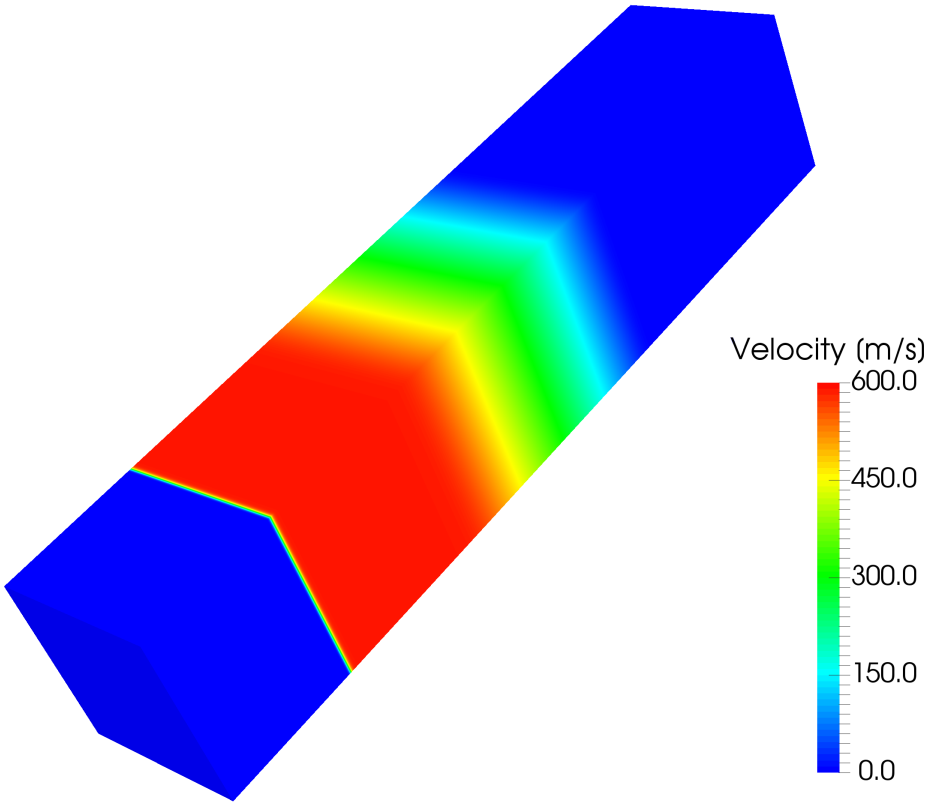
Tutorial Three

Patching Fields



5th edition, Sep. 2019

Editorial board:
- Bahram Haddadi
- Christian Jordan
- Michael Harasek

Compatibility:
- OpenFOAM® 7
- OpenFOAM® v1906

Cover picture from:
- Bahram Haddadi

Contributors:
- Bahram Haddadi
- Clemens Gößnitzer
- Jozsef Nagy
- Vikram Natarajan
- Sylvia Zibuschka
- Yitong Chen

ISBN 978-3-903337-00-8

Publisher: chemical-engineering.at

For more tutorials visit: www.cfd.at

Background

1. Initial and Boundary Conditions

Before running the numerical solver, it is important to set up initial and boundary conditions correctly for the problem. Ill-defined boundary conditions will result in non-convergence or incorrect results.

The initial conditions provide the starting values for the solver and once specified, the value is put into the center of every cell in the whole domain. As the solver starts to run, the initial values will be replaced by newly calculated values. Any starting values can be used for 1st iteration. However the better the initial values, the faster the convergence. Initial conditions are mandatory for transient problems, but not absolutely required for steady state problems.

On the other hand we need to also provide boundary conditions. These will connect the simulation models with its surroundings. The values specified are located at the boundary faces of the domain, where their solution will be kept unchanged during the simulation, as the solver will not calculate them. Most boundary conditions are either steady state or transient.

In OpenFOAM®, we can use the *setFields* utility to specify a non-uniform initial condition, and this will be the focus of Tutorial Three. In addition, the boundary conditions are specified in the files in the 0 directory.

2. Courant-Friedrichs-Lewy (CFL) condition

In this tutorial, we will create meshes with 100, 1000 and 10000 cells in one dimension. However, one cannot simply increase the number of cells (i.e. decrease the cell size) without changing the time step size accordingly. This is because when running a numerical scheme (e.g. the Gauss linear scheme; more details can be found in Tutorials Four and Five), the solution is reached using the information propagated by waves or particles from one cell to the adjacent cell. To ensure a physical solution it is essential that the physical flow information e.g. velocity, composition, etc. is received by all cells within the calculation domain. It needs to be guaranteed that the information transport does not "overtake" the physical transport, otherwise the scheme will be unable to access the information required to form the solution.

The above criteria is known as the Courant-Friedrichs-Lewy (CFL) condition, and it is a necessary condition for convergence. For one-dimensional problems, it can be written as:

$$Co = \frac{u\Delta t}{\Delta x} \leq 1$$

The key dimensionless number here is the Courant number, Co, which needs to be less or equal to one.

Note: u is the velocity magnitude of compound in the 1D direction, Δt is the simulation time step size and Δx is the mesh size in the 1D direction.

As it is obvious from the equation by decreasing the mesh size (i.e. Δx), the time step size (Δt) should also be adjusted (decreased) for reaching a stable and convergent solution. Therefore the CFL condition is useful in helping us choose a suitable time step size for our simulation. A common

way of selecting the time step size is to keep Courant number at 1, using the maximum possible u and the smallest possible mesh size, a Δt that fits the criteria can be calculated.

rhoPimpleFoam – shockTube

Simulation

Use the rhoPimpleFoam solver; simulate 0.007 s of flow inside a shock tube, with a mesh with 100, 1000 and 10000 cells in one dimension, for initial values 1 bar/0.1 bar and 10 bar/0.1 bar.

OpenFOAM® v1906: use sonicFoam!

Objectives

- To understand the setFields utility
- Learn how to specify initial and boundary conditions
- Investigate effect of grid resolution

Data processing

Import your simulation into ParaView, and compare results.

1. Pre-processing

Open tutorial

Copy the tutorial from the following directory to your working directory

`$FOAM_TUTORIALS/compressible/rhoPimpleFoam/laminar/shockTube`

OpenFOAM® v1906: `$FOAM_TUTORIALS/compressible/sonicFoam/laminar/shockTube`

OpenFOAM® v1906: Create a copy of 0.orig folder and rename it to 0!

system directory

Looking at the *blockMeshDict* file, it is obvious that it is a 1D mesh, because of the number of mesh cells in y and z directions is one, and also in `boundary` section, plates vertical to these directions are defined as `empty`. The mesh density can be set in the `blocks` part by changing x direction mesh size (e.g. change it from `1000` to `100` or `10000`).

Another important file is *setFieldsDict*, which is used by the tool `setFields` for patching (assign an amount to a region) in the simulation. For example, here a pressure of 1 bar is set as the default value for the entire region (from -5 to 5 in x direction), then half of the region (from 0 to 5) is patched with a pressure of 0.1 bar.

```
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *//

defaultFieldValues ( volVectorFieldValue U ( 0 0 0 ) volScalarFieldValue T 348.432
volScalarFieldValue p 100000 );

regions         ( boxToCell { box ( 0 -1 -1 ) ( 5 1 1 ) ; fieldValues ( volScalarFieldValue T
278.746 volScalarFieldValue p 10000 ) ; } );

// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *//
```

In the `defaultFieldValues`, a value is assigned to the whole domain, for example here, the velocity has been set everywhere to zero, the temperature to 348.432 K, and the pressure to 100000 Pa. In the `regions`, a specific value is patched to a certain region of the domain. In this example the region is defined as a cube, by the coordinates of one of its diagonals in `boxToCell` .

After choosing the region, the new values are assigned to the parameters (e.g. temperature at 278.746 K and pressure at 10000 Pa).

2. Running simulation

First the mesh needs to be created:

`>blockMesh`

In order to assign the values which were set in the setFieldDict:

`>setFields`

Then run:

`>rhoPimpleFoam`

OpenFOAM® v1906: `>sonicFoam`

Note: In the 10000 cell case with 10 bar and 0.1 bar, the simulation will crash with the default `deltaT` *(*`1e-5`*); after checking the same case with 1000 cells, you will find that the maximum Co is around 0.6 (*`Time = 0.001`*):*

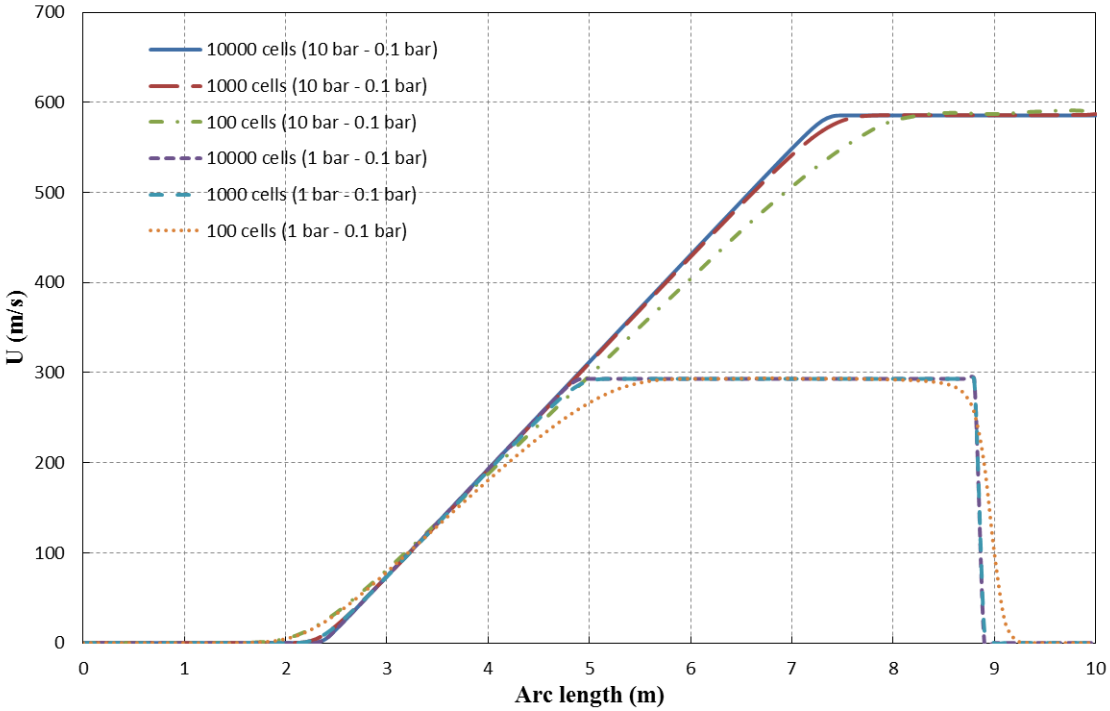
```
Courant Number mean: 0.0508555 max: 0.589018
```

In the case with 10000 cells, the number of cells is increased by factor 10, so the cell size is reduced by factor 10. For keeping the Courant number in the same range (around 0.6), according to the "Background" section, `deltaT` *should be decreased by factor 10. After reducing it to* `1e-6` *the simulation will run smoothly.*

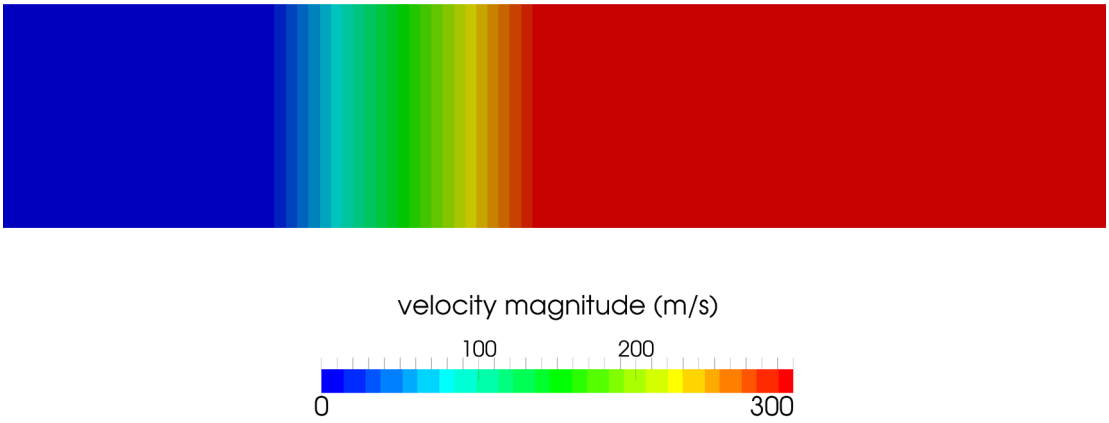
Note: After running setFields for the first time, the files in the 0 directory are overwritten. If the mesh is changed these files are not compatible with the new mesh and the simulation will fail. To solve this problem replace the files in the 0 directory with the files in the 0.orig or the files with suffix ".orig", e.g. p.orig in the 0 directory. In the OpenFOAM® files or directories with suffix ".orig" ("original") usually contain the backup files. If a command changes the original files these files can be replaced.

3. Post-processing

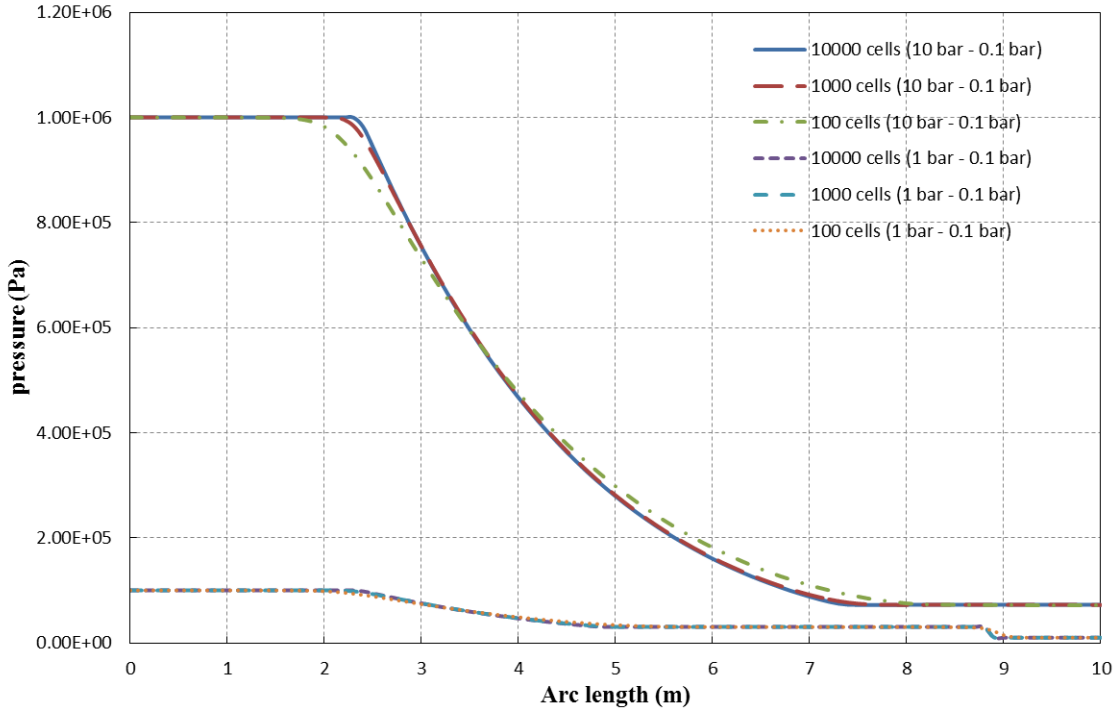
The simulation results are as follows:



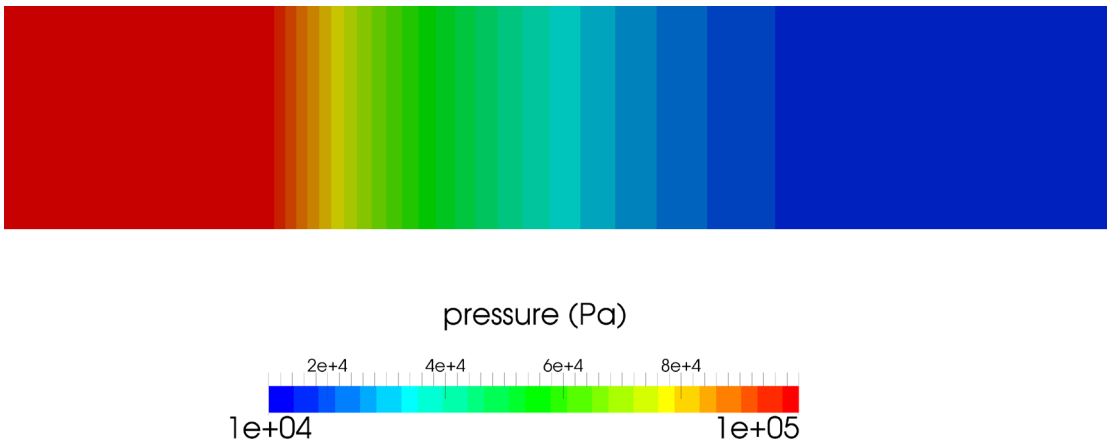
Velocities for different configurations along tube at t = 0.007 s



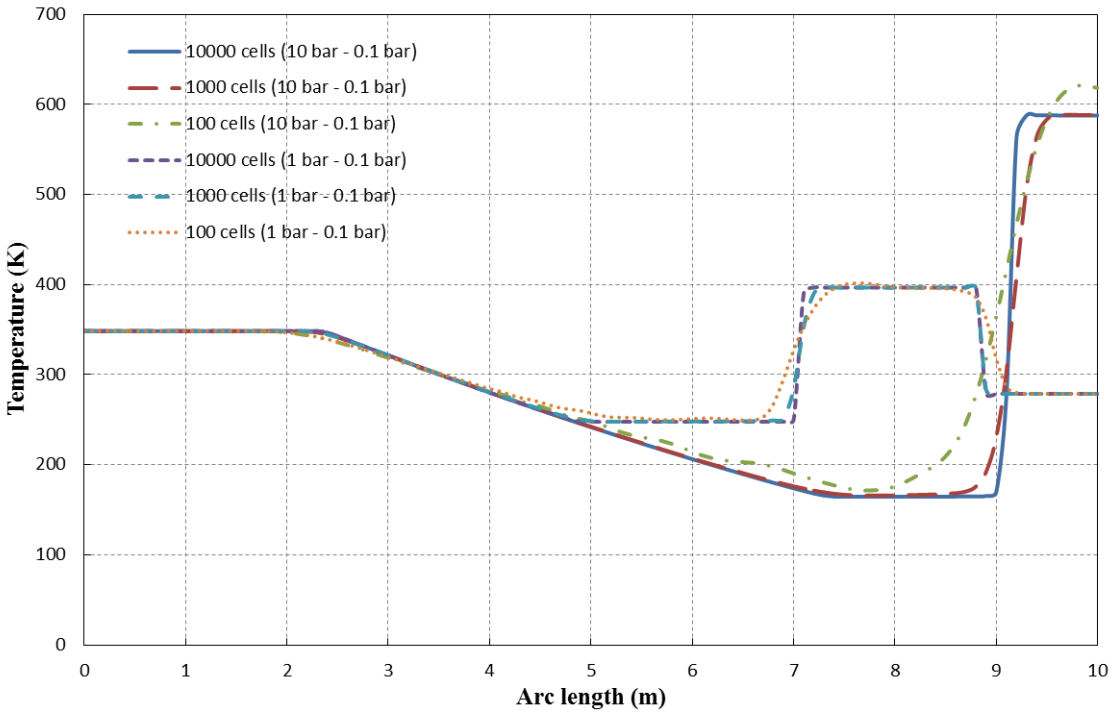
Velocity along tube axis for 10 bar/0.1bar and 10000 cells case at t = 0.007 s



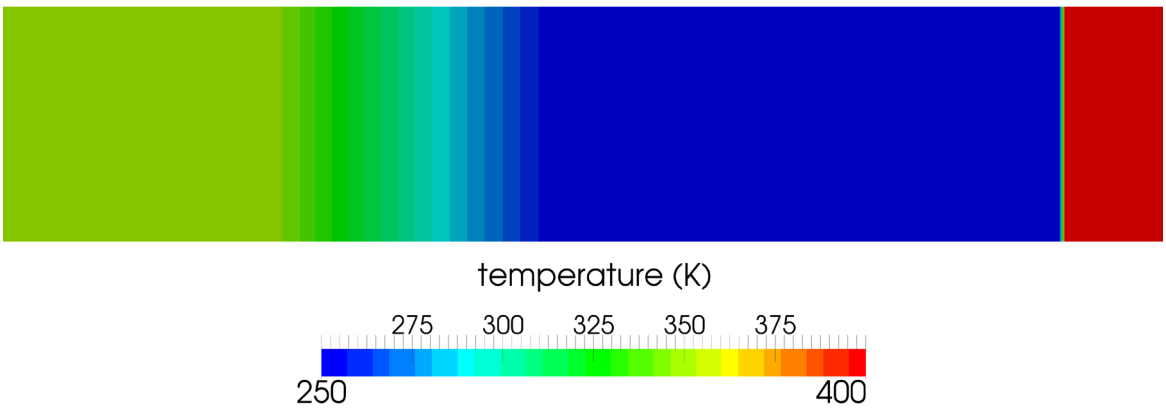
Pressures for different configurations along tube at t = 0.007 s



Pressure along tube axis for 10 bar/0.1bar and 10000 cells case at t = 0.007 s



Temperature for different configurations along tube at t = 0.007 s



Temperature along tube axis for 10 bar/0.1bar and 10000 cells case at t = 0.007 s

ISBN 978-3-903337-00-8