BCAM Summer School on Advanced Computation in Fluid Mechanics New Theory of Flight

Lab 2: Adaptive HPC simulation of turbulent flow

Johan Jansson, Niyazi Cem Degirmenci, Johan Hoffman

June 30, 2016

0 HPC environment

This lab will be based on using the Beskow supercomputer at KTH. You should have received an account at the beginning of the summer course. For login information, see the info page at PDC (https://www.pdc.kth.se/resources/software/login-1).

Once the configuration on your local computer is complete, you should be able to login like so:

ssh username@beskow.pdc.kth.se

You will also need to install ParaView (http://www.paraview.org/) on your local computer for the visualization.

To visualize, copy the automatically generated *.vtu* files to your local computer and open the files with ParaView.

1 Introduction

In this lab session you will use the FEniCS [1] framework for automated solution of partial differential equations (PDE) to formulate finite element methods (FEM) and solver the resulting discrete systems. You will specifically use the FEniCS-HPC branch with good scaling on supercomputers. The aim is to compute solutions for the incompressible Euler equations for a cube and wing geometry with adaptive error control, and a slip boundary layer model, which allows very cheap computation compared to no-slip, which requires an enormous expense in resolving the boundary layer.

The goal of this session is to:

1. Learn how to use FEniCS-HPC on a supercomputer: how to login, how to install and compile FEniCS-HPC and solvers with C++, how to launch simulation jobs, how to visualize the solution and how to modify the solver.

2. Adaptively compute solutions to the incompressible Euler equations with slip boundary condition for a cube case, representing a basic case for investigating adaptivity and stabilization for turbulent flow.

   investigating the slip-separation mechanism.

3. Adaptively compute solutions to the incompressible Euler equations with slip boundary condition for a NACA0012 wing case, investigating the mechanism of flight and quantitatively validating the solution against experiments.

2 FEniCS-HPC installation

1. Login to Beskow
   `ssh username@beskow.pdc.kth.se`
2. Copy the Summer School installation package to your local directory, unzip and cd the directory:
   ```
   cp /cfs/klemming/nobackup/j/jjan/bcam-ss/fenics-hpc-bcam-ss.zip .
   unzip fenics-hpc-bcam-ss.zip
   cd fenics-hpc-bcam-ss
   ```
3. Build and install the package (use: source init_beskow.sh for just setting up the environment):
   `source build_beskow.sh`
4. Build unicorn-minimal:
   ```
   cd unicorn-minimal
   make -j 4
   ```

3 Simulation cases

Meshes and data for two simulation cases are given in the package: a cube and a NACA0012 wing. The aim of this lab is to carry out these simulations yourself on Beskow, with the possibility to inspect and modify the unicorn-minimal/FEniCS-HPC source code, and to validate the results against the experimental data for the wing, and also verify e.g. the convergence of the adaptive method to a stable drag value for the cube.

Another important aspect is visualization, here you can perform advanced visualization with ParaView of the simulation results.

3.1 Cube case

Example visualizations for the cube case are given in figures 3.1-3.1.

The procedure to start the cube simulations are as follows:

For the cube case, the primary aim is to verify that the simulation framework works correctly, and to explore the adaptiviy. Each adaptive iteration is stored in the folder iter_XX with XX the number of the adaptive iteration.

1. cd cube_sim01 (directory for example simulation)
2. cp ../demo . (copy the just-compiled program)
3. sbatch job-adaptive.script (run the adaptive simulation)
4. Inspect the log1 file for diagnostics, and eventually the .vtu solution files

3.2 NACA0012 wing case

Example visualizations for the NACA0012 wing case are given in figure 3.2.

The procedure to start the NACA0012 simulations are as follows:

1. cd wing_sim01 (directory for example simulation)
2. cp ../demo . (copy the just-compiled program)
3. sbatch job-adaptive.script (run the adaptive simulation)
4. Inspect the log1 file for diagnostics, and eventually the .vtu solution files

The angle of attack is given in the job-adaptive.script file as an argument to the solver program. One possibility is to study a few different angles of attack to see the difference in the flow field.

References

[1] FEniCS. Fenics project. *http://www.fenicsproject.org*, 2003.

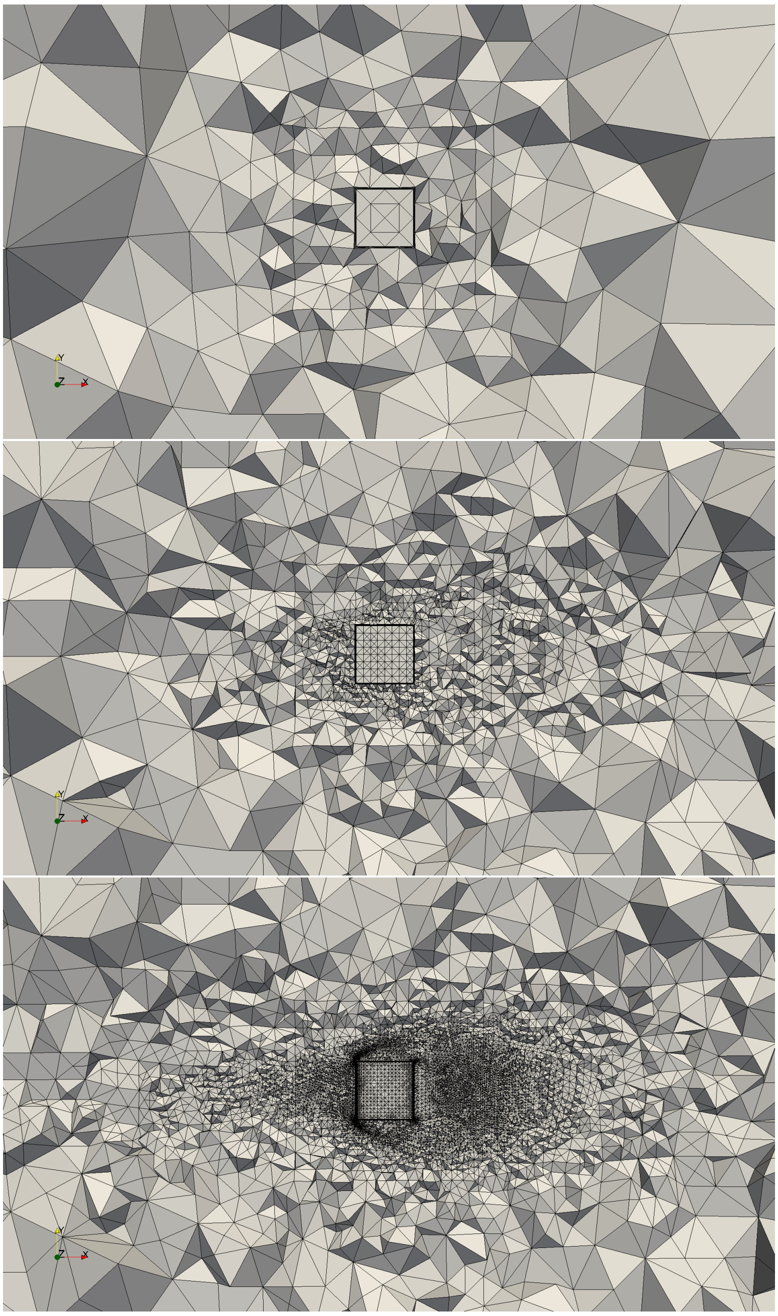

Figure 1: Clip visualizations of the mesh for the cube case.

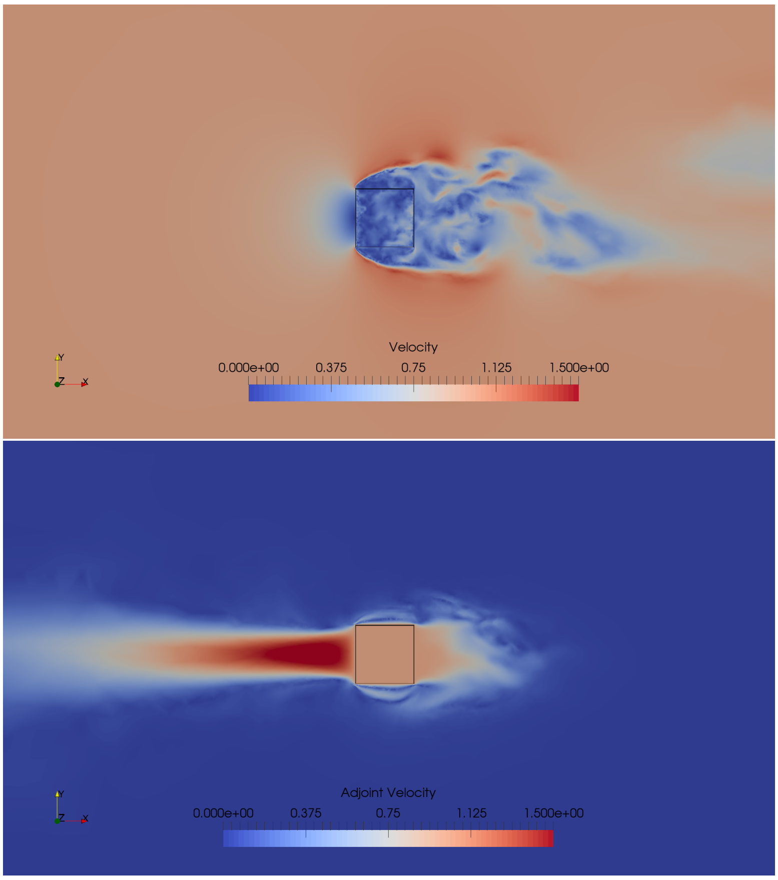

Figure 2: Primal and dual velocity visualizations for the cube case.

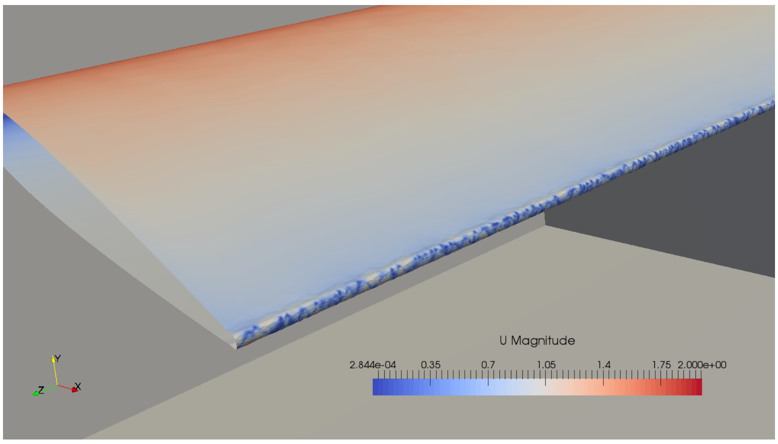

Figure 3: Primal velocity visualizations for the wing case, showing slip-separation.