DTU Compute
Department of Applied Mathematics and Computer Science

DTU

The Incompressible Navier-Stokes equation using IPCS scheme on several domains

Federico Crespo (s200161), Michail Anevlavis (s190015), Michele Rocca (s192632)
Technical University of Denmark (DTU)

Motivation

The Finite Element Method (FEM) is a numerical framework for solving Differential Equations that is widely used when a great flexibility on the shape of the domain is needed. Incompressible Navier-Stokes equations (INSE) are a set of partial differential equations which describe the motion of viscous fluid substances by assuming that the volume of fluid elements is constant.We solve the equations by using the python library FEniCS [2] for obtaining the velocity profile and the pressure in several domains by changing the shape of the walls of a pipe and inserting different type of obstacles for the fluid. In this project the INSE are solved using the Incremental Pressure Correction Scheme (IPCS)[1]. The results shows the flexibility of the numerical framework in cases that imitate some real-world problems such as fusiform aneurysm and stenosis in arteries, the double slit experiment, fluid resistance on an Airfoil.

Model equations

The Incompressible Navier-Stokes equation

$$\begin{cases} \rho\left(\frac{\partial u}{\partial t} + u\cdot\nabla u\right) = \nabla\cdot\sigma(u,p) + f \\ \nabla\cdot u = 0 \end{cases} \quad (1)$$

where σ is the Cauchy stress tensor which for a Newtonian fluid is defined as $\sigma(u,p) = 2\mu\varepsilon(u) - pI$

The first equation comes (we can see that is similar to Newton's second law) from the conservation of momentum and the second one comes from the conservation of mass and the incompressibility of the fluid.

Here, $u(x,y)$ is the unknown velocity vector, $p(x,y)$ is the unknown pressure as the scalar field, ρ is the fluid density, μ is the dynamic viscosity, f is the body force per unit mass, and $\varepsilon(u) = (\nabla u + (\nabla u)^T)/2$ is the symmetric gradient.

Variational Formulation

The weak formulation for the homogeneous problem is

$$\rho\int_\Omega \left(\frac{\partial u}{\partial t} + u\cdot\nabla u\right) v\, d\Omega - \int_\Omega (\nabla\cdot\sigma(u,p))\, v\, d\Omega = 0$$

the second term can be further expanded integrating by parts

$$-\int_\Omega (\nabla\cdot\sigma(u,p))\, v\, d\Omega = \int_\Omega (\sigma(u,p))\, \nabla v\, d\Omega - \int_{\partial\Omega} \sigma(u,p)\hat{n}\, v\, d\partial\Omega$$

where $\sigma(u,p)\hat{n}$ is the Boundary traction.

We can assume that the derivative in the direction of the channel is zero at the outflow

$$\sigma(u,p)\hat{n} = \mu(\nabla u\hat{n} + (\nabla u)^T\hat{n}) - pI\hat{n} = \mu\nabla u\hat{n} - pI\hat{n}$$

Hence using the short hand notation

$$\rho\,\langle \partial u/\partial t + u\cdot\nabla u, v\rangle + \langle\sigma(u,v),\varepsilon(v)\rangle + \langle p\hat{n} - \mu\nabla u\hat{n}, v\rangle_{\partial\Omega} = 0 \quad (2)$$

Splitting Strategy

The system has a so-called saddle point structure and requires special techniques to be solved efficiently. The idea We want to consider the two equations in (1) separately. The IPCS is a modification of the Chorin's method based on the (Helmholtz) decomposition of any vector field into a solenoidal part and an irrotational part. The IPCS [1] consists of three steps

- Compute a tentative velocity u^* using the old-step pressure

$$\rho\,\langle (u^* - u^n)/\Delta t + u^n\cdot\nabla u^n, v\rangle + \left\langle \sigma(u^{n+1/2}, v), \varepsilon(v)\right\rangle + \left\langle p^n\hat{n} - \mu\nabla u^{n+1/2}\hat{n}, v\right\rangle_{\partial\Omega} = 0 \quad (3)$$

- Use u^* for computing the new pressure

$$\langle\nabla p^{n+1}, \nabla q\rangle = \langle\nabla p^n, \nabla q\rangle - \langle\nabla\cdot u^*, q\rangle/\Delta t \quad (4)$$

- Requiring that $\nabla\cdot u^{n+1} = 0$ we end up with the

$$\langle u^{n+1}, v\rangle = \langle u^*, v\rangle - \left\langle\nabla(p^{n+1} - p^n), v\right\rangle/\Delta t \quad (5)$$

Implementation and Performance

FEniCS is a open-source python library that allows us to solve PDE's numerically using FEM. A domain can be generating by adding or substracting simple geometrical domains.It can produce automatically a triangular mesh for any domain entered (with the use of 'generate_mesh()').

The implementation performed by our group consists on defining the needed boundaries, boundary conditions and sizes of the mesh in order for everything to be computed without any major numerical errors (divergences).

Simulation	Δt	mesh res.	cpu time	$CO_2[g]$ cons.
One Cylinder	10^{-3}	4585	1557.5 s	3.375
Two Cylinders	10^{-3}	2436	499.710 s	1.083
Double Slit	$5{\cdot}10^{-5}$	5204	2345.77 s	5.0825
Aneurysm	10^{-3}	4225	886.29 s	1.92
Stenosis 1 ob.	$2{\cdot}10^{-4}$	1453	1368.994 s	2.967
Stenosis 2 ob.	$12{\cdot}10^{-5}$	1338	1298.181 s	2.814
Airfoil ob.	10^{-4}	3510	3492.11 s	7.566

Conclusions

We can clearly see turbulence (due to high Reynolds numbers) in almost all figures, since the parameters have been chosen in order to show so. The turbulence is being shown without introducing high enough errors in the algorithm (even with its chaotic nature).

For most of the simulations in domains with sharp corners a smaller time-step was needed for avoiding numerical divergence.

Finally, the INSE can be applied to various solid geometries and it can be used to solve real life problems in various fields of science.

Bibliography

- KATUHIKO GODA. "A Multistep Technique with implicit Difference Schemes for Calculating Two- or Three-Dimensional Cavity Flows". In: (Journal of Computational Physics 1979).
- Anders Logg Hans Petter Langtangen. "Solving PDEs in Python The FEniCS Tutorial I". In: (SpringerOpen 2016).

Numerical Results

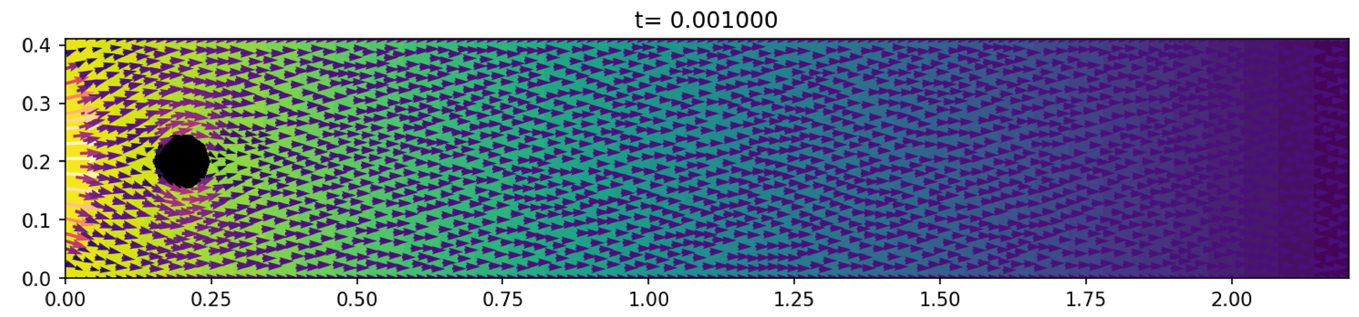

Figure 1: Flow through a channel with a cylinder as an obstacle

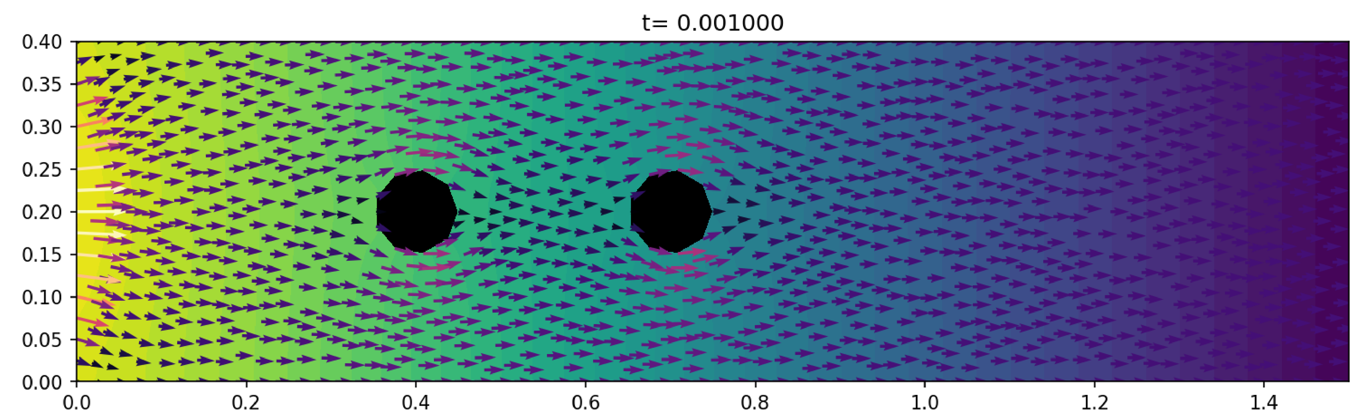

Figure 2: Flow through a channel with two cylinders as obstacles

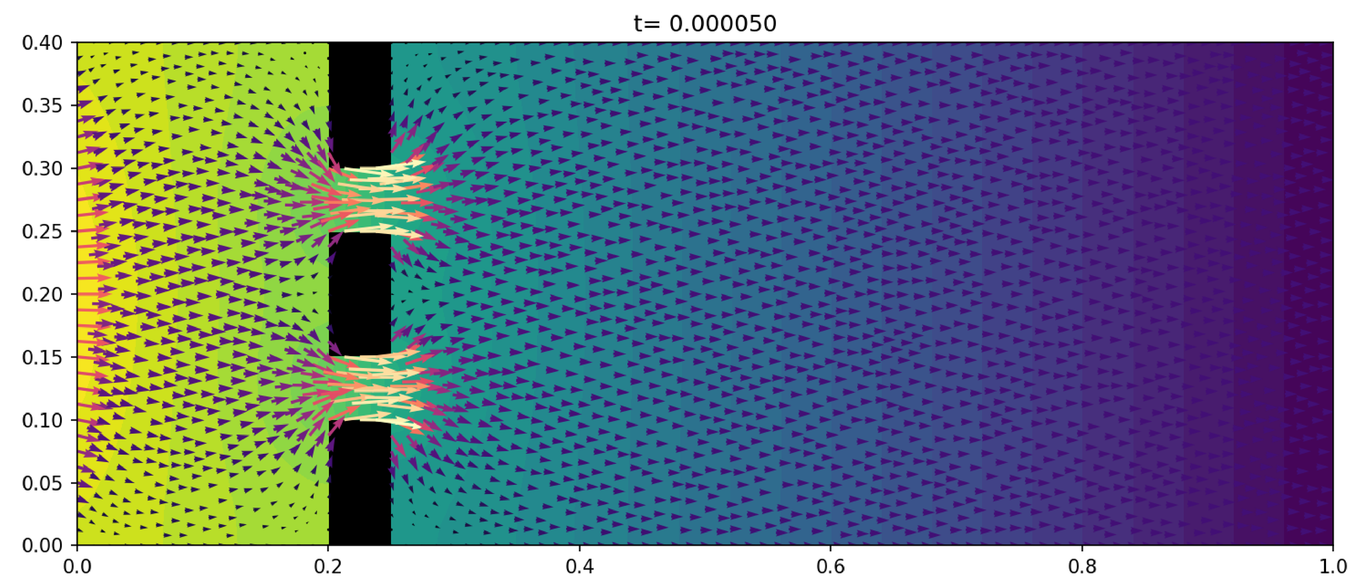

Figure 3: Flow through a channel with a double slit as an obstacle

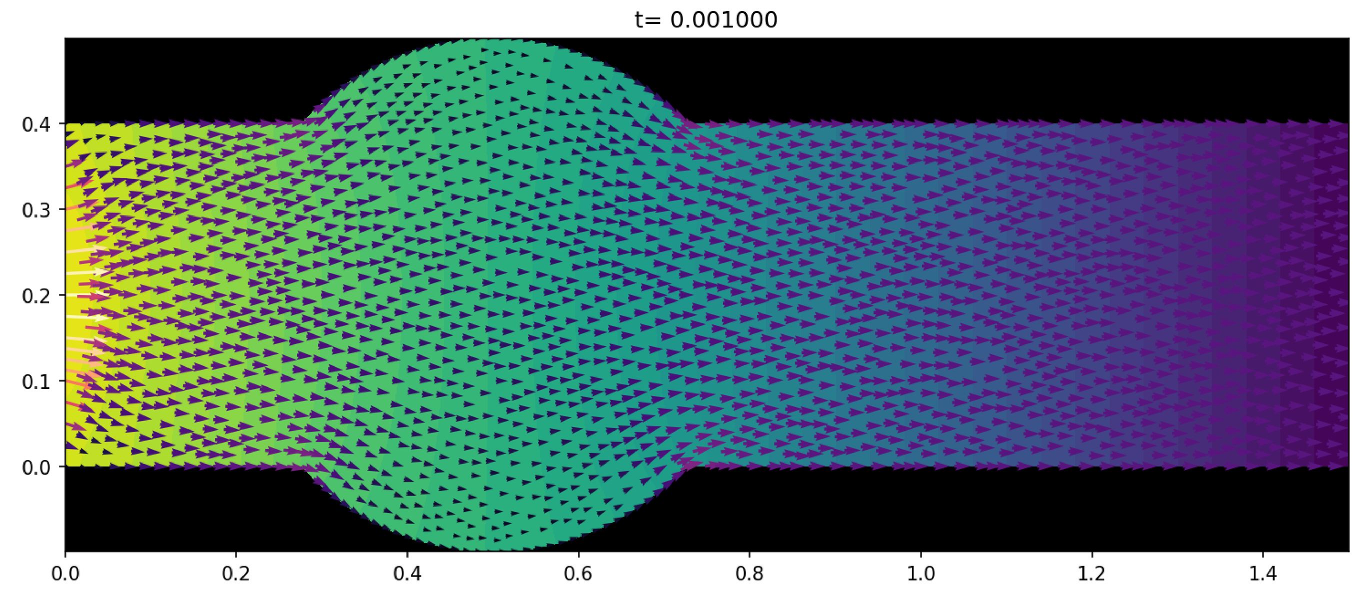

Figure 4: Flow through a channel simulating an Arterial Aneurysm

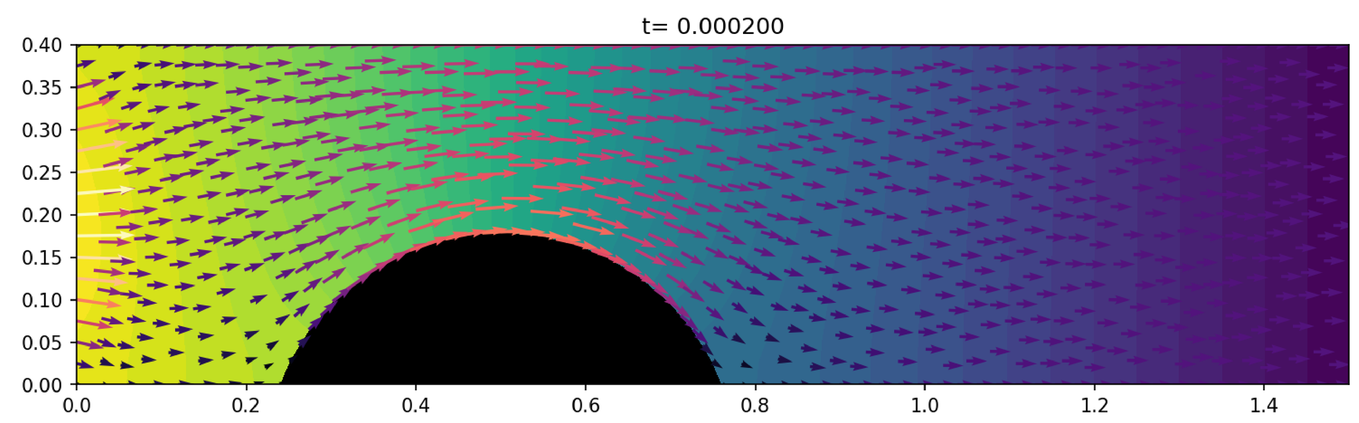

Figure 5: Flow through a channel simulating an Arterial Stenosis with one obstacle

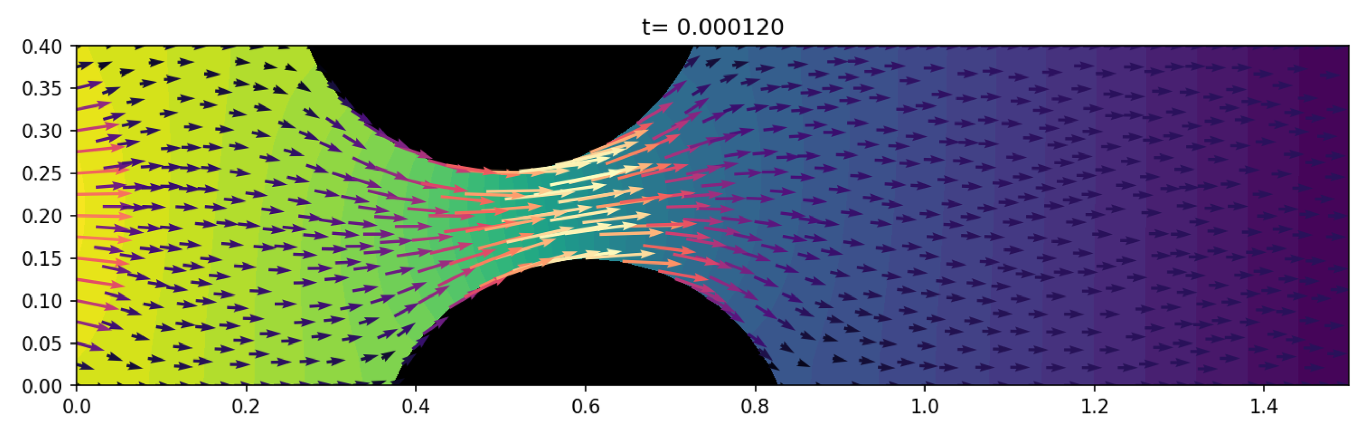

Figure 6: Flow through a channel simulating an Arterial Stenosis with two obstacles

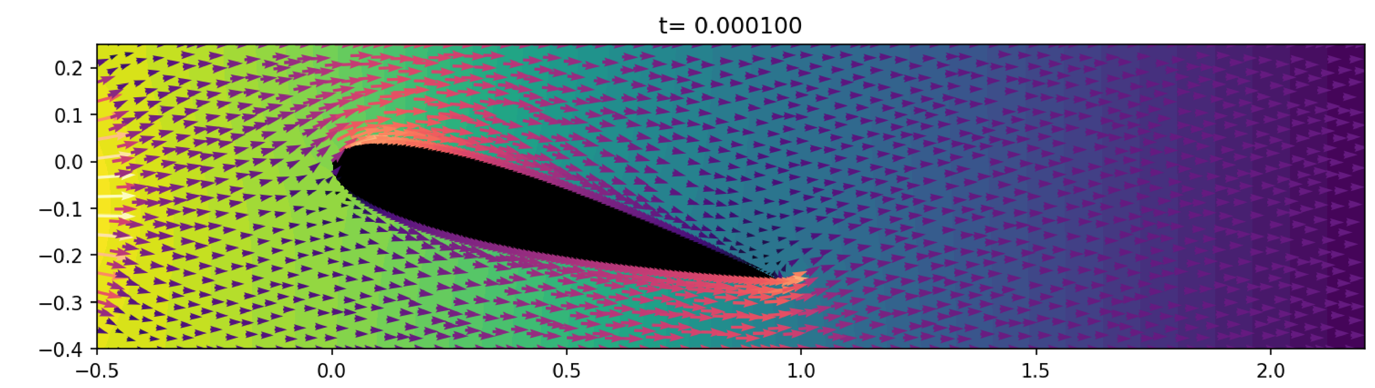

Figure 7: Flow through a channel with a NACA0018 airfoil at a 15º angle (takeoff Boeing 314) as an obstacle