22.315 - Homework 1

Name: Lorenzo Mazzocco **Due Date:** February 23 2023
HW Partner: Witiwat Jiragoontansiri **Code:** StarCCM+

Contents

1 GEOMETRY **2**

2 MESHING **3**

3 PHYSICS AND MODEL SETUP **4**

4 SOLVER AND CONVERGENCE **5**

5 POSTPROCESSING AND DNS BENCHMARK **6**

5.1 Model Consistency 6

5.2 Bulk Velocity U^+ 6

5.3 Turbolent Kinetic Energy k^+ 7

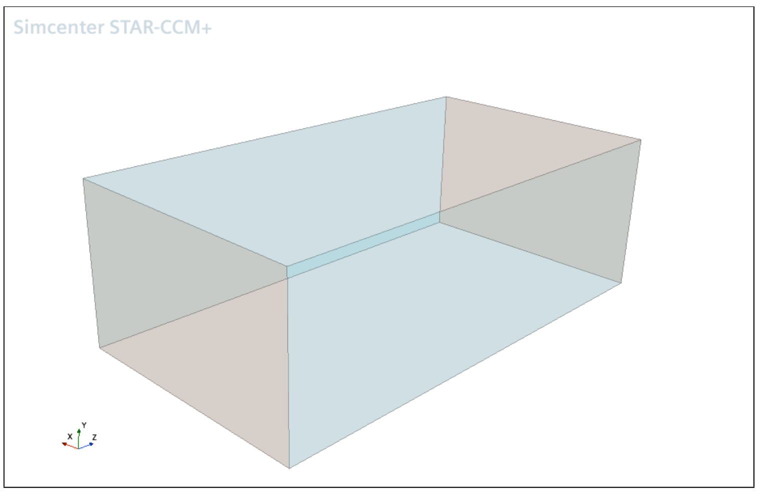
6 CHALLENGES AND CODE **9**

1. GEOMETRY

The geometry is a cuboid constructed from the origin of the coordinate system. It extends in the following directions:

- x: 3.2m so $x \in [0, 3.2]$
- y: 2.0m so $y \in [0, 2.0]$
- z: 6.0m so $z \in [0, 6.0]$

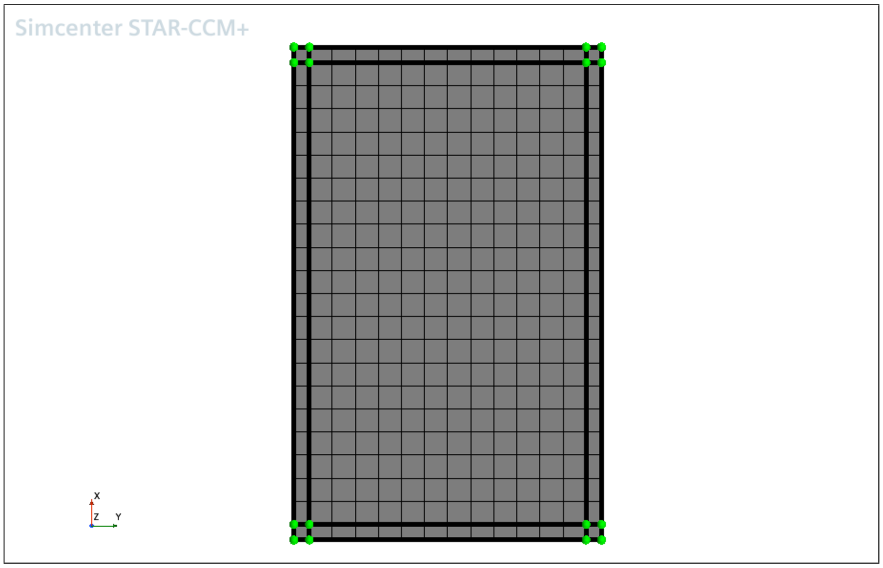
The following domain is obtained:



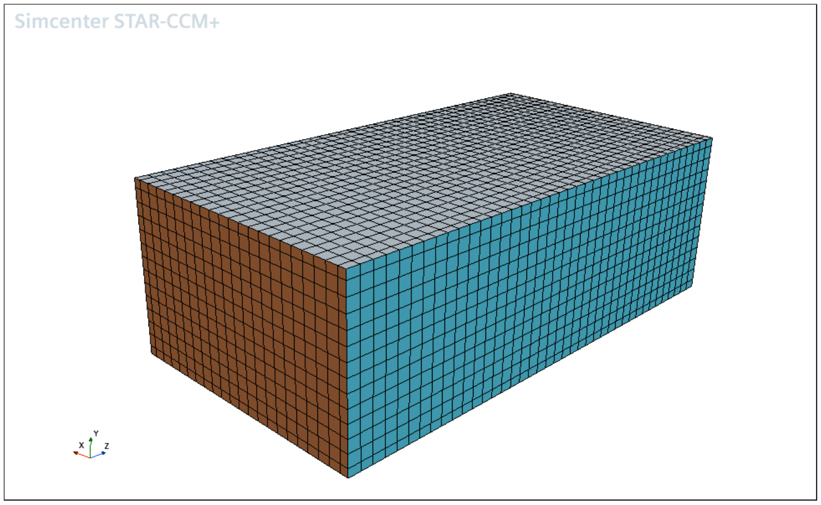
For the x-normal faces (left and right) a "Symmetric Plane" boundary condition was selected. For the y-normal faces (top and bottom) a "Wall" boundary condition was selected with no-slip condition by default. For the z-normal faces (inlet and outlet) a "Pressure Outlet" boundary condition was selected with absolute pressure for both surfaces set at 0 Pa. Then on the inlet and outlet faces a periodic interface was created of the type "Fully-Developed Interface Boundary" with a pressure jump of $\Delta p = 2.535 MPa$.

2. MESHING

A **Directed Mesh** was used with source surface the inlet and target surface the outlet. Patches of 15 cm were used and 1 layer of 10 cm next to top, bottom, left and right surfaces:



Then a Volume Distribution with 40 layers (corresponding to a size of 0.15 cm) was specified and the 2D mesh finally extended in 3D:

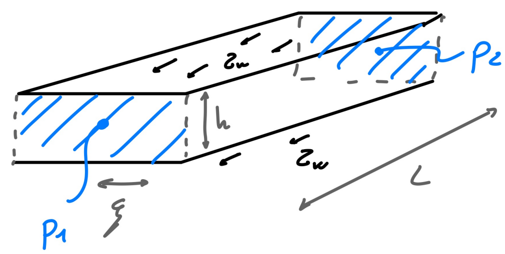


3. PHYSICS AND MODEL SETUP

For the sake of simplicity the following constant material parameters were chosen:

- **Density:** $\rho = 1$ kg/m^3
- **Dynamic Viscosity:** $\mu = 1\ Pa \cdot s$

The only pressure drops in the system are frictional (given by the top and bottom walls). A simple free body diagram can give us the relationship between the pressure drop Δp and the wall shear stress τ_w:



Balancing the forces in the z direction:

$$\Delta p \cdot \xi \cdot h - 2 \cdot \tau_w \cdot L \cdot \xi = 0 \tag{1}$$

$$\Delta p \cdot h - 2 \cdot \tau_w \cdot L = 0 \tag{2}$$

$$\tau_w = \frac{\Delta p \cdot h}{2L} \tag{3}$$

Finally we can compute the pressure drop given Re_τ:

$$Re_\tau = \frac{\delta \cdot u_\tau}{\nu} \longrightarrow u_\tau = \frac{Re_\tau \cdot \nu}{\delta} \longrightarrow \frac{\tau_w}{\rho} = \frac{Re_\tau^2}{\delta^2} \cdot \frac{\mu^2}{\rho^2} \longrightarrow \tau_w = \frac{Re_\tau^2}{\delta^2} \cdot \frac{\mu^2}{\rho} \tag{4}$$

Substituting τ_w as a function of Δp:

$$\frac{\Delta p \cdot h}{2L} = \frac{Re_\tau^2}{\delta^2} \cdot \frac{\mu^2}{\rho} \longrightarrow \Delta p = \frac{Re_\tau^2}{\delta^2} \cdot \frac{2L\mu^2}{\rho h} \tag{5}$$

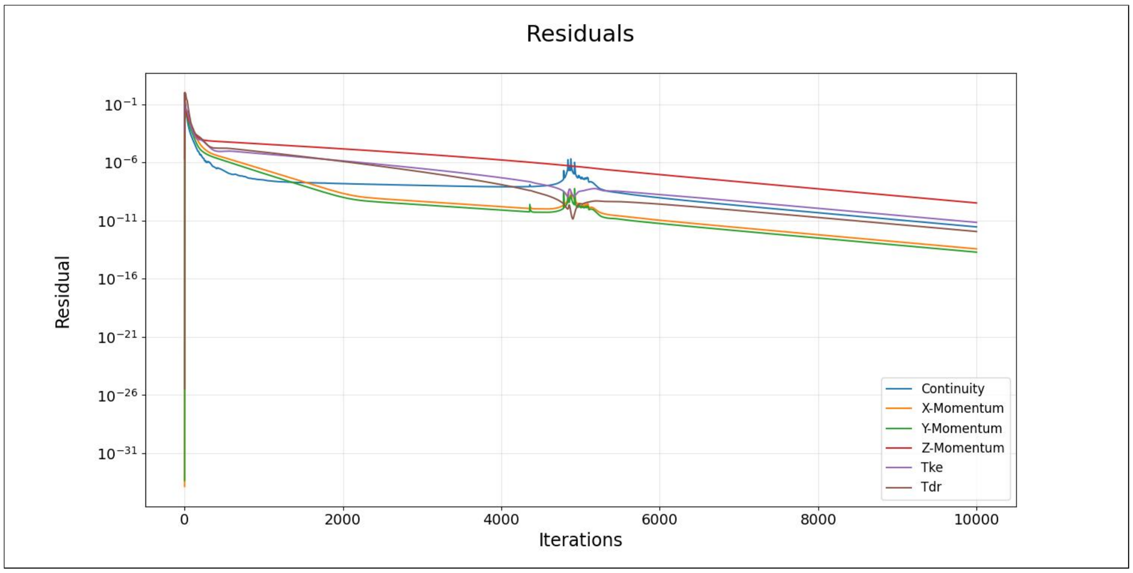
Knowing that by setup $\delta = \mu = \rho = 1$, $Re_\tau = 650$ and the geometric data one finds $\Delta p = 2.535$ MPa. This is the pressure difference value we will impose to the periodic interface.

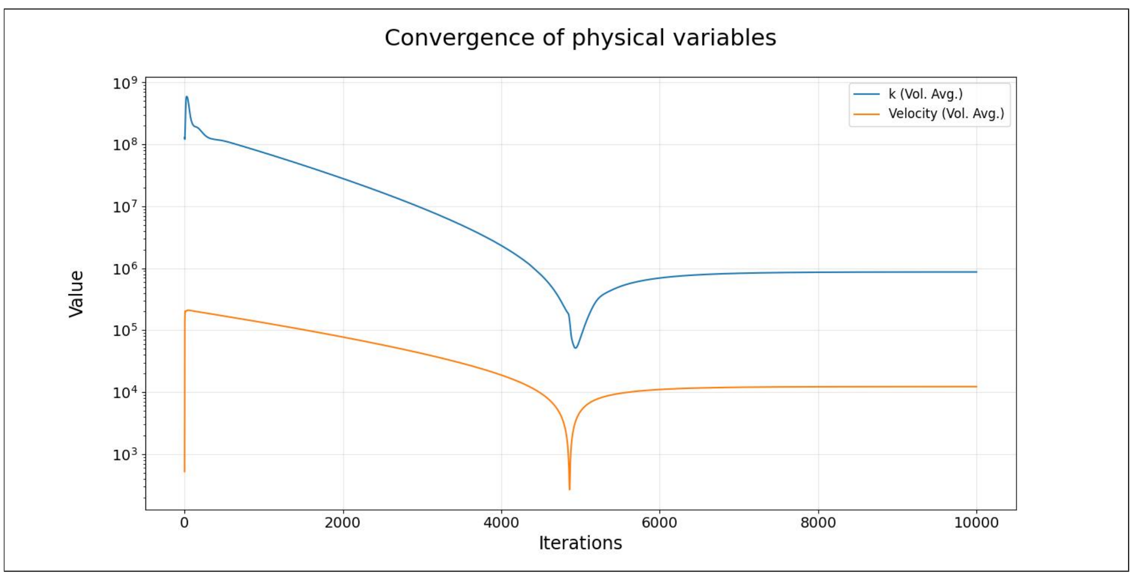
4. SOLVER AND CONVERGENCE

A steady state $k-\varepsilon$ solver was selected. The variables are all inizialized to zero. To check convergence on top of the residuals we monitor two physical quantities (volume averaged over the whole domain):

- velocity magnitude
- turbolent kinetic energy

After 10.000 iterations we stop the solver:



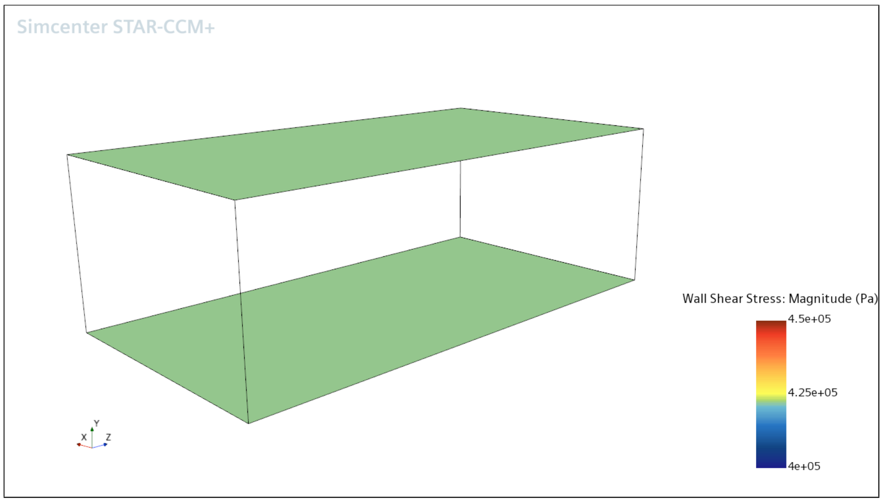


5. POSTPROCESSING AND DNS BENCHMARK

Data for the DNS benchmark is taken from the data provided in this file. The data from the simulation undergoes a first elaboration within StarCCM+ (user-defined field functions are plotted) and then data exported from plots is post-processed in a Jupyter Notebook.

5.1. Model Consistency

First of all I verified that the simulation results are coherent with the physical model imposed. I did so by extracting the wall shear stress. Using a simple scalar scene on the wall boundaries I obtained the following:



The distribution is for all practical purposes uniform (confirming again convergence). The values range from 422232 Pa to 422235 Pa.

From eq. (3) we know that the shear stress as a function of the pressure drop across the channel is:

$$\tau_w = \frac{\Delta p h}{2L} = 422500 Pa \tag{6}$$

We can attribute the discrepancy of about 270 Pa to numerical errors and accept that model consistency is verified.

5.2. Bulk Velocity U^+

Data for the DNS benchmark is taken from this report in section 1.5 (Flow conditions). We can compute the normalized bulk velocity using the values for Re_m and Re_τ provided for Case 5:

$$U^+_{DNS} = \frac{U_m}{u_\tau} = \frac{Re_m}{2Re_\tau} = 18.8877 \tag{7}$$

In StarCCM we can extract the bulk mean velocity creating an ad hoc scalar field function for the streamwise component of the velocity vector field and dividing it by u_τ. Then a simple report on the volume average of said function yields:

$$U^+_{RANS} = 18.83627 \tag{8}$$

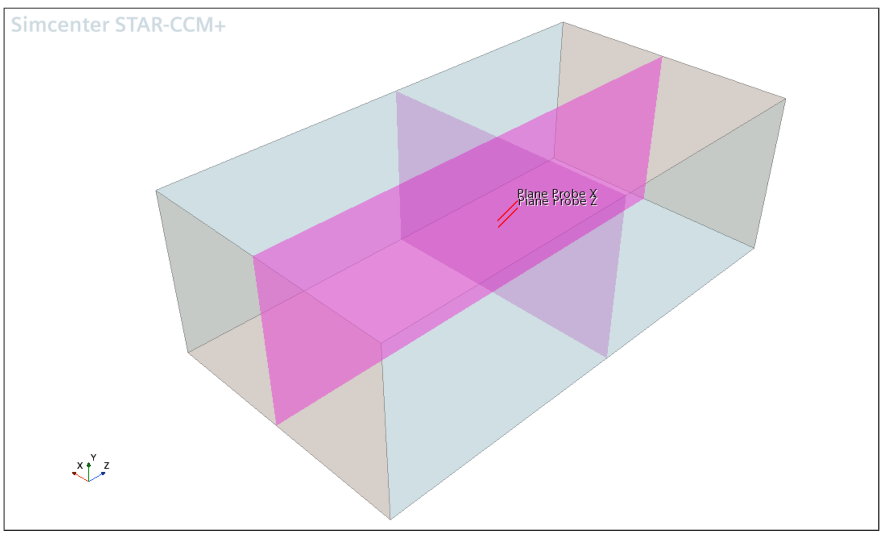
This corresponds to an error of 0.27%.

5.3. Turbolent Kinetic Energy k^+

First of all I created two user-defined scalar field functions according to their definition: $y+$ and $k+$.

To extract the data from the cells I used the combination of two planes:

- Plane X: defined by $z = 3$, spans in the x and y directions.
- Plane Z: defined by $x = 1.6$, spans in the z and y directions.

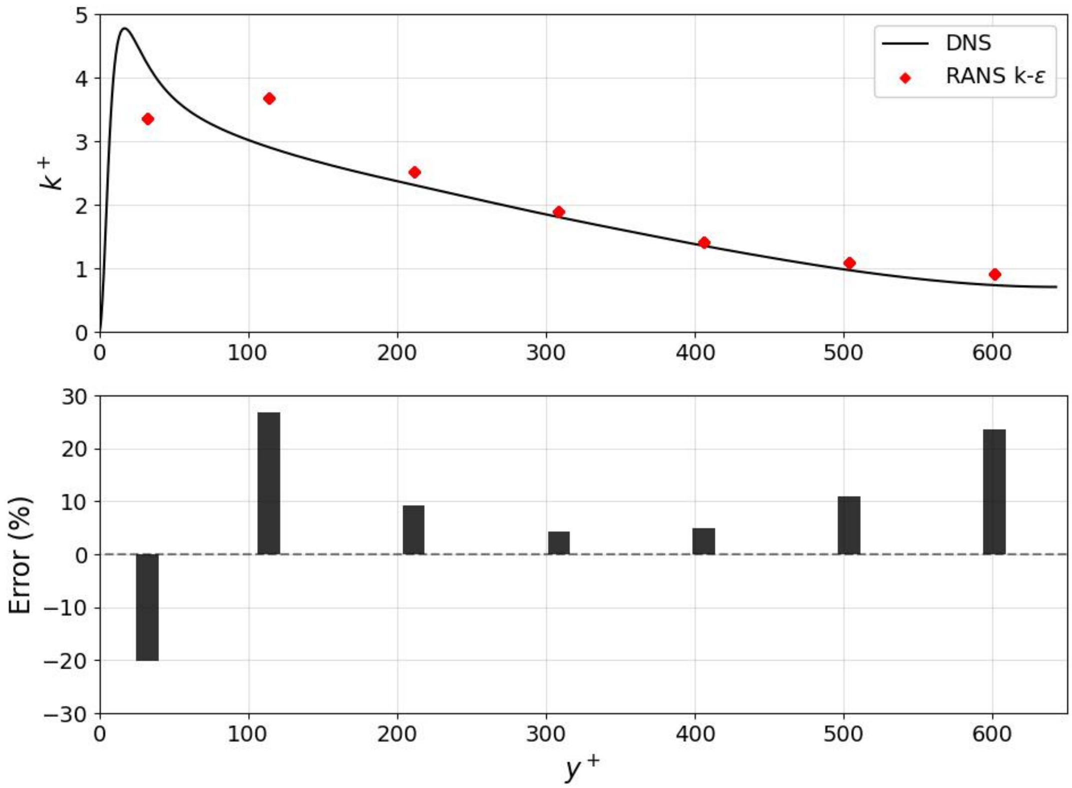


Using this method instead of a simple Line Probe (which would extract only 7 values) helped me verify the convergence of the solution which shouldn't show a dependence on the x or z positions. Indeed scattering all the points in a cartesian plot shows no dependence on those variables (we see 7 clusters of multiple, perfectly overlapping points).

To get the DNS turbolent kinetic energy I extracted the RMS values for all the velocity fluctuations (u,v,w). The RMS values can be found in the Turbolent Statistics - Rms section (4.1.2) under columns u_rms, v_rms, w_rms. To get the normalized turbolent kinetic energy one can just sum up the squares of the RMSs:

$$k^+ = \frac{1}{2}\left[u^2_{RMS} + v^2_{RMS} + w^2_{RMS}\right] \tag{9}$$

Plotting our results vs the DNS ones I obtained the following:



6. CHALLENGES AND CODE

I only encountered two challenges while doing this PSET:

1. STARCCM+ would select a more sophisticated model than the wanted standard RANS $k - \varepsilon$, then finding the model settings in the main menu to verify that it was indeed the one I intended was not trivial to a new user like me
2. Initially it was challenging to understand the nomenclature and convention in the DNS data report

All the code can be found in this public GitHub Repository.