

Assignment 3 - Turbulent flow around a 3D cylinder

In this assignment we will study **usikker paa rett beskrivelse ...**

Generating the mesh

In the example available from **usikker paa hvordan du vil ordne dette** you will find a highly refined cylinder mesh. Running simulations on this mesh will take extremely long, but complex geometries often demand high mesh resolutions to generate reasonable results. However, since we know the exact geometry, we can make controlled compromises to the mesh.

For the case of a cylinder, we will need the most resolution around the cylinder itself as well as in the region behind it where the turbulent flow will dominate. Using graded mesh refinement we can accomplish this by making the cells closer to the center smaller than the ones on the boundaries. The mesh blocks are placed in such a way that they make resemble a scandinavian flag. Run **blockmesh** on the provided case and visualize the mesh in paraFOAM using the *wireframe* display option on a slice through the circular cylinder plane.

Open the **blockMeshDict** and familiarize with the different points making up the cylinder and the cavity. Perform the following changes:

- Reduce the amount of cells in the z -direction to 39.
- Reduce the amount of cells in the x -direction of the center blocks above and below the cylinder to 12.
- Reduce the amount of cells in the y -direction of the middle-lying blocks to the left and right of the cylinder (inlet/outlet) to 12.

Once this is done you should instead of a largest-to-smallest cell ratio of 20, reduce it to 10 overall. Visualize the new mesh and comment on the changes.

Running the simulations

Use the Smagorinsky and oneEqEddy models with a timestep of 0.0025 to generate the initial data. A Python-script `simulationQueueer.py` for queuing processes is provided on the **usikker paa hvordan du vil ordne dette**. Open it in a text editor and you will find instructions on how to use it.

Use the `homogeneousDynSmagorinsky` and `dynLagrangian` models (remember that these models have their own necessities, see earlier assignments). Comment the convergence using these models.

Changing the cylinder radius

Re-enter the `blockMeshDict` file and double the radius of the cylinder. You will need double the space in the center and middle blocks, as well as new arcs in the *edges dictionary*. Arcs are given three points, and will interpolate a line through them. The first two points, p_1 and p_2 , are given by their corresponding number on the mesh. These are the start- and endpoints. The last point, p_3 , is given as a vector at the end. In our case, for a perfect circle, you will need to make sure this relation is kept

$$|p_3| = \sqrt{|p_1|^2 + |p_2|^2} \equiv R.$$

Performs the conversion and visualize the new mesh before running new simulations for the previously given cases. Comment on the results and compare them to the previous case.