022

Introduction

We consider the following Stokes problem on the domain $\Omega \subset \mathbf{R}^2$ for the velocity $\mathbf{u} : \Omega \to \mathbb{R}^2$ and the pressure $p : \Omega \to \mathbb{R}$:

$$\begin{aligned} -\Delta \mathbf{u} + \nabla p &= \mathbf{f} \quad \text{in } \Omega \\ \text{div}\mathbf{u} &= 0 \quad \text{in } \Omega \\ \mathbf{u} &= \mathbf{g} \quad \text{on } \partial\Omega \end{aligned} \tag{1}$$

where $\mathbf{f} : \Omega \to \mathbb{R}^2$ and $\mathbf{g} : \partial\Omega \to \mathbb{R}^2$ are two given function. In particular assume that $\mathbf{f} \in [H^{-1}(\Omega)]^2 = \mathbf{H}^{-1}(\Omega)$ and $\mathbf{g} \in [H^{1/2}(\partial\Omega)]^2 = \mathbf{H}^{1/2}(\partial\Omega)$.

This problem corresponds to solving a flow of an incompressible fluid where viscous forces dominate inertial forces (i.e. low Reynolds number). The equations are essentially stationary and linearized Navier-Stokes equations neglecting inertial terms. The dynamic viscosity is set to 1 in this formulation of the problem.

1 Question 1

Suppose, ab absurdo, that there exists a solution $\mathbf{u}$ to problem 1 but $\int_{\partial\Omega} \mathbf{g} \cdot \mathbf{n} \neq 0$. It results that :

$$\int_{\partial\Omega} \mathbf{g} \cdot \mathbf{n} = \int_{\partial\Omega} \mathbf{u} \cdot \mathbf{n} = \int_{\Omega} \text{div}\mathbf{u} = \int_{\Omega} 0 = 0$$

We used the fact that $\mathbf{u}$ is a solution to the equation. Furthermore the divergence theorem was applied. This results however in a contradiction to the assumption. Hence this proves that $\int_{\partial\Omega} \mathbf{g} \cdot \mathbf{n} = 0$ is a necessary condition for the existence of a solution.

From a physical point of view, this corresponds to the fact that there can be no net flux outwards from the domain Ω as the flow has no source of fluid inside the domain.

Question 2

Weak formulation

Problem 1 has to be expressed in a weak formulation. To do so we multiply the first equation by a test function $\mathbf{v}$ and integrate over the domain Ω. Hence we get :

$$\begin{aligned}
&\int_\Omega -\Delta \mathbf{u}\,\mathbf{v} + \int_\Omega \nabla p\,\mathbf{v} = \int_\Omega \mathbf{f}\,\mathbf{v} \\
\Leftrightarrow &\int_\Omega \nabla \mathbf{u}{:}\nabla \mathbf{v} - \int_{\partial\Omega} (\nabla \mathbf{u}\cdot n)\,\mathbf{v} - \int_\Omega p\,\mathrm{div}\mathbf{v} + \int_{\partial\Omega} p\,(\mathbf{v}\cdot\mathbf{n}) = \int_\Omega \mathbf{f}\,\mathbf{v} \\
\Leftrightarrow &\int_\Omega \nabla \mathbf{u}{:}\nabla \mathbf{v} - \int_\Omega p\,\mathrm{div}\mathbf{v} = \int_\Omega \mathbf{f}\,\mathbf{v} + \int_{\partial\Omega} (\nabla \mathbf{g}\cdot\mathbf{n})\,\mathbf{v} - \int_{\partial\Omega} p\,(\mathbf{v}\cdot\mathbf{n})
\end{aligned}$$

In the development above we did an integration by parts and then used the fact that $\mathbf{u} = \mathbf{g}$ on $\partial\Omega$.

Analogously we multiply the second equation by another test function (q) and integrate over Ω , using the same properties the equation becomes :

$$-\int_\Omega q\,\mathrm{div}\mathbf{u} + \int_{\partial\Omega} q\,(\mathbf{u}\cdot\mathbf{n}) = 0$$

Functional spaces

It remains the question of which are the functional spaces to be imposed for the solution of our problem. In the first equation the gradient of the test function $\mathbf{v}$ appears. Thus we want to make the integral for the test function vanish on the boundary. A natural choice is : $[H_0^1(\Omega)]^2 = \mathbf{H}_0^1(\Omega)$ This space is formally defined by :

$$\mathbf{H}_0^1(\Omega) = \left\{\mathbf{u}\ :\ \mathbf{u}\in L^2(\Omega), \nabla\mathbf{u}\in L^2(\Omega), \text{and} \mathbf{u}|_{\partial\Omega} = 0\right\}$$

Thanks to this definition the two border integrals will vanish. For the second test function $\mathbf{q}$, nothing special appears, nothing more than integral over the domain, so a natural space is $L^2(\Omega)$. On the other hand, for the spaces of solutions, to apply the Lax-Milgram theorem we need the same space of functions. However in this case this is not possible. We have a boundary condition on the border for $\mathbf{u}$. So we consider the affine space $[H_g^1(\Omega)]^2 = \mathbf{H}_g^1(\Omega)$ defined as :

$$\mathbf{H}_g^1(\Omega) = \left\{\mathbf{u}\ :\ u\in L^2(\Omega), \nabla\mathbf{u}\in L^2(\Omega),\ \text{and}\ \mathbf{u}|_{\partial\Omega} = g\right\}$$

For the pressure, we notice that the solution is unique up to a constant term. To fix this we can impose that $\int_\Omega p = 0$. We choose $L^2_0(\Omega)$ as the space for pressure, and we impose this space also for q.

$$L^2_0(\Omega) = \left\{ p \in L^2(\Omega) \ : \ \int_\Omega p = 0 \right\}$$

The resulting problem reads : Find $(\mathbf{u}, p) \in \mathbf{H}^1_g(\Omega) \times L^2_0(\Omega)$ such that $\forall (\mathbf{v}, q) \in \mathbf{H}^1_0(\Omega) \times L^2_0(\Omega)$ the following holds :

$$\begin{aligned} \int_\Omega \nabla \mathbf{u} \colon \nabla \mathbf{v} - \int_\Omega p \operatorname{div} \mathbf{v} &= \int_\Omega \mathbf{f}\, \mathbf{v} \\ - \int_\Omega q \operatorname{div} \mathbf{u} &= 0 \end{aligned} \tag{2}$$

Mixed finite element element discrete formulation

We will write the inifite dimensional problem in a finite dimensional abstract form : Find $(\mathbf{u}_h, p) \in \mathbf{V}_h(\Omega) \times Q_h(\Omega)$ such that $\forall (\mathbf{v}_h, q) \in \mathbf{V}_{h,0}(\Omega) \times Q_{h,0}(\Omega)$ the following holds :

$$\begin{aligned} a(\mathbf{u}_h, \mathbf{v}_h) - b(\mathbf{v}_h, p_h) &= F(\mathbf{v}_h) \\ b(\mathbf{u}_h, q_h) &= 0 \end{aligned} \tag{3}$$

The discrete case has to verify the **descrete inf-sup** condition. A the common stable choice is the mini-element which consist of an enriched $\mathbb{P}_1$ finite element space with so-called bubble functions and $\mathbb{P}_1$ FEM space for the pressure.

A priori error estimate

The a priori error for a discritization scheme of the mixed formulation is given by :

$$\|\mathbf{u} - \mathbf{u}_h\|_{\mathbf{Q}_h} \leqq C_1(\alpha_h) \inf_{q_h \in \mathbf{Q}_h} \|p - q_h\| + C_2(\alpha_h, \beta_h) \inf_{\mathbf{v}_h \in \mathbf{V}_h} \|\mathbf{u} - \mathbf{v}_h\| \tag{4}$$

Where α_h (corsivity of the bilinear form a) and β_h (from inf-sup condition) have to be bounded from below to ensure uniform stability to a quasi optimal solution. For our mini-element these conditions are satisfied. Due to the $\mathbb{P}_1$ convergence of the velocity space the error can be bounded by :

$$\|\mathbf{u} - \mathbf{u}_h\|_{\mathbf{V}_h} \leqq Ch \tag{5}$$

Here C is a constant independent of h. Note as well that a similar formulation can be obtained for the pressure.

Question 3

We implemented the problem with the functional spaces given in the previous section. The complete is shown in the annex 1. To fix the constant part of the pressure (we don't have any pressure boundary conditions) we solve a slightly different problem :

Find $(\mathbf{u}_h^\epsilon, p) \in \mathbf{V}_h(\Omega) \times \mathbf{Q}_h(\Omega)$ such that $\forall(\mathbf{v}_h, q) \in \mathbf{V}_{h,0}(\Omega) \times \tilde{Q}_h(\Omega)$ the following holds :

$$
\begin{aligned}
a(\mathbf{u}_h^\epsilon, \mathbf{v}_h^\epsilon) - b(p_h^\epsilon, \mathbf{v}_h^\epsilon) &= F(\mathbf{v}) \\
b(p, \mathbf{v}) - c(p_h^\epsilon, q_h^\epsilon) &= 0
\end{aligned} \tag{6}
$$

These equations are called the perturbed Stokes equations. It is simple to verify that the zero mean pressure condition is verified. ... show that there exists a unique solution to this problem and that the error of the perturbed problem is not too far away from the actual problem we wanted to solve. This strategy is called the penalty method and helps to make the matrix formulation of the discrete problem symmetric, semi-definite, and sparse. This property is desirable from a computational point of view (for iterative solvers). For completeness, we will solve the perturbed problem even if this is not necessary for the following questions as we use a direct solver.

In our case, we know that pressures will diverge at the corners of the top boundary when we refine the mesh. Problems thus mostly occur close to the top and we will define the mesh to be finer close to this boundary. The space is divided into two rectangular regions. Specifically, the upper part will have a side length of 0.1 of the total square. In each region, the mesh will have a certain number of vertices per unit length (N1 for the regions with less interest and N2 for the other region). The resulting mesh and solutions are depicted below:

Question 4

To reach higher accuracy, we can also use two non-conforming meshes and connect them thanks to Lagrange multipliers. Due to this separation, we can ask for more precision where we need and release accuracy in the area without interest. This is the so-called Mortar method with dual Lagrange multipliers. We implemented this method and the code is as well given in the annex. In addition for the Lagrange multipliers, there has to be a certain mesh and corresponding FEM space. The mesh just consists of the interface between the two domains, but in FreeFEM++ this is not possible, so we took the mesh as the border of the upper mesh. Then we used $\mathbf{P}_0$ elements for each multiplier which ensure continuity on the normal and tangential components of the velocity. In contrast to a conforming mesh, we now need two finite element spaces on two different meshes defined independently. To deal with it we defined several variational

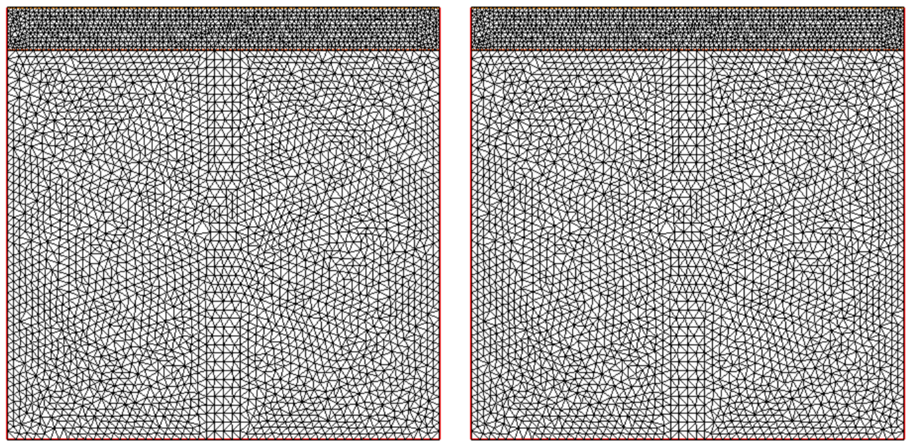

Figure 1: Conforming mesh and resulting solution using the Mini-element: N1=20 and N2=200

forms and extracted the associated matrix form to reconstruct our problem as a linear system.

To see the difference between the previous method and we tested it with different mesh sizes, the results can be seen below.

After a certain coefficient, the mesh is too dense to see the difference between them.

Comparing these results to the previous ones, we can see that the results are similar, but we could reach higher accuracy around the interest regions. The pressure is still diverging as expected when refining the mesh.

Comparison between conforming and non-conforming meshs

Question 5

Annex

Below the code for solving the Stokes problem on a conforming and non-conforming are shown.

Stokes on conforming mesh

```
// parameters
real eps = 1e-6; // for penalty method

real frac = 0.1; // fraction of side length in upper area
int n2 = 300; // mesh size in upper area
```

```
int n1 = n2 * frac; // mesh size in lower area

// mesh
border a1(t=0, 1){x=t; y=0; label=1;};
border b1(t=0, 1-frac){x=1; y=t; label=1;};
border c12(t=0, 1){x=1-t; y=1-frac; label=2;}; // border between areas
border d1(t=frac, 1){x=0; y=1-t; label=1;};
border a2(t=0, frac){x=1; y=1-frac+t; label=1;};
border b2(t=0, 1){x=1-t; y=1; label=3;};
border c2(t=0, frac){x=0; y=1-t; label=1;};

mesh Th = buildmesh(a1(n1)+b1((1-frac)*n1)+c12(n1)+d1((1-frac)*n1)+a2(frac*n2)+b2(n2)+c2(frac*n2));

plot(Th, wait=1);

// fespaces
fespace Xh(Th, [P1b, P1b, P1]);
Xh [ux, uy, p], [vx, vy, q];

// problem
solve stokes ([ux, uy, p], [vx, vy, q])
  =   int2d(Th)(dx(ux)*dx(vx) + dy(ux)*dy(vx) + dx(uy)*dx(vy) + dy(uy)*dy(vy))
      - int2d(Th)(p*(dx(vx) + dy(vy)))

        - int2d(Th)(q*(dx(ux) + dy(uy)))
      - int2d(Th)(eps*p*q)

      + on(3, ux=1, uy=0)
        + on(1, ux=0, uy=0);

// plot
plot([ux, uy], p, value=true, wait=true);
```

Stokes on non-conforming mesh

```
// parameters
real eps = 1e-6; // for penalty method

real frac = 0.1; // fraction of side length of upper area
int n2 = 300; // mesh size in upper area
int n1 = n2 * frac; // mesh size in lower area

// lower mesh
border a1(t=0, 1){x=t; y=0; label=1;};
border b1(t=0, 1-frac){x=1; y=t; label=1;};
border c1(t=0, 1){x=1-t; y=1-frac; label=2;};
border d1(t=frac, 1){x=0; y=1-t; label=1;};

mesh Th1 = buildmesh(a1(n1)+b1((1-frac)*n1)+c1(n1)+d1((1-frac)*n1));

// upper mesh
border a2(t=0, frac){x=1;y=1-frac+t; label=1;};
```

```
border b2(t=0, 1){x=1-t; y=1; label=3;};
border c2(t=0, frac){x=0; y=1-t; label=1;};
border d2(t=0, 1){x=t; y=1-frac; label=2;};

mesh Th2 = buildmesh(a2(frac*n2)+b2(n2)+c2(frac*n2)+d2(n2));

plot(Th1, Th2, wait=1);

// interface mesh
mesh interface = emptymesh(Th2);

// lower fespaces and problem
fespace Xh1(Th1, [P1b, P1b, P1]);
Xh1[ux1, uy1, p1];

varf stokes1([ux, uy, p], [vx, vy, q], solver=sparsesolver)
  =   int2d(Th1)(dx(ux)*dx(vx) + dy(ux)*dy(vx) + dx(uy)*dx(vy) + dy(uy)*dy(vy))
      - int2d(Th1)(p*(dx(vx) + dy(vy)))

      - int2d(Th1)(q*(dx(ux) + dy(uy)))
      - int2d(Th1)(eps*p*q)

      + on(1, ux=0, uy=0);

// upper fespaces and problem
fespace Xh2(Th2, [P1b, P1b, P1]);
Xh2[ux2, uy2, p2];

varf stokes2([ux, uy, p], [vx, vy, q], solver=sparsesolver)
  =   int2d(Th2)(dx(ux)*dx(vx) + dy(ux)*dy(vx) + dx(uy)*dx(vy) + dy(uy)*dy(vy))
      - int2d(Th2)(p*(dx(vx) + dy(vy)))

      - int2d(Th2)(q*(dx(ux) + dy(uy)))
      - int2d(Th2)(eps*p*q)

      + on(3, ux=1, uy=0)
      + on(1, ux=0, uy=0);

// interface fespace
fespace Ih(interface, [P0, P0]);
Ih [lambda, mu];

// lagrange multipliers
varf lagrange1([lambda, mu], [ux, uy, p], solver=sparsesolver) = int1d(Th2, 2)(uy*lambda-ux*mu);
varf lagrange2([lambda, mu], [ux, uy, p], solver=sparsesolver) = int1d(Th2, 2)(-uy*lambda+ux*mu);

// solving
matrix A1 = stokes1(Xh1, Xh1);
real[int] B1 = stokes1(0, Xh1);
matrix A2 = stokes2(Xh2, Xh2);
real[int] B2 = stokes2(0, Xh2);

matrix L1 = lagrange1(Ih, Xh1);
```

```
matrix L2 = lagrange2(Ih, Xh2);
real[int] B3(Ih.ndof); B3 = 0;

matrix A = [[A1, 0,L1],
        [0, A2, L2],
        [L1',L2',0]];
set(A, solver=sparsesolver);

real[int] B = [B1, B2, B3];
real[int] S1(Xh1.ndof), S2(Xh2.ndof), multipliers(Ih.ndof);
real[int] sol = A^-1 * B;
[S1, S2, multipliers] = sol;

p1[] = S1;
p2[] = S2;
lambda[] = multipliers;

// plot
plot([ux1, uy1], p1,[ux2, uy2], p2,value=true, wait=true);
```

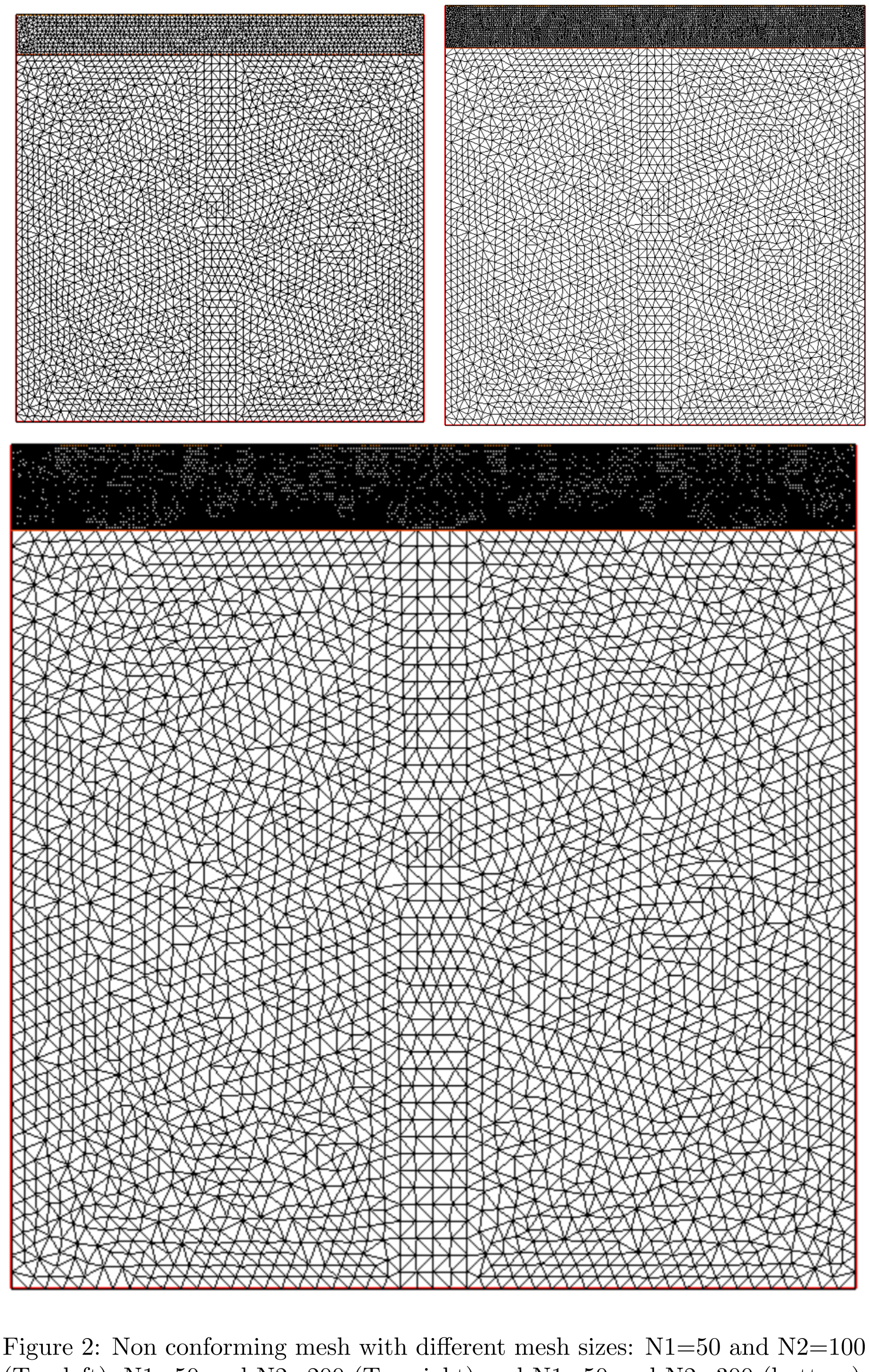

Figure 2: Non conforming mesh with different mesh sizes: N1=50 and N2=100 (Top left), N1=50 and N2=200 (Top right) and N1=50 and N2=300 (bottom)

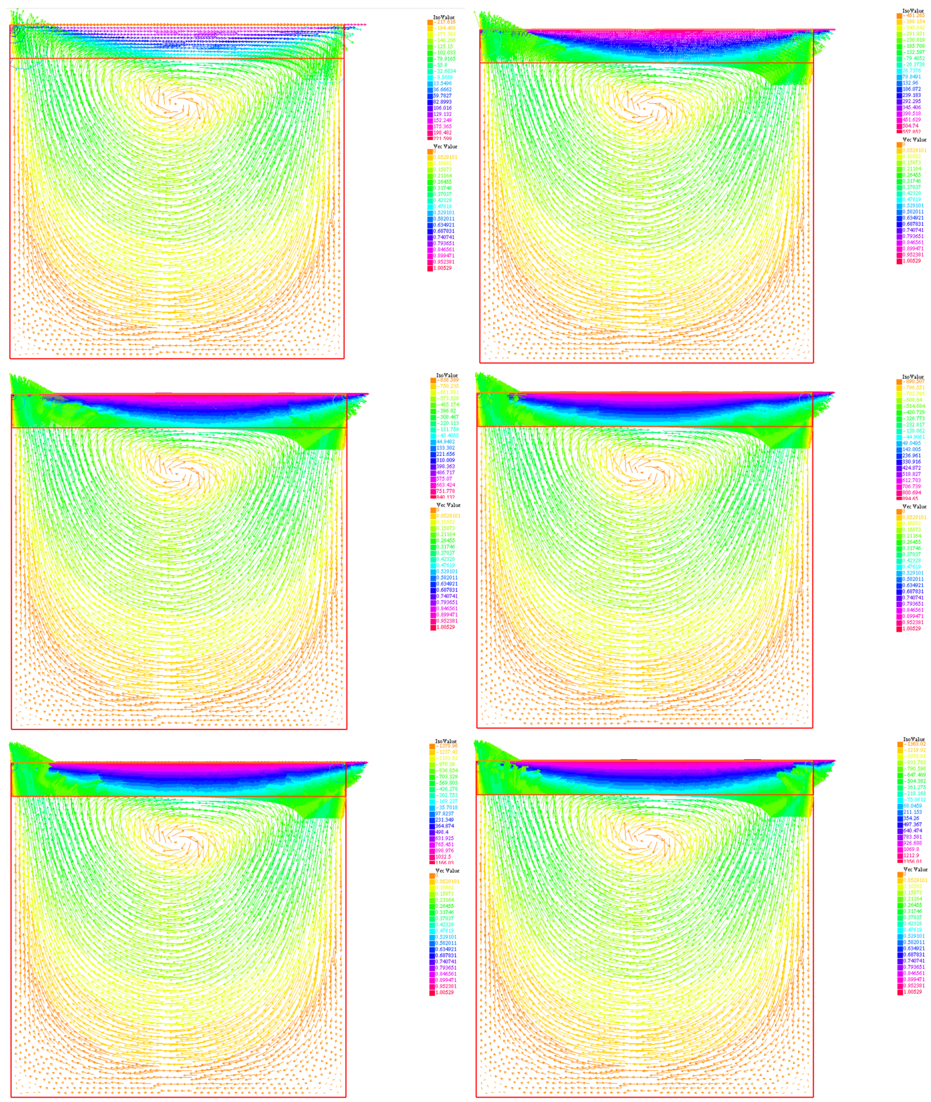

Figure 3: Solutions for different different mesh sizes: N1=50 and N2=100 (Top left), N1=50 and N2=200 (Top right), N1=50 and N2=300 (middle left), N1=50 and N2=400 (middle right), N1=50 and N2=500 (bottom left), N1=50 and N2=600 (bottom right)

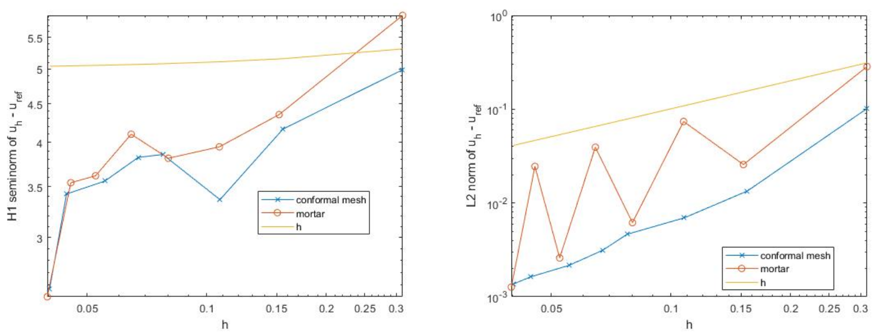

Figure 4: H^1 seminorm and L^2 norm of velocity versus a reference solution, for conformal and non-conformal meshs