

Comparison of FEM with FVM applied to multiphysics cases - I

Introduction to OpenFOAM Finite Volume Toolbox

Raffaele Ponzini, CINECA

07/21

Univerza v Ljubljani



TECHNISCHE
UNIVERSITÄT
WIEN



VSB TECHNICAL
UNIVERSITY
OF OSTRAVA

IT4INNOVATIONS
NATIONAL SUPERCOMPUTING
CENTER



Co-funded by the
Erasmus+ Programme
of the European Union

This project has been funded with support from the European Commission.

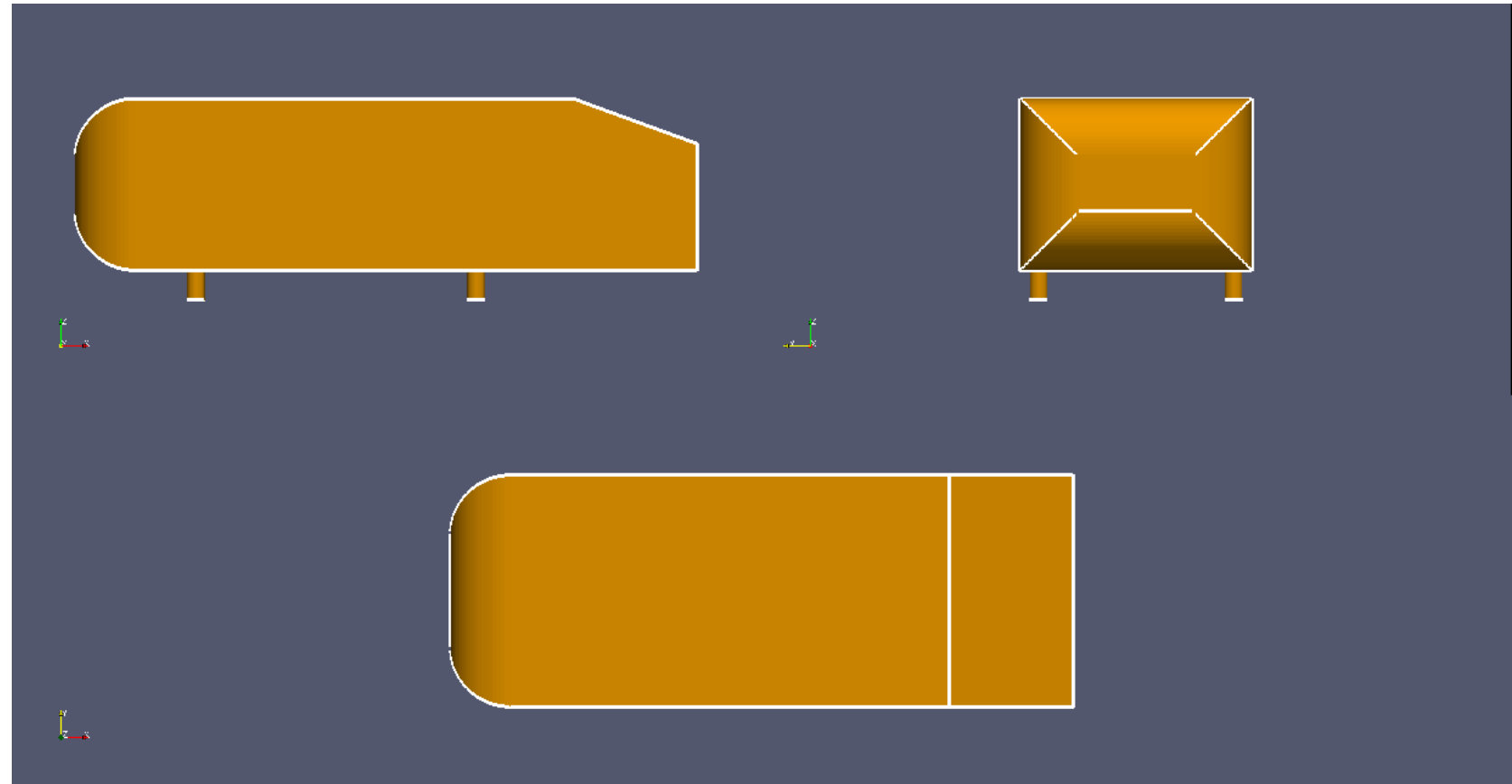
This publication [communication] reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Guided examples: 'hacking' the motorbike tutorial

- Problem description
- Running the tutorial for a different geometry
- Solution monitoring during calculation
- Analyzing main results
- Using a cluster
- Closing remarks

- The motorbike tutorial is widely used example to show the ability of OpenFOAM to solve complex 3D real life industrial problems
- The motorbike tutorial refers to a virtual wind tunnel over a 3D model of a motorbike with driver
- External aerodynamics virtual wind tunnel is one of the most studied kind of application in CFD since ever
- As discussed in the first part we can work starting from a given tutorial and 'hack it' to study our own problem

- The definition of the geometry is given in S.R. Ahmed, G. Ramm, Some Salient Features of the Time-Averaged Ground Vehicle Wake, SAE-Paper 840300, 1984
- The usual name of this cad is Ahmed body as for the first author name



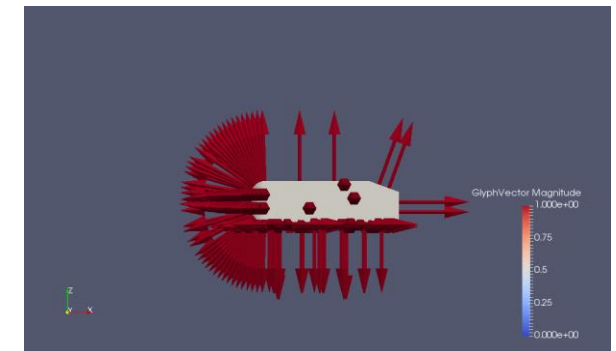
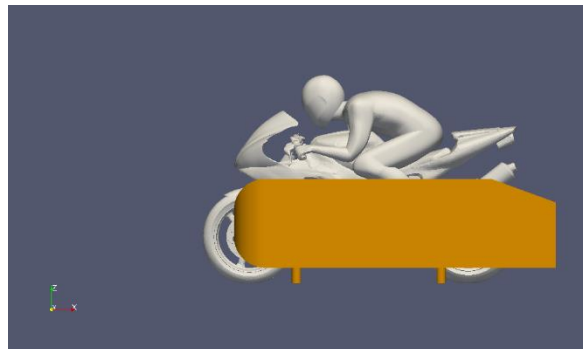
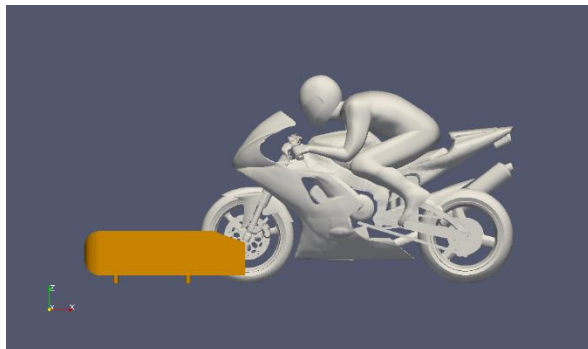
- Once we had loaded the OF environment we can get the tutorial by just copying it in our working space

```
cp -r $FOAM_TUTORIALS/incompressible/simpleFoam/motorBike .
```

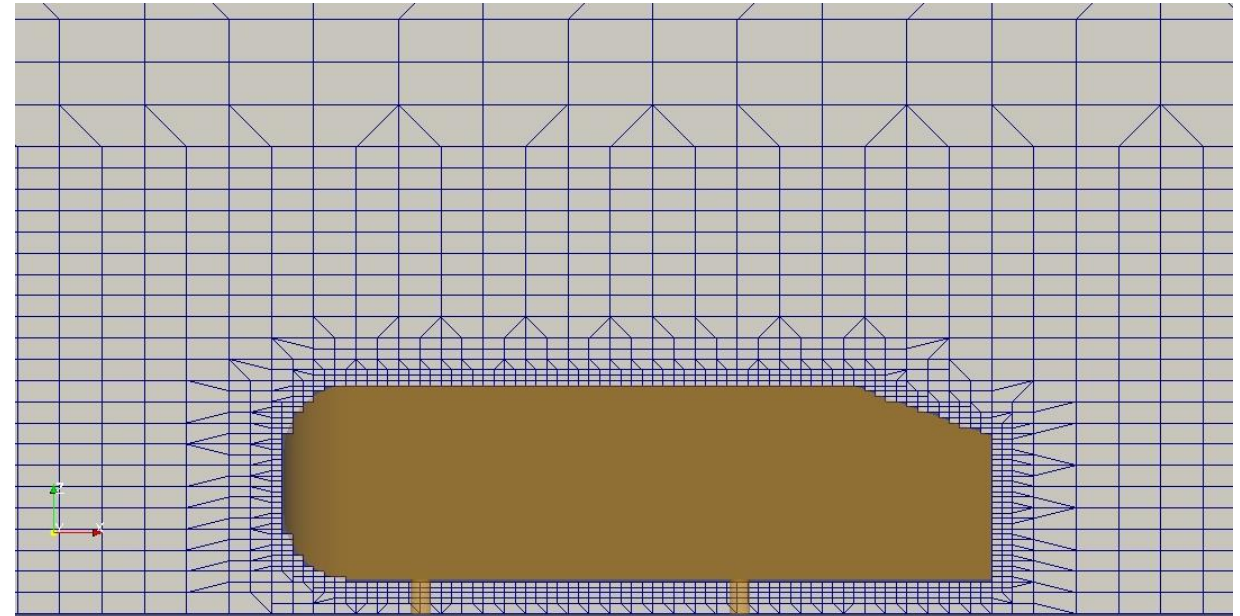
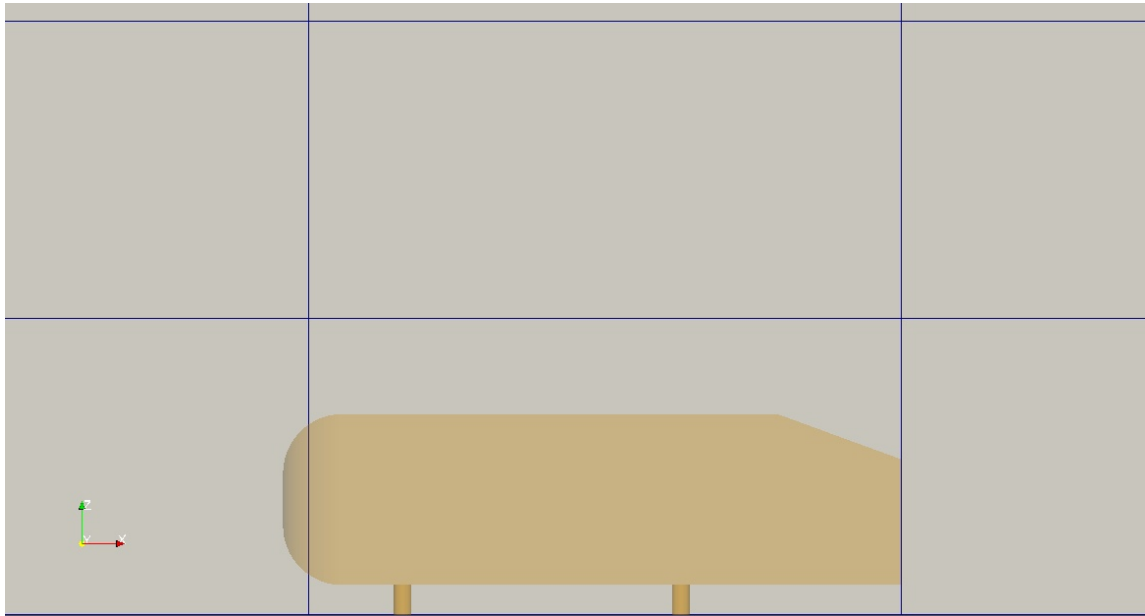
- We can then substitute the motorBike.obj geometry with our own geometry
- We can scale and move our geometry to match the dimensions/location of the motorbike.obj geometry in order to take advantage of the overall tutorial dictionary
- If we don't do this we should then modify blockMesh and snappyHexMesh dictionary to get the effective dimensions and position of our geometry

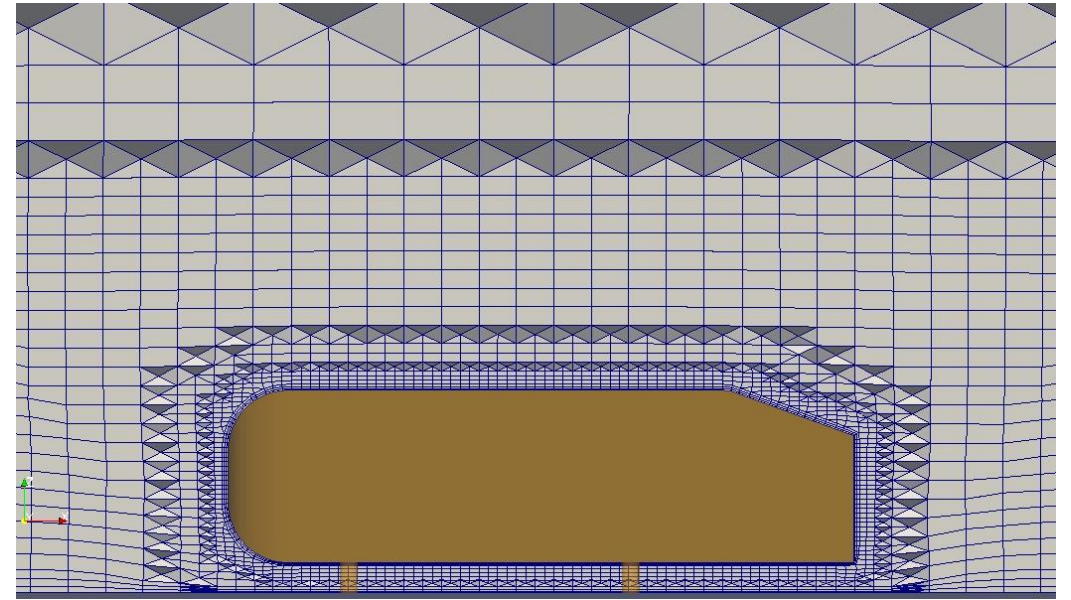
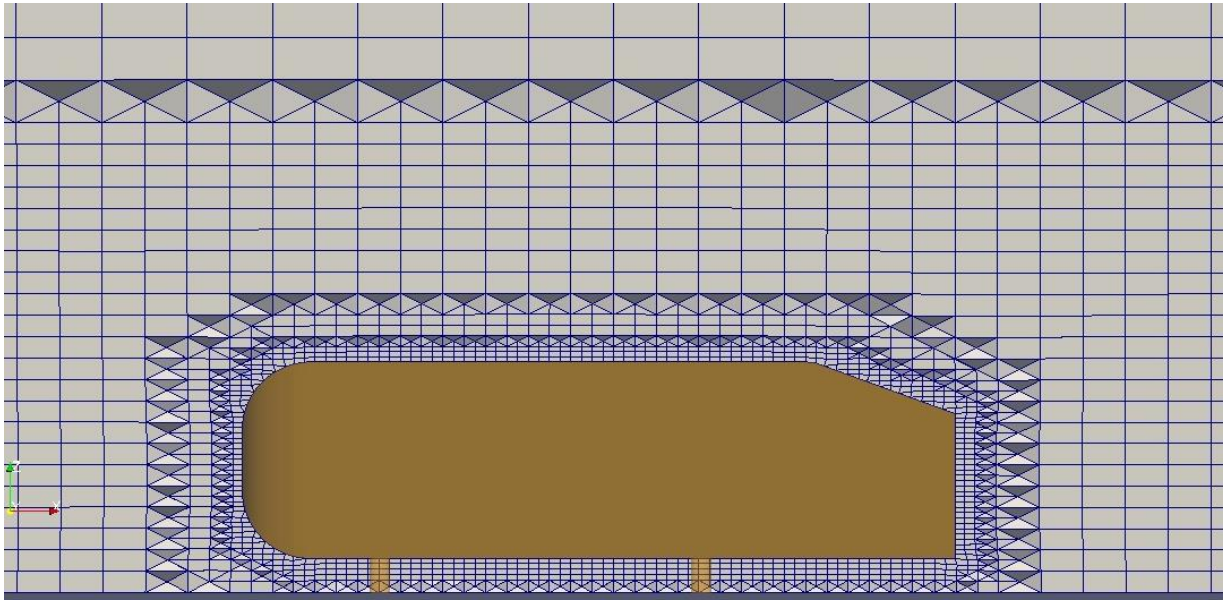
- We can move and scale the ahmed geometry to match the motorbike geometry using the convenience geometry management functions available in OF just after checking the bounding box extrema using the *surfaceCheck* function to get the values of scaling and translating:

```
surfaceTransformPoints -scale '(2 2 2)' ahmed.stl ahmed2moto.stl  
surfaceTransformPoints -translate '(2 0 0)' ahmed2moto.stl motorhacked.stl  
surfaceConvert motorhacked.stl constant/triSurface/motorBike.obj
```

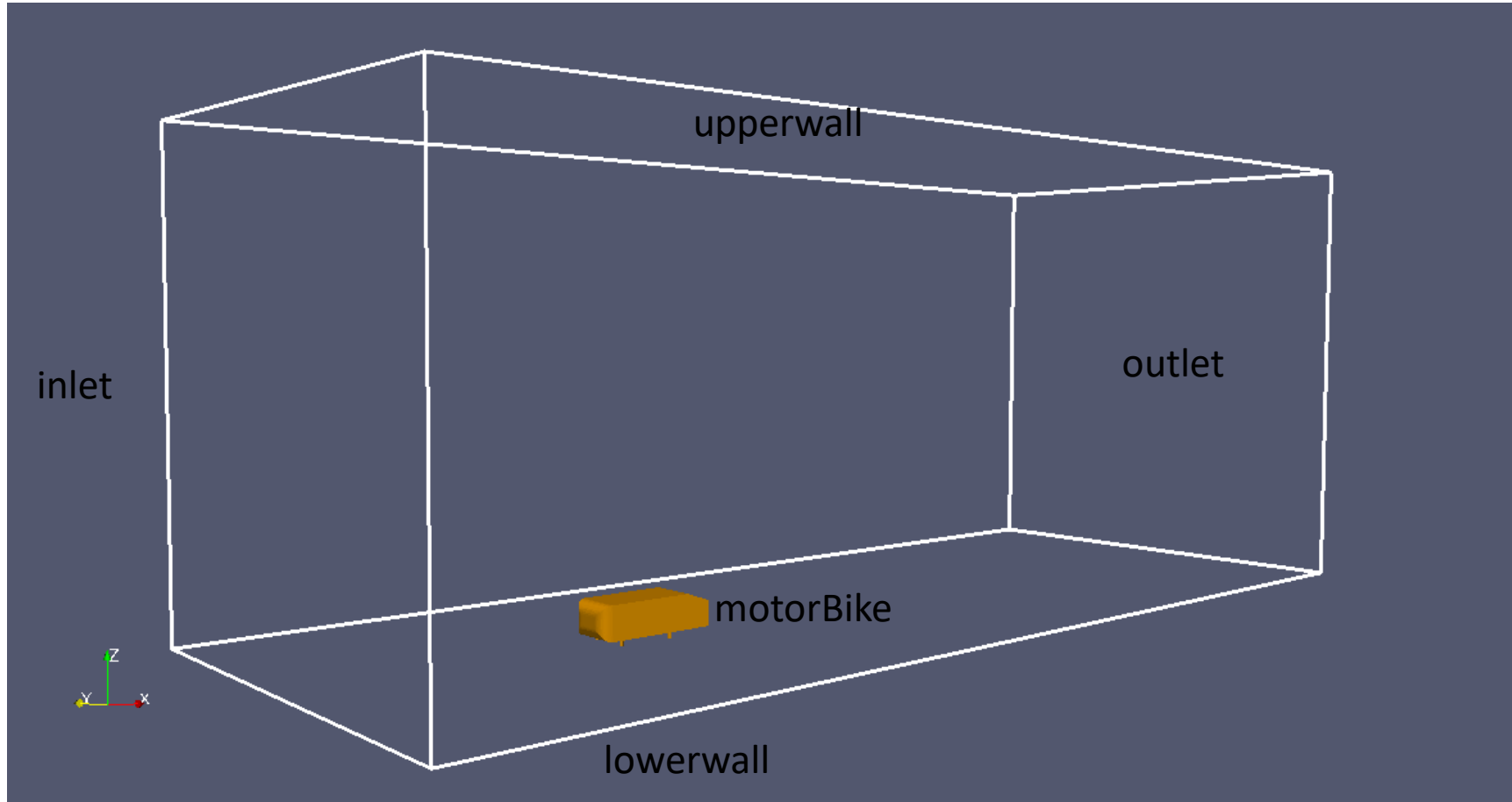


- Now that we are set with the geometry we can take advantage of the overall tutorial settings and we can run it in a 'blind' way using the *Allrun* script (commenting the geometry copy part)





- The virtual wind tunnel test case is a steady state case for an incompressible fluid (air)
- The U_{inlet} value is set at 20 m/s
- The Reynolds number order of magnitude is $2.0E+06$ so fully turbulent
- The motorBike tutorial uses a $k\Omega SST$ turbulent model to enable the closure of the N-S equations for the RANS solver
- The reference freestream pressure is set to 0



- As said is a virtual wind tunnel case
- The BC setting are very standard and we can use the *patchSummary* function to have a quick look

```

patch  : inlet
  scalar      nut      calculated
  scalar      k        fixedValue
  scalar      p        zeroGradient
  scalar      omega    fixedValue
  vector      U        fixedValue

patch  : outlet
  scalar      nut      calculated
  scalar      k        inletOutlet
  scalar      p        fixedValue
  scalar      omega    inletOutlet
  vector      U        inletOutlet

```

wall : **lowerWall**

<i>scalar</i>	<i>nut</i>	<i>generic</i>
<i>scalar</i>	<i>k</i>	<i>generic</i>
<i>scalar</i>	<i>p</i>	<i>zeroGradient</i>
<i>scalar</i>	<i>omega</i>	<i>generic</i>
<i>vector</i>	<i>U</i>	<i>fixedValue</i>

patch : **upperWall&sides**

<i>scalar</i>	<i>nut</i>	<i>calculated</i>
<i>scalar</i>	<i>k</i>	<i>slip</i>
<i>scalar</i>	<i>p</i>	<i>slip</i>
<i>scalar</i>	<i>omega</i>	<i>slip</i>
<i>vector</i>	<i>U</i>	<i>slip</i>

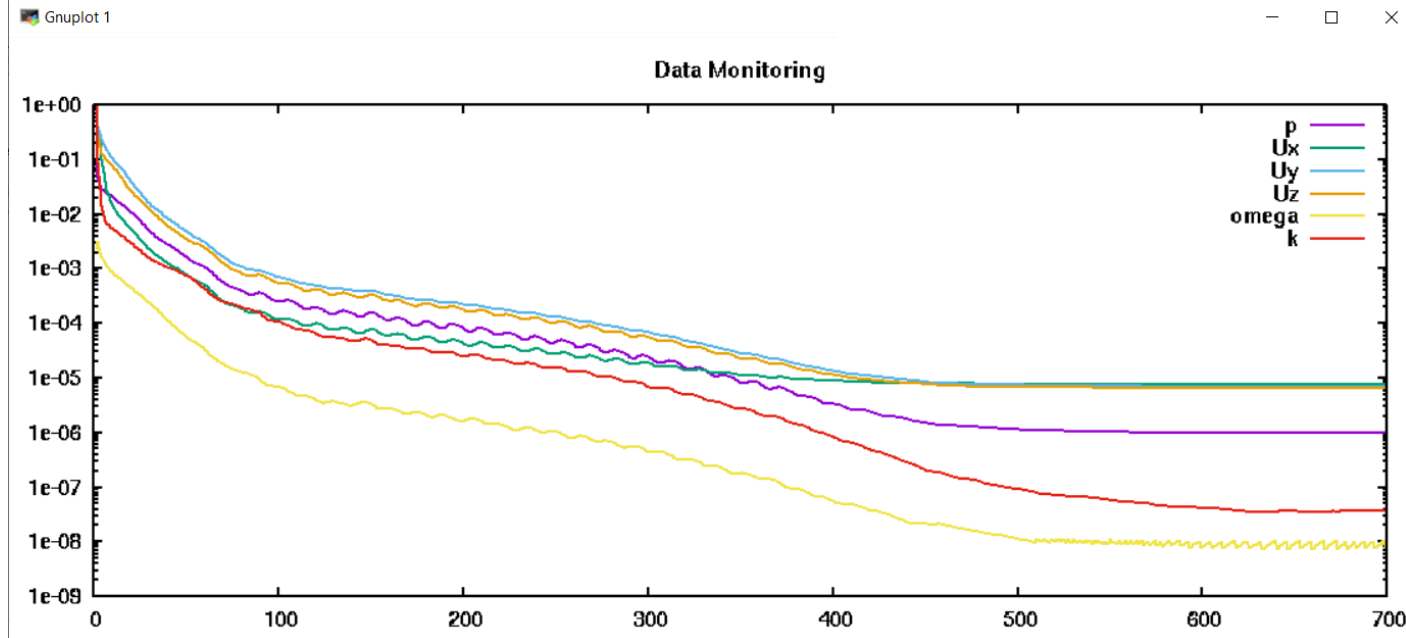
wall : **motorBike**

<i>scalar</i>	<i>nut</i>	<i>generic</i>
<i>scalar</i>	<i>k</i>	<i>generic</i>
<i>scalar</i>	<i>p</i>	<i>zeroGradient</i>
<i>scalar</i>	<i>omega</i>	<i>generic</i>
<i>vector</i>	<i>U</i>	<i>noSlip</i>

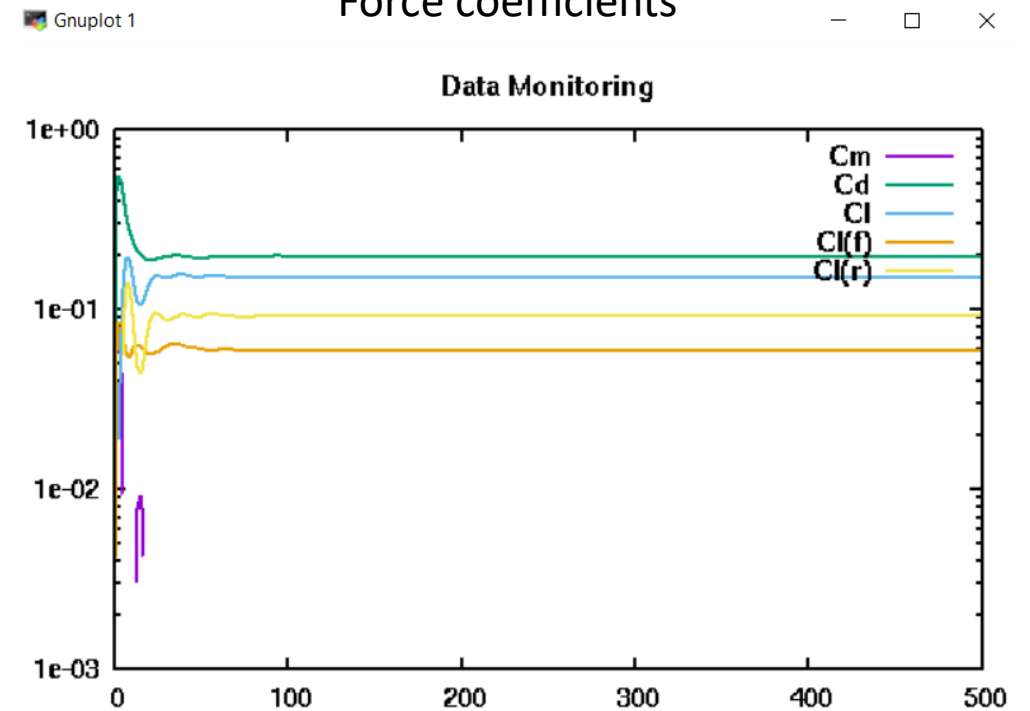
- The fluid is air at ambient temperature
- Into the constant/transportProperties dictionary we have

```
transportModel    Newtonian;  
nu                [0 2 -1 0 0 0 0] 1.5e-05;
```

Residuals

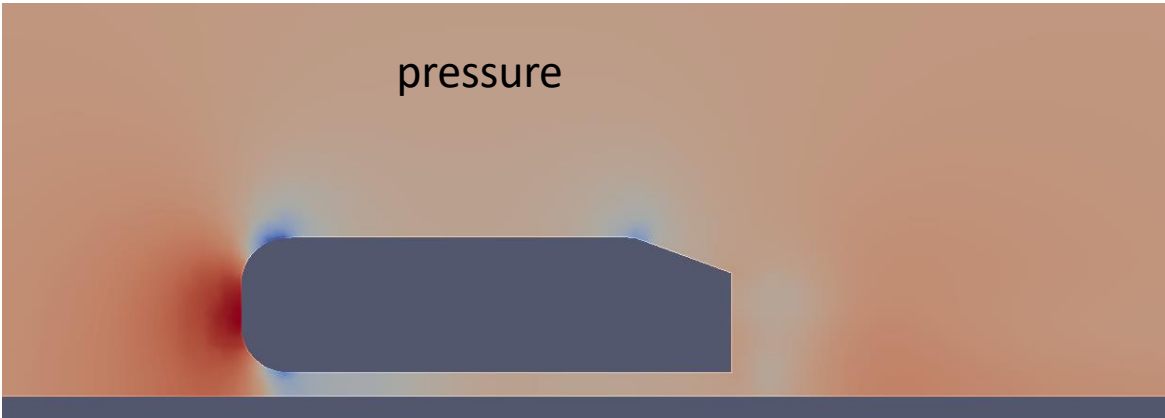


Force coefficients

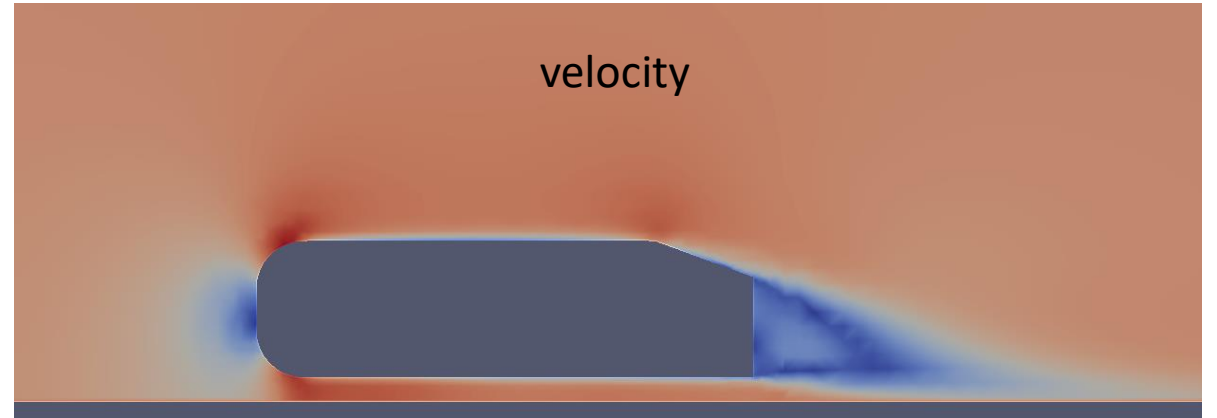


```
foamMonitor -logscale -l postProcessing/residuals/0/residuals.dat  
foamMonitor -l postProcessing/forceCoeffs1/0/forceCoeffs.dat
```

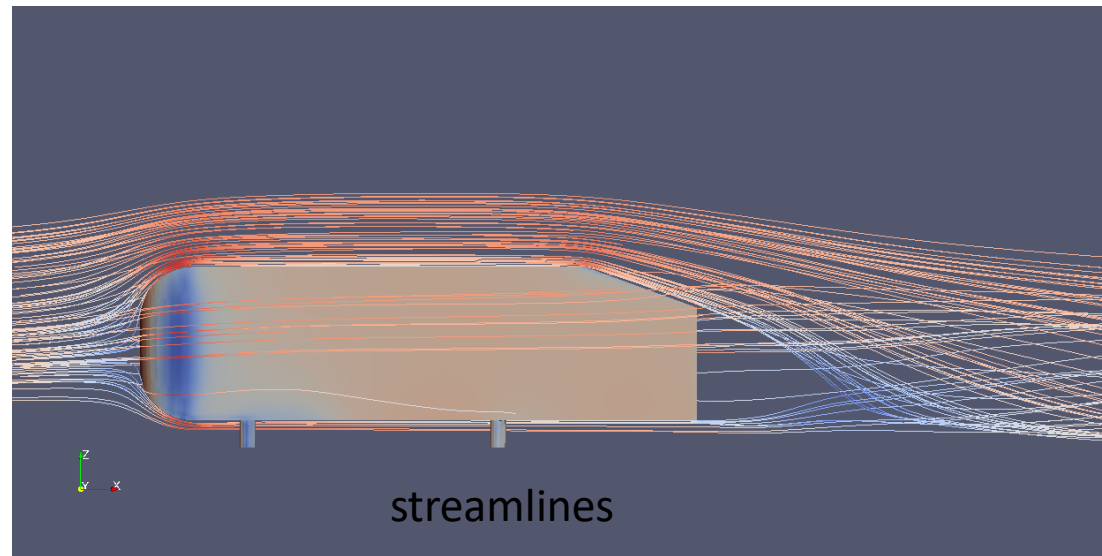
pressure



velocity



streamlines



- This course is hosted by a group of HPC centers, it is important to understand which are the main differences between using OF on our local PC or using an HPC cluster:
 - The access to computational resources (so called computational nodes) is performed using a job scheduler that need to be instructed (use user guide infos local to the hpc center)
 - Most computer centers are already providing dedicated modules to enable the correct usage of the OF environment instructed (use user guide infos local to the hpc center)
 - The GUI for plotting and data visualizing can be slowed down if there is no dedicated graphic data compression protocols instructed (use user guide infos local to the hpc center)
- The use that I suggest is to develop locally to your PC the strategy for meshing and solving and use the HPC infrastructure for number crunching

- Staying at the ‘haked’ *motorBike* tutorial presented so far the script that will do the work, assuming that:
 - You are working on a *myTest/* dir
 - you have the *ahmed.stl* geometry in a *myCAD/* subdir in *myTest/*
 - **Load the env for OF using the module:**

```
>> module load OpenFOAM/8-foss-2020b
```
 - **Get all the files from the original tutorial:**

```
>> cp -r $FOAM_TUTORIALS/incompressible/simpleFoam/motorBike/* .
```
 - **Modify the Allrun script to avoid the copy of the original *motorBike.obj* file by just commenting the line number 8:**

```
#cp $FOAM_TUTORIALS/resources/geometry/motorBike.obj.gz constant/triSurface/
```

You can launch the script going in your *myTest* dir and using:

```
>> sbatch myOFrun.sh
```

```
#!/bin/bash
#SBATCH --export=ALL,LD_PRELOAD=
#SBATCH -p haswell
#SBATCH -n 6 # total number of cores requested
#SBATCH -J OF_test
#SBATCH -e sbatch.err
#SBATCH -o sbatch.out
#SBATCH -t 01:00:00 # time limit hh:mm:ss
#SBATCH --reservation=sctrainmodule purge

#load OF module to have the OF env set
module load OpenFOAM/8-foss-2020b

#transform and convert your ahmed.stl file into the motorBike.obj
cd myCAD/
surfaceTransformPoints -scale '(2 2 2)' ahmed.stl ahmed2moto.stl
surfaceTransformPoints -translate '(2 0 0)' ahmed2moto.stl motorhacked.stl
surfaceConvert motorhacked.stl ../constant/triSurface/motorBike.obj

cd ../

./Allrun
```

- OpenFOAM is a complete toolbox to perform CFD state of the art modelling
- The overall OF environment allows the user to easily setup 2D/3D models, solve it, monitoring relevant numerical and physical quantities and finally perform meaningful data visualization and analysis
- The open-source business model of such a complete CFD toolbox, in conjunction with technological development of HPC clusters and public clouds, have in some sense changed the perspective on the usage of CFD technology for many users either in academy either in industry
- *Vertical* (single larger cases) or *horizontal* (many cases at the same time) parallelism is now a reality without any license added cost
- In order to not lose this kind of '*democratization of CFD*' it is important to know that the OpenFOAM Foundation is rising funding to keep working on it: <https://openfoam.org/news/funding-2021/>

Thank you for your attention!

<http://sctrain.eu/>

Univerza v Ljubljani



TECHNISCHE
UNIVERSITÄT
WIEN



VSB TECHNICAL
UNIVERSITY
OF OSTRAVA

IT4INNOVATIONS
NATIONAL SUPERCOMPUTING
CENTER



Co-funded by the
Erasmus+ Programme
of the European Union

This project has been funded with support from the European Commission.

This publication [communication] reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.