Mandatory exercise MEK4470/9470, Spring 2025

1 Implement a 1D solver using TVD

Consider Example 5.1 in Versteeg et al. Implement a solver using TVD for convection and show convergence in the L2-norm. Choose any or all of the TVD schemes from the Table on p170. As a suggestion, but not a requirement, try to implement a generic TVD solver where the scheme can be chosen from the command line.
What properties do TVD schemes possess?

An L2-norm can be defined for a real function $u(x)$ in a domain Ω as

$$|u| = \sqrt{\int_\Omega u^2 dx}$$

The integral can be computed in Python using, for example, the trapz method for trapezoidal integration. See also methods for integration in Scipy. In measuring the error norm, replace u by the vector of nodal errors (your solution in the nodes minus the exact solution in the nodes).

2 Backwards facing step

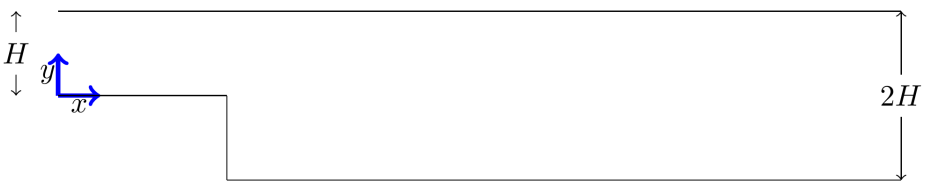


Figure 1: Backwards facing step.

Implement a backwards facing step case in OpenFOAM, see Fig. 1. Assume incompressible flow and use the icoFoam solver. Set the inlet velicity first to a constant value, and use zero gradient for velocity on the outlet. Set the pressure to a constant value 0 on the outlet and use zero gradient on the inlet. Use no-slip on the walls with zero gradient for the pressure. The Reynolds number for the backwards facing step is defined as $Re = \overline{u}H/\nu$, where ν is the kinematic viscosity and $\overline{u} = \int_0^H u(y)dy$ is the mean velocity of the inlet.

1. Use Re=200 and a constant inlet velocity. Fine the size of the recirculation bubble.
2. Use a parabolic velocity inlet profile $u(y) = \overline{u}y(H-y)$ and recompute the Re=200 case. Is there a difference for the recirculation bubble?
3. Use Re=1000. In this case verify that the solution becomes unstable if you are using default numerical schemes. Find an appropriate scheme that removes the wiggles (upwind/TVD). Report the size of the recirculation bubble for this Re.

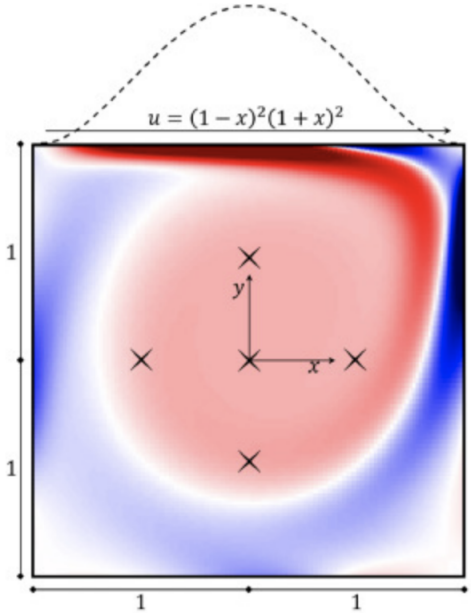


Figure 2: Regularized lid driven cavity.

3 Regularized lid driven cavity

In this task we will solve the 2D regularized lid driven cavity flow, illustrated in Fig. 2. The computational domain is $\Omega = [-1, 1]^2$ and the flow is assumed to be laminar. The top lid has a regularized velocity $u(x, 1) = (1 - x^2)(1 + x)^2$, where $u(x, y)$ is the velocity vector component in the x-direction. All the remaining walls have no-slip boundary conditions. The Reynolds number is defined as $\mathrm{Re} = 2/\nu$, where ν is the kinematic viscosity. See also Lee et al. [1].

1. Consider the icoFoam solver. Explain how this algorithm works with reference to the implementation in the link.

2. Use icoFoam and solve the regularized lid driven cavity flow in 2D with $Re = 100$ and $Re = 1000$. Use at least 2 different numerical schemes for convection and discuss the different behaviour. Do the same numerical schemes work for both $Re = 100$ and $Re = 1000$?

3. Compute the streamfunction and plot it using Paraview.

4. Find the location of the center of the main vortex in the flow for $Re = 250$.

Hint

For the cavity the center of the vortex is located at the same place as the minimum value of the stream function, and you can compute stream functions using the foamPostProcess tool. Also, you can set nonuniform boundary conditions using the 'codedFixedValue' option. Since this is difficult to find information

about online, the solution is given here. Put this in U instead of the regular movingWall:

```
movingWall
    {
    type            codedFixedValue;
    value           uniform (1 0 0);
    name            regularized;

    code
    #{
        const vectorField& Cf = patch().Cf();
        vectorField& field = *this;

        forAll(Cf, faceI)
        {
            const scalar x = Cf[faceI][0];
            field[faceI] = vector((1-x)*(1-x)*(1+x)*(1+x),0,0);
        }
    #};
    }
```

References

[1] Michael W. Lee, Earl H. Dowell, and Maciej J. Balajewicz. A study of the regularized lid-driven cavity's progression to chaos. *Communications in Nonlinear Science and Numerical Simulation*, 71:50–72, 2019.