Computation

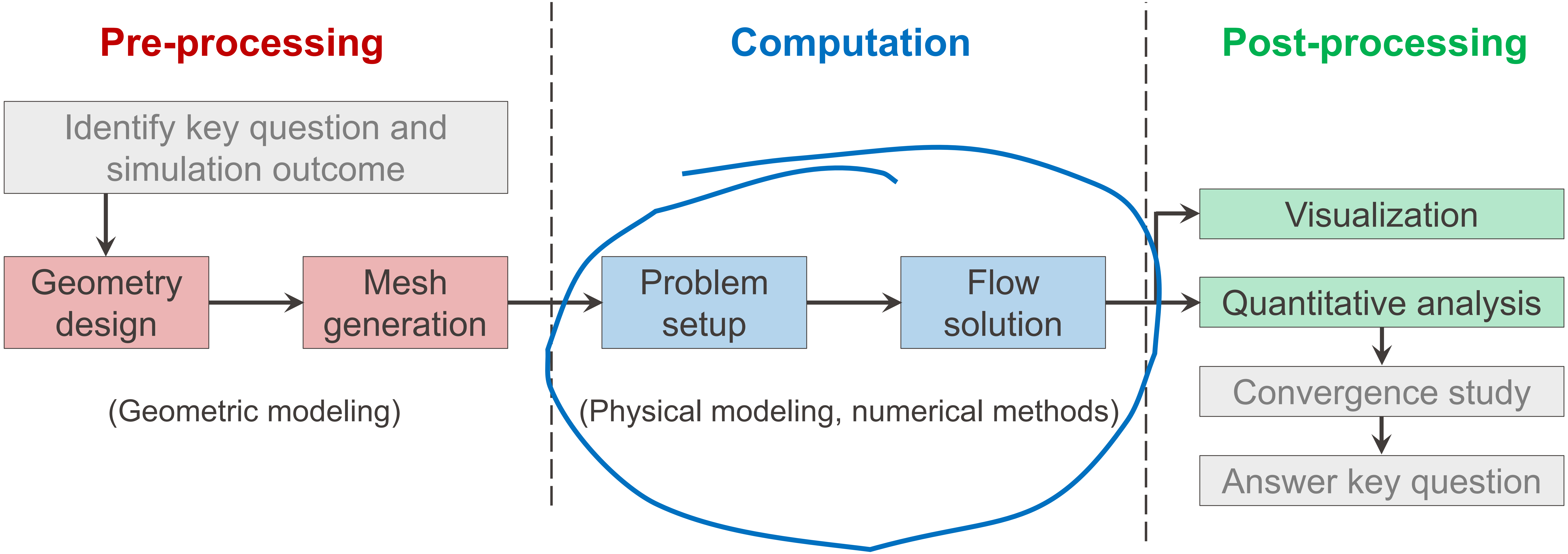
Numerical Flow Simulation

Edouard Boujo

Fall 2025

École
polytechnique
fédérale
de Lausanne

Numerical simulation workflow



- Choose physical model
- Define boundary conditions
- Choose numerical method
- Initialize solution
- Run solver
- Monitor residuals and solution to determine when convergence is reached

Choosing the physical model

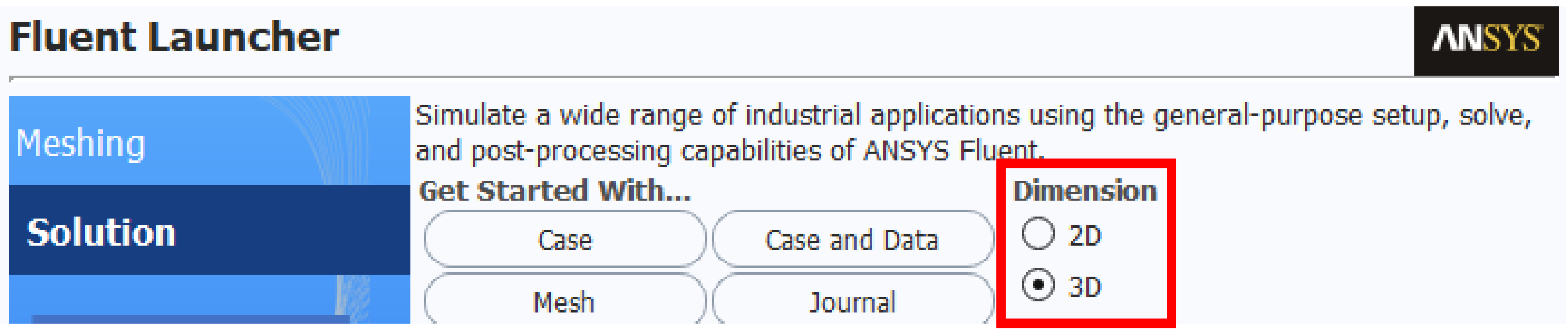
- The following physical models may be available:
  - Fluid flow (minimum default)
  - Turbulence models
  - Multi-phase flow
  - Solidification/melting
  - Heat transfer (convection, conduction, radiation)
  - Chemical species transport, reacting flows
  - User-defined scalar transport
  - …

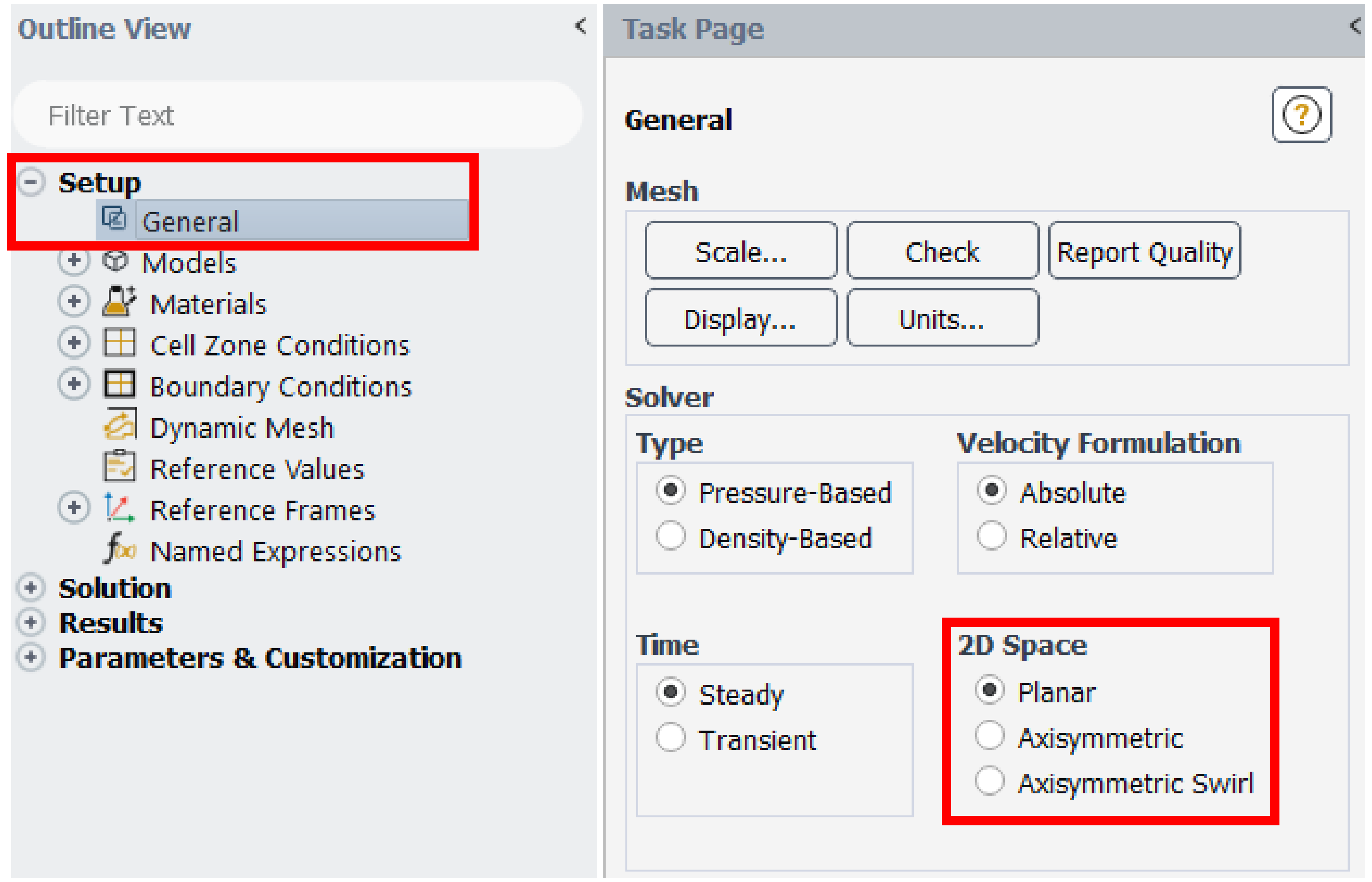
- This choice determines the set of equations to be solved → need for appropriate model parameters, fluid properties, boundary conditions, etc.

- Fluid properties (density, viscosity, thermal conductivity, specific heat…):
  - Generally defined in a database
  - Additional materials can generally be defined

Choosing the physical space: 2D vs 3D

- In Fluent:





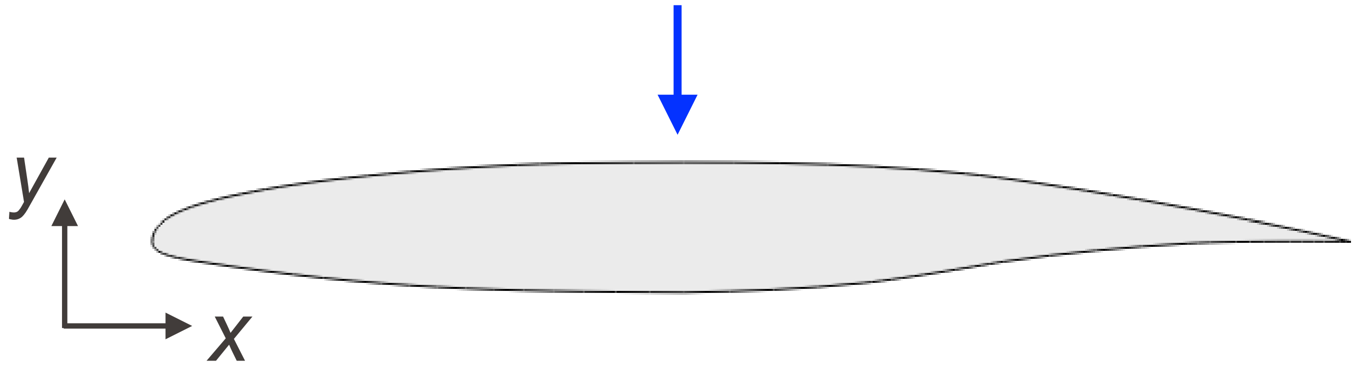
If 2D, choose between:

- **Planar**: solve for (u_x, u_y) functions of (x,y); i.e. $u_z=0$, $d/dz=0$.
- **Axisymmetric**: solve for (u_r, u_z) functions of (r,z); i.e. $u_\theta=0$, $d/d\theta=0$.
- **Axisymmetric with swirl**: solve for (u_r, u_θ, u_z) functions of (r,z); i.e. $d/d\theta=0$ but $u_\theta \neq 0$.

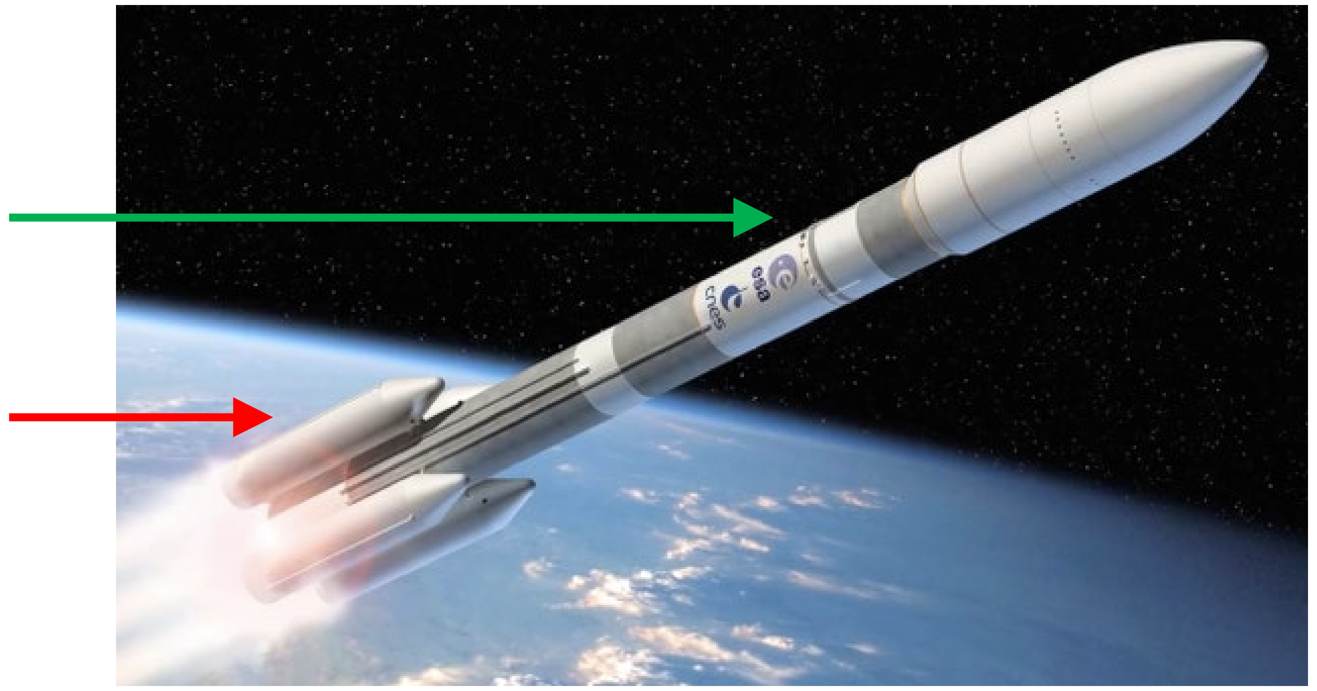
Choosing the physical space: 2D vs 3D

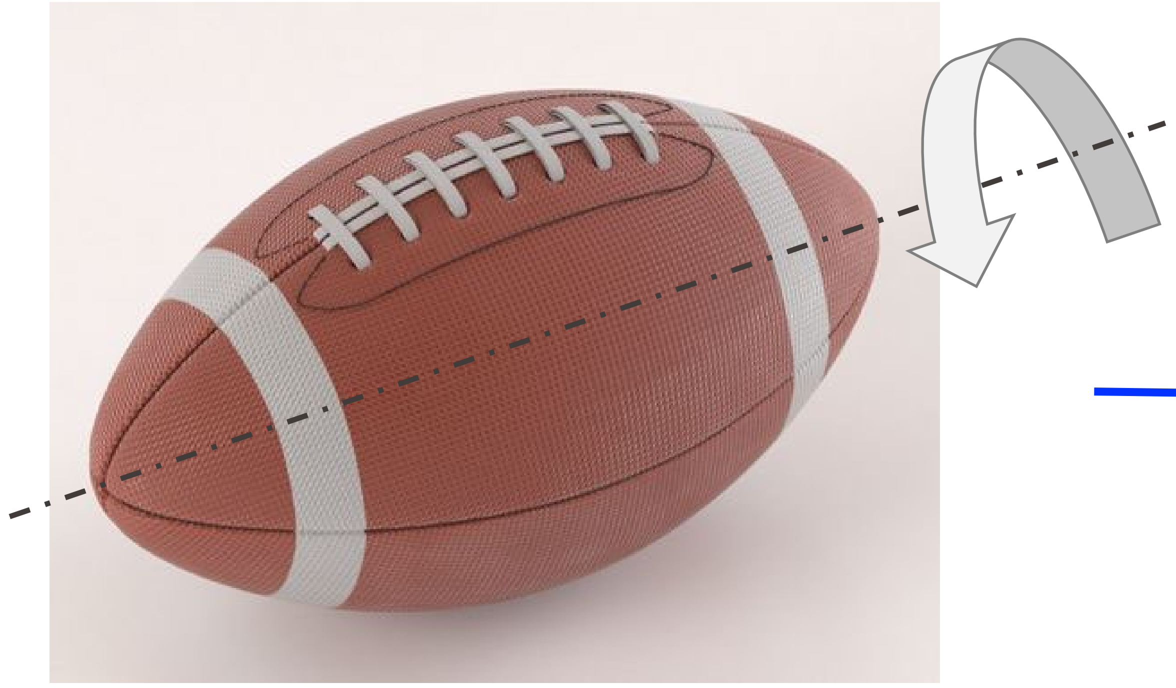
- 2D planar is a good approximation if:
  - end effects are negligible,
  - geometry varies "slowly" in z,
  - spanwise velocity u_z is "small".

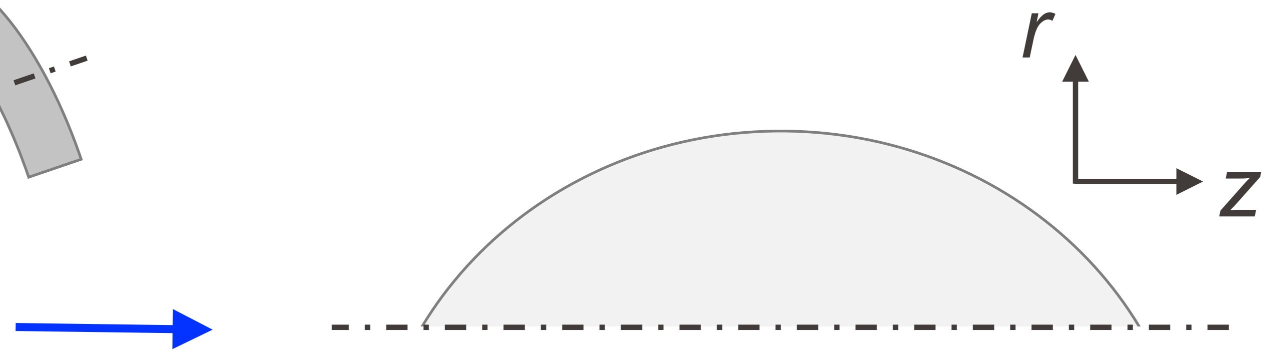
  } d/dz=0



- 2D axisymmetric:
  - applicable only if geometry and BCs are truly axisymmetric (d/dθ=0),
  - can include swirl (azimuthal velocity $u_\theta \neq 0$).

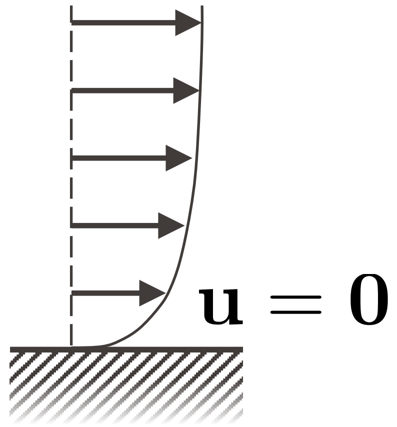


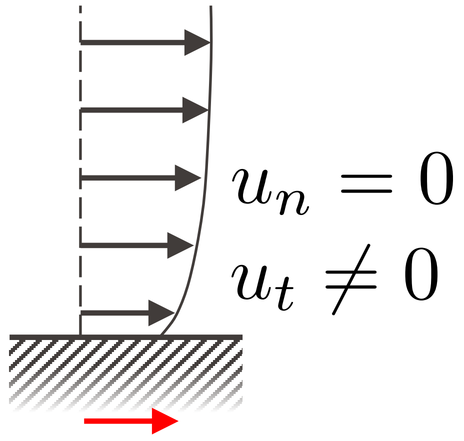




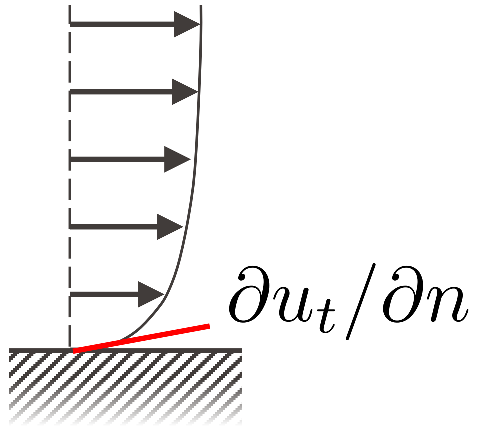
Defining boundary conditions (1/2)

- BC types generally available:
  - Inlet/outlet boundaries:
    - Velocity inlet
    - Mass flow inlet/outlet (compressible)
    - Outflow (incompressible)
    - Pressure inlet/outlet
    - Pressure far-field (compressible)
    - …
  - Wall:
    - No-slip on stationary/moving wall (specified velocity)



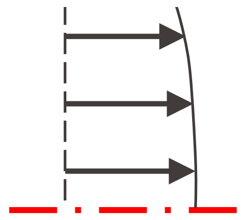


    - Shear stress (applied shear stress; slip wall)



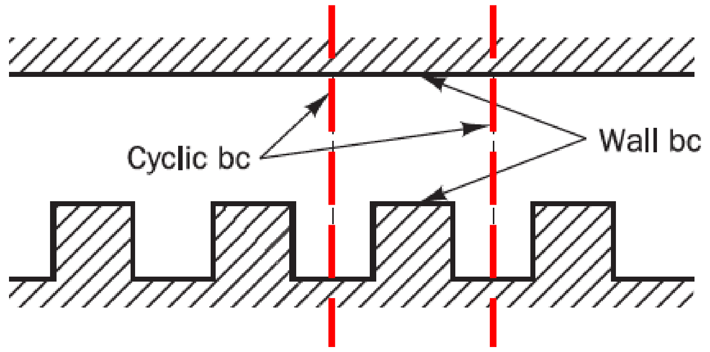
Defining boundary conditions (2/2)

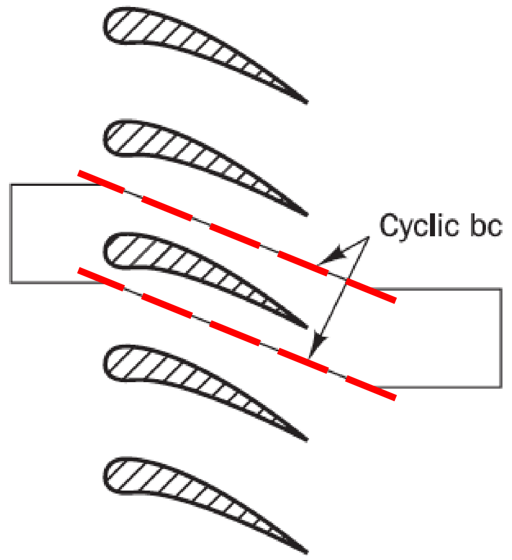
- BC types generally available:
  - Geometry reduction:
    - Symmetry

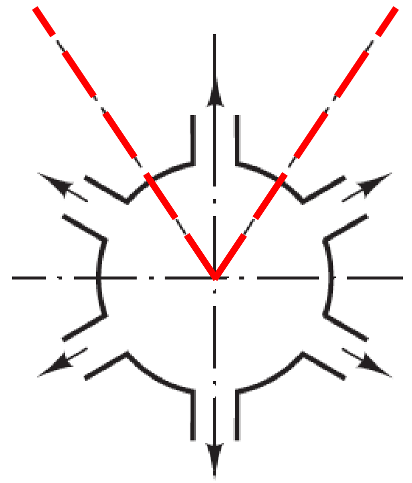


$$u_n = 0$$
$$\partial u_t / \partial n = 0$$

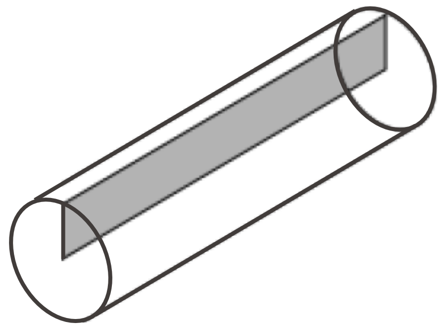
    - Periodic (translation/rotation)







    - Axis (axisymmetry)



$$u_r = u_\theta = 0$$
$$\partial u_z / \partial r = 0$$

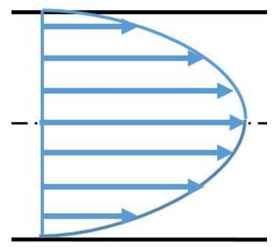
  - Internal BC:
    - Fan

    - Porous media (soil/rock, ceramic, powders, fuel cells/batteries etc.)

    - “Two-sided” wall (infinitely thin wall)
    - Solid (heat conduction, no flow)

Boundary conditions: examples

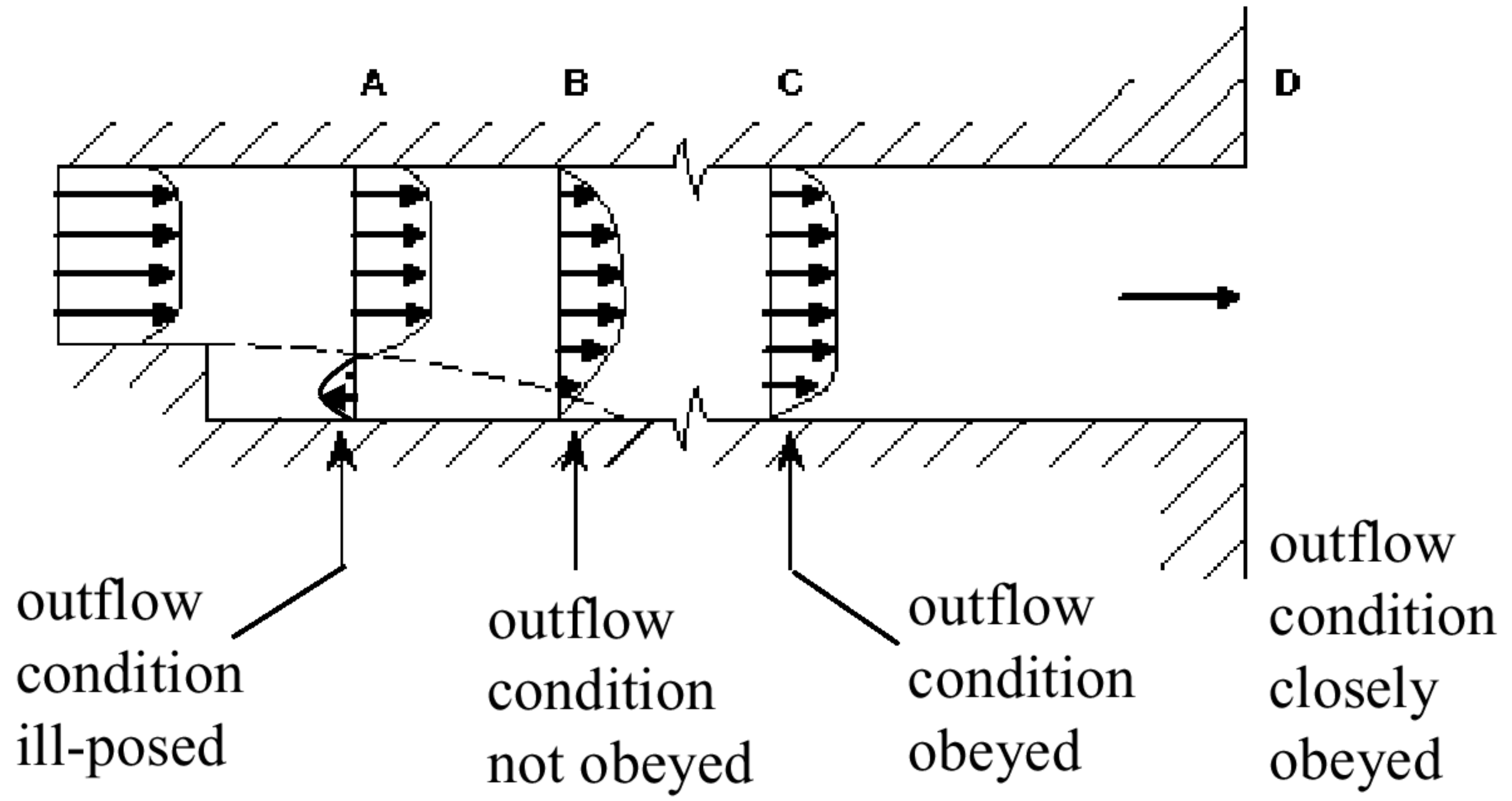
- Velocity inlet:
  - Define velocity vector (magnitude + direction)
  - Fluent default: uniform. Non-uniform condition can be imposed too.



- Pressure inlet/outlet:
  - Requires “gauge pressure” as input. At the same time, fixes reference pressure in flow.
  - Useful if unknown velocity/flow rate.
  - Useful if “free” boundary in an external flow.

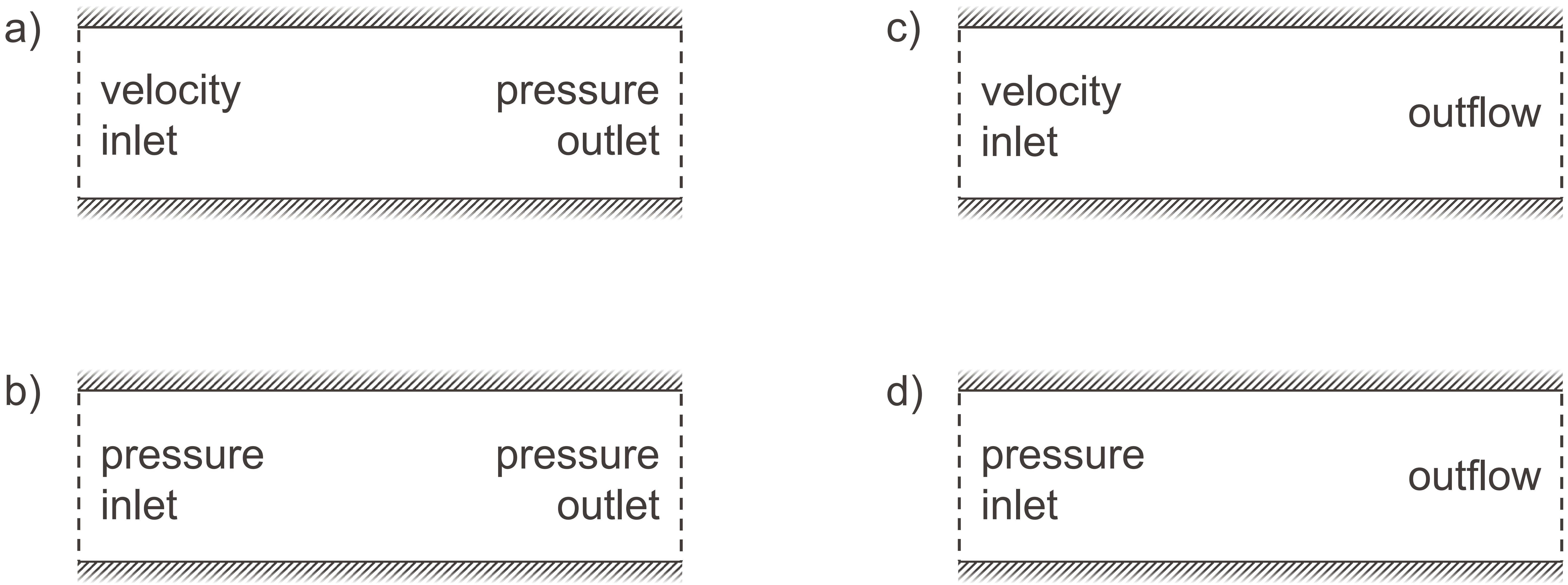
Boundary conditions: examples

- Outflow:
  - Zero normal gradients for all flow variables except pressure
  - Extrapolate required information from interior
  - Useful if unknown details of flow velocity and pressure
  - Appropriate if exit flow is close to fully developed solution
  - Not ideal if backflow (“pressure outlet” BC better in this case)
  - Only for incompressible flow



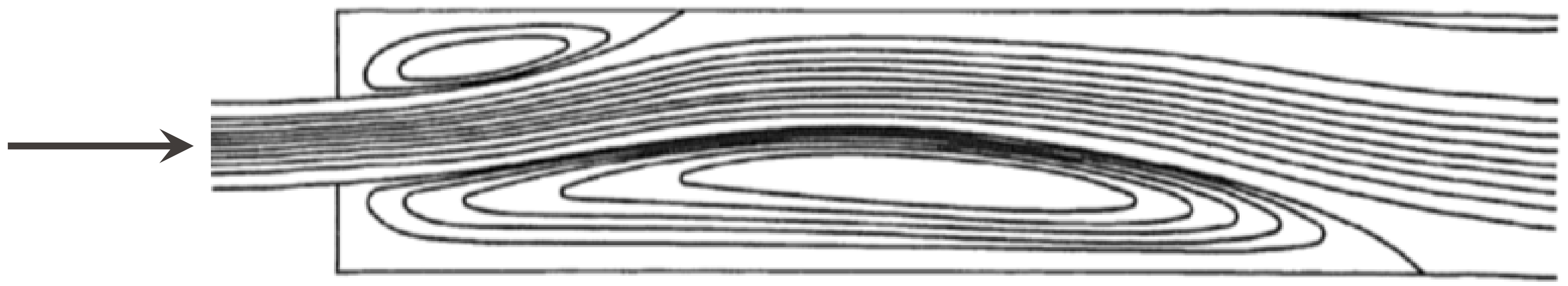
Boundary conditions

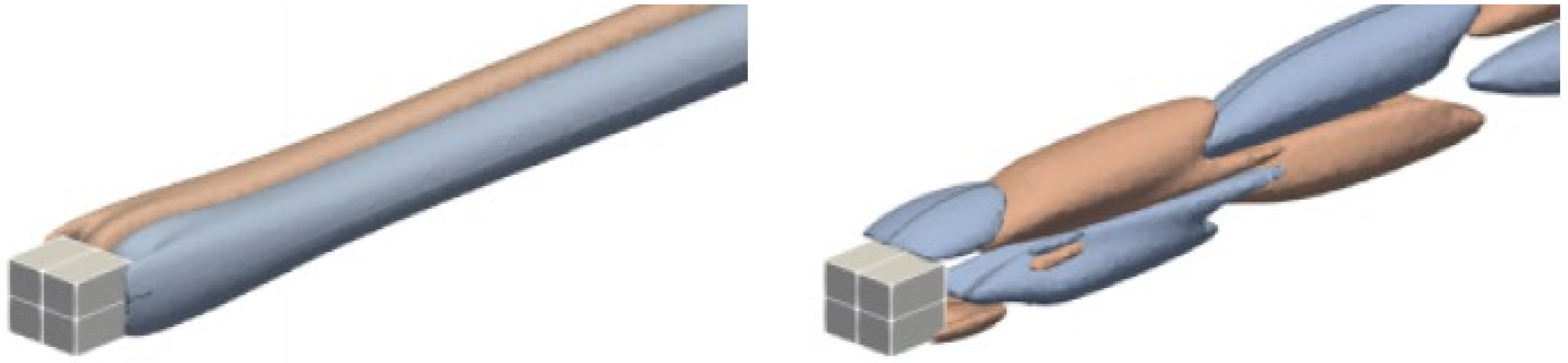
- Quiz: which configuration/s is/are **not** valid? Why?



Boundary conditions: examples

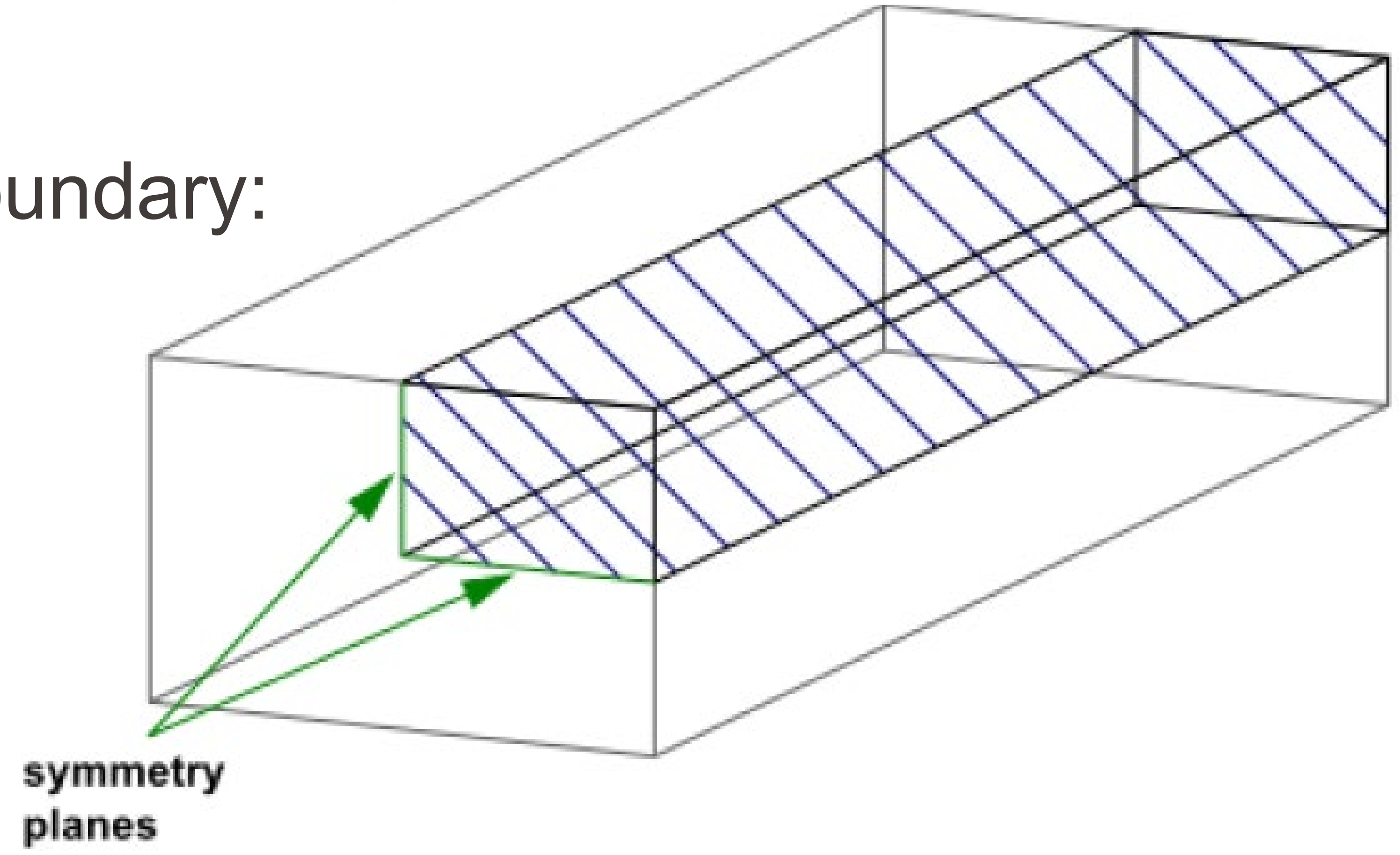
- Symmetry:
  - Very useful to reduce the size of the domain.
  - Appropriate when geometry + BCs + expected flow pattern are all mirror-symmetric. Should be used whenever possible!
  - Beware of symmetry-breaking bifurcations (symmetric geometry and BCs, but asymmetric flow).





  - Zero flux of all quantities through symmetry boundary:

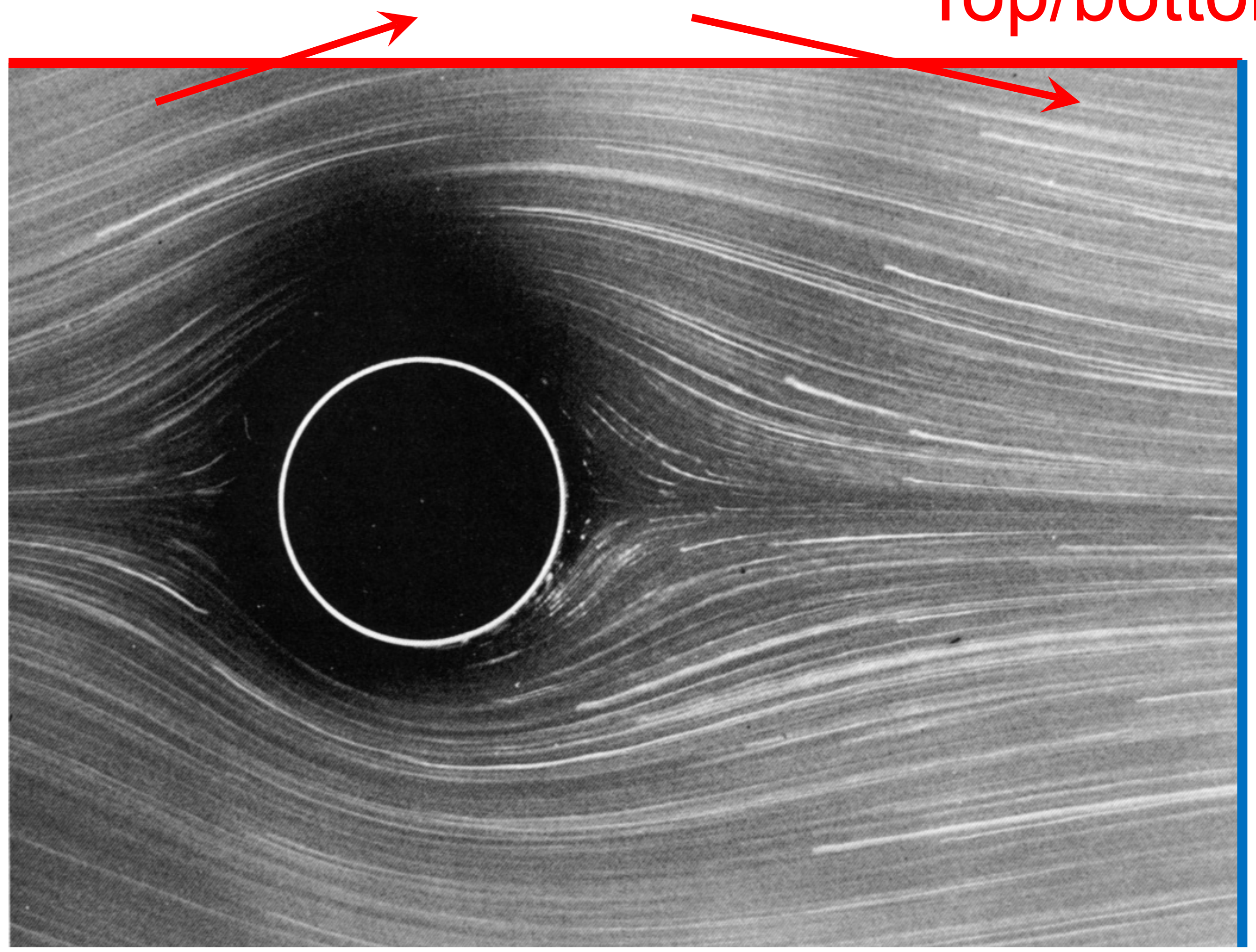
$$u_n = 0, \quad \partial\phi/\partial n = 0$$



Boundary conditions

- Solution and convergence strongly depend on choice of BCs!
- BCs are often an approximation/simplification of the reality.
  - Example: external flow

Top/bottom boundaries:



Numerical Flow Simulation

Symmetry BC $\partial u/\partial y = 0, v = 0$? Not satisfied (non-zero *v*).

“Moving wall” BC with free-stream velocity $u = U_\infty, v = 0$? Not satisfied (*u* unknown, non-zero *v*).

Slip BC $\partial u/\partial y = 0$? Not satisfied (shear unknown).

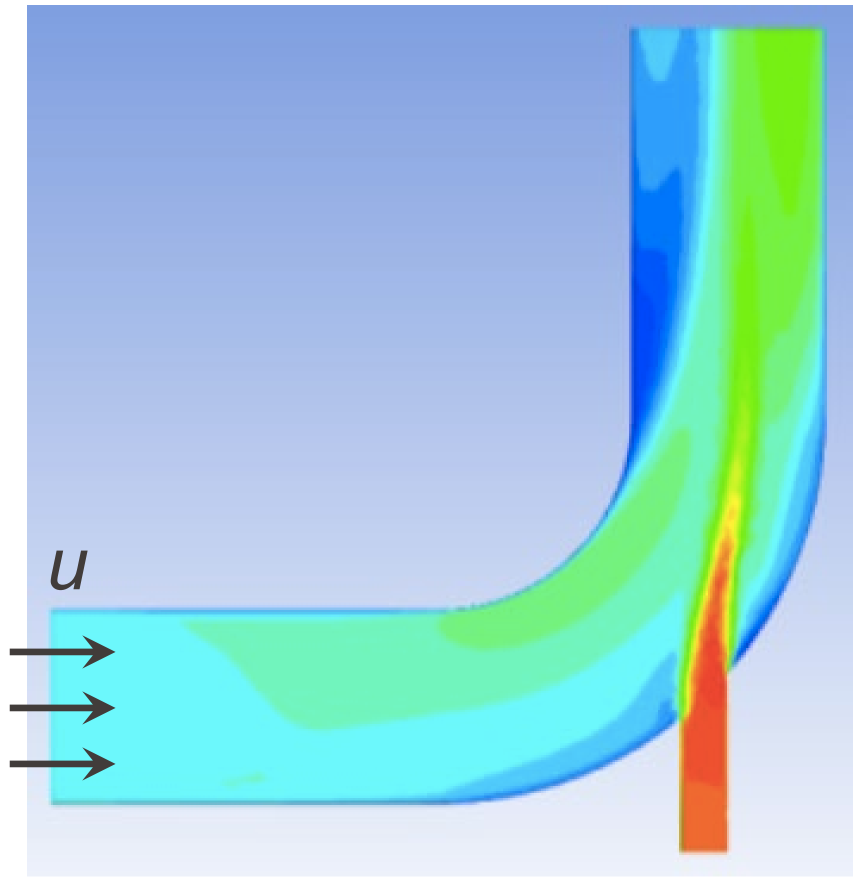
These BCs become better approximations farther away.

Outlet boundary:

Outflow BC ($\partial \mathbf{u}/\partial x = 0$) and pressure outlet BC are good approximations only far downstream.

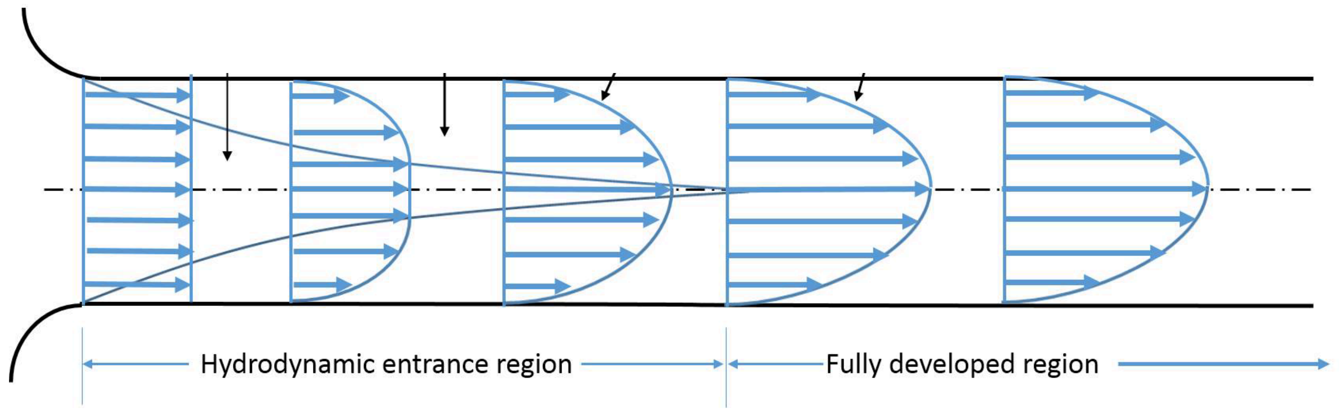
Boundary conditions

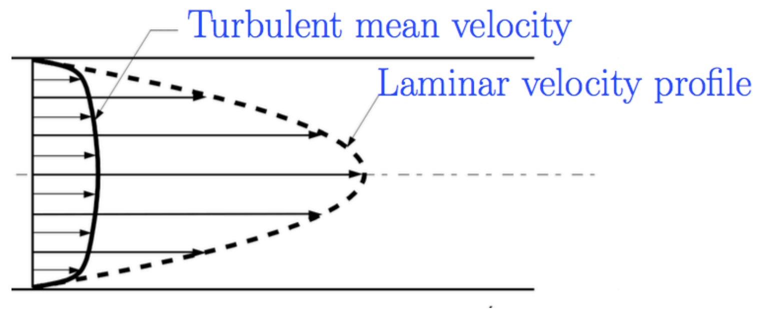
- Solution and convergence strongly depend on choice of BCs!
- BCs are often an approximation/simplification of the reality.
  - Example: internal flow

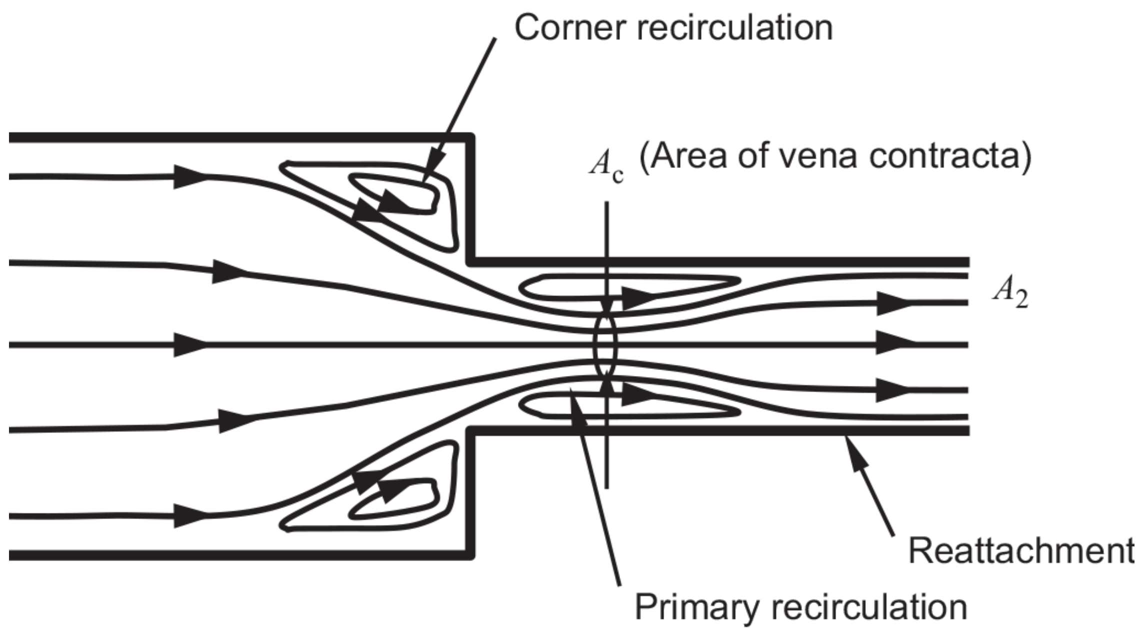


It is common to impose a uniform inlet velocity, but the actual profile may well be non-uniform.

Upstream pipe: non-uniform profile (developing/fully developed; laminar/turbulent).





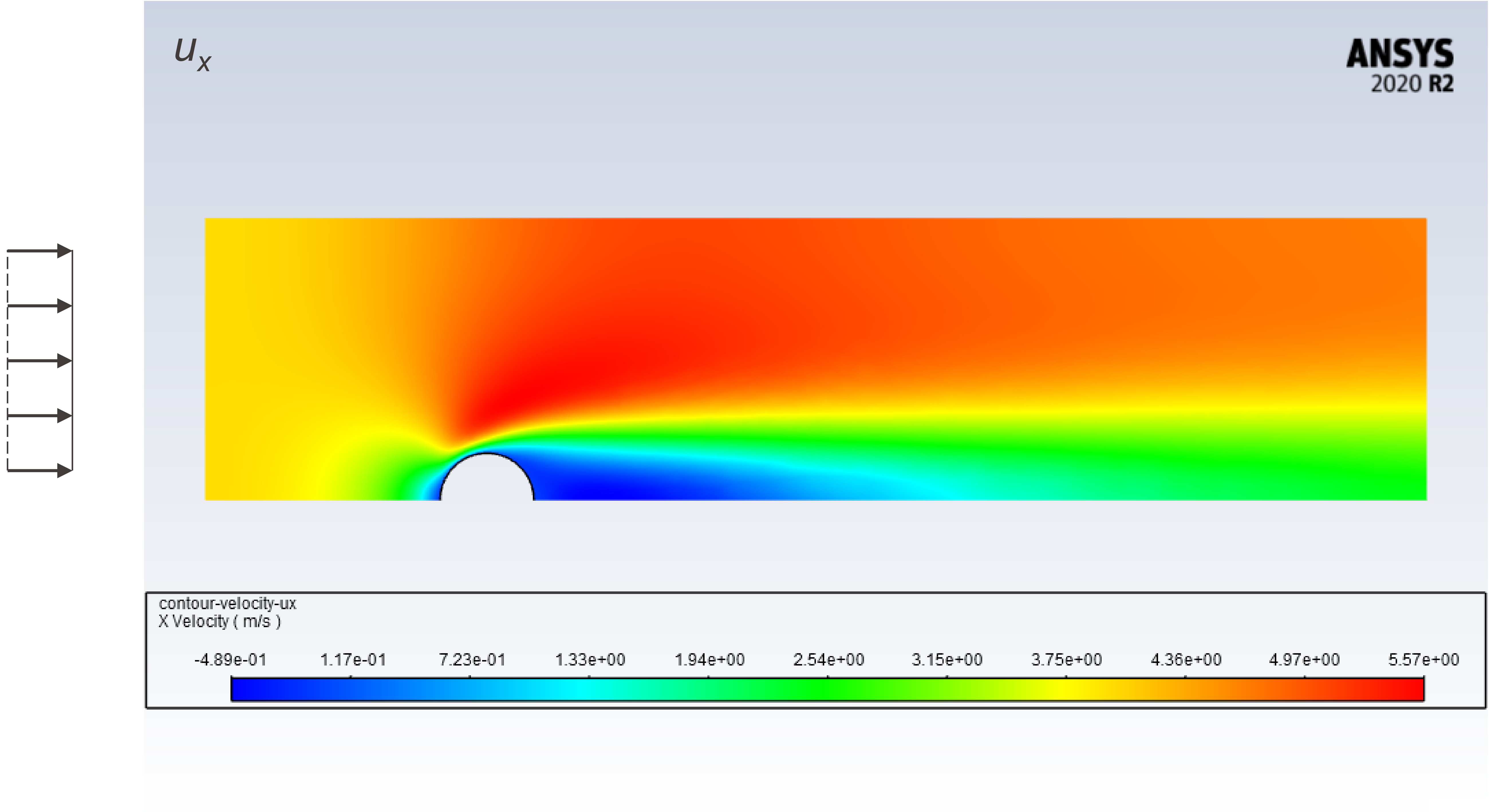


Sudden contraction: non-uniform profile (recirculation).

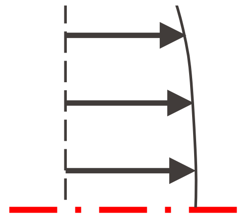
Smooth convergent nozzle: close to uniform near entrance.

Boundary conditions

- Symmetry



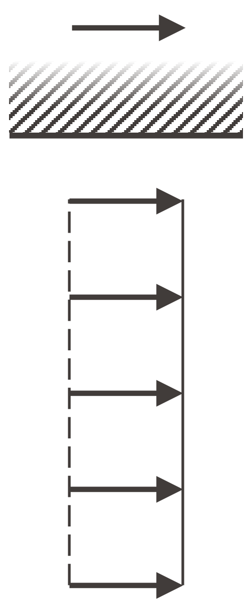
Note: a slip wall BC with zero shear stress is equivalent to a symmetry BC for velocity, but does not impose anything on other fields (e.g. temperature).

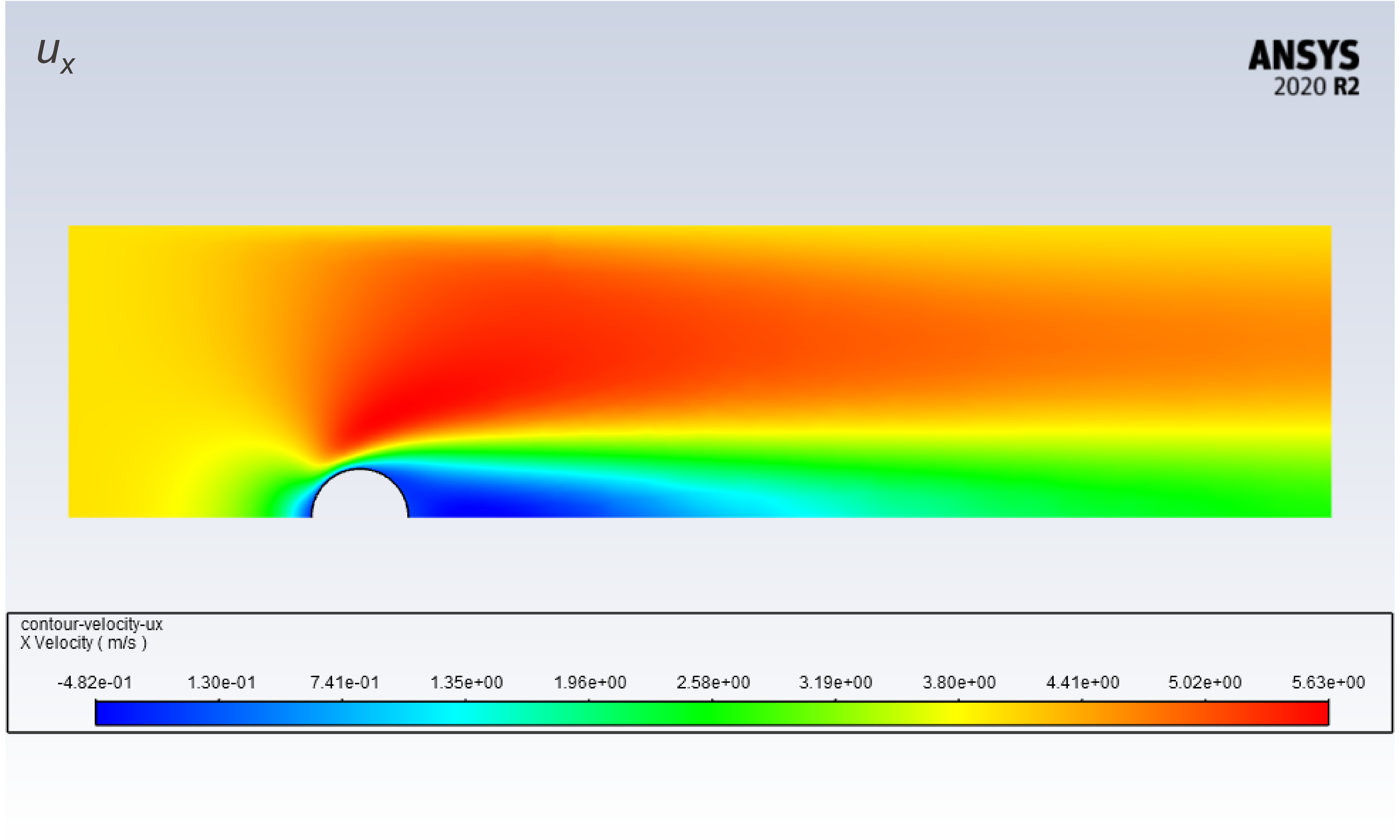


$$u_n = 0$$
$$\partial u_t / \partial n = 0$$

Boundary conditions

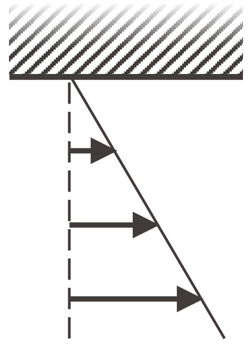
- Moving wall

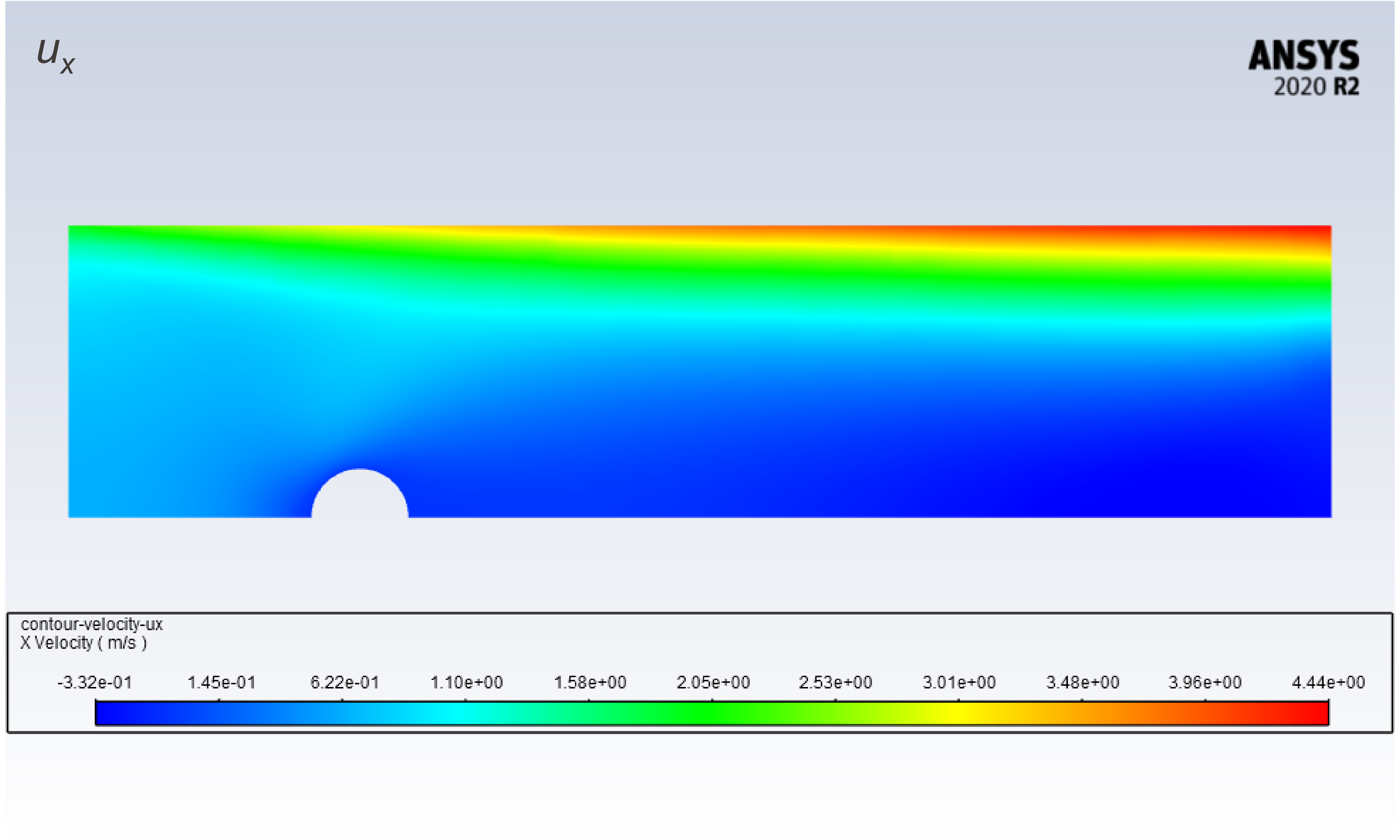




Boundary conditions

- Wall shear stress

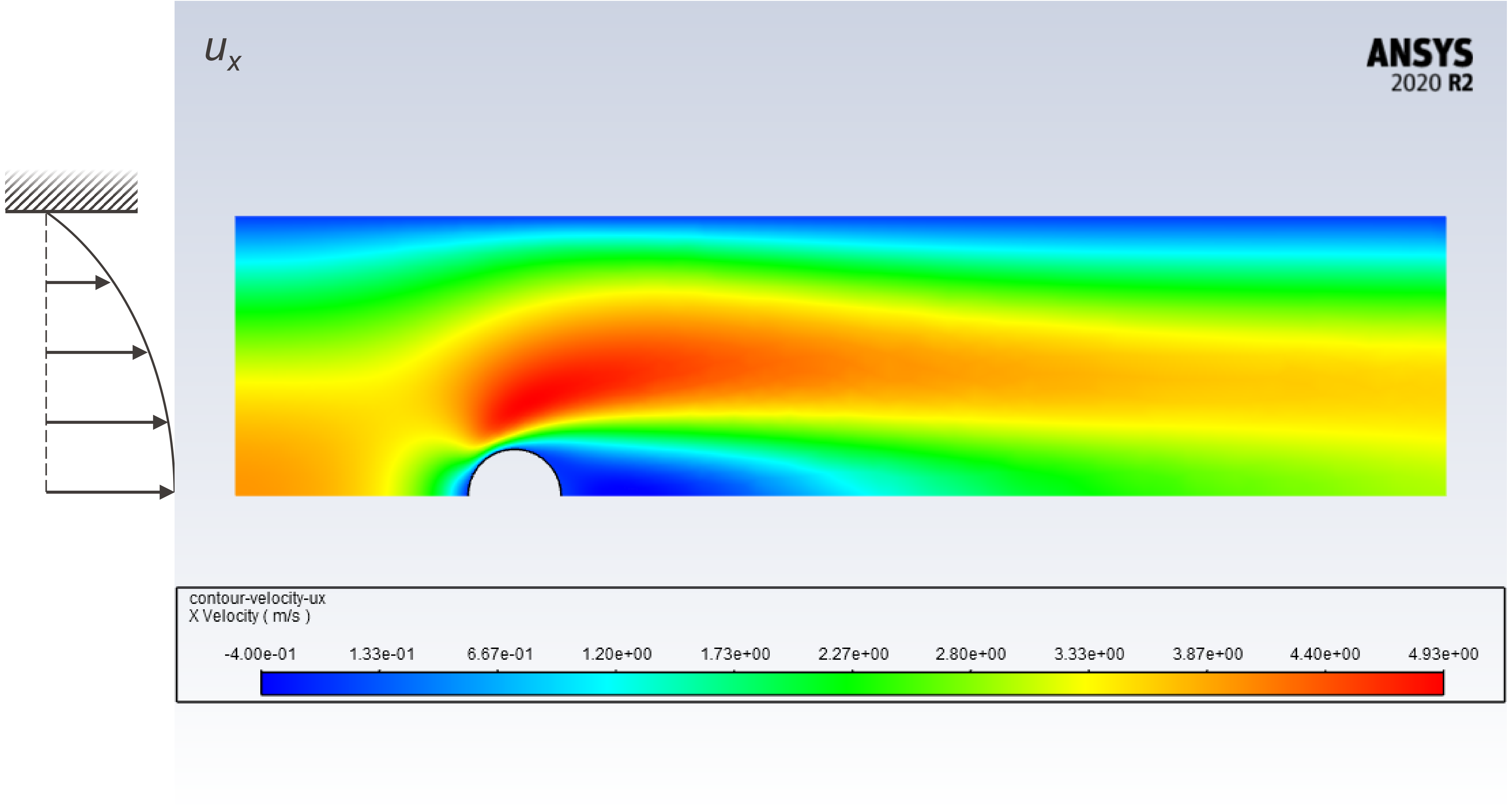




Boundary conditions

- Non-constant BCs that depend on space, on time, or on the solution.

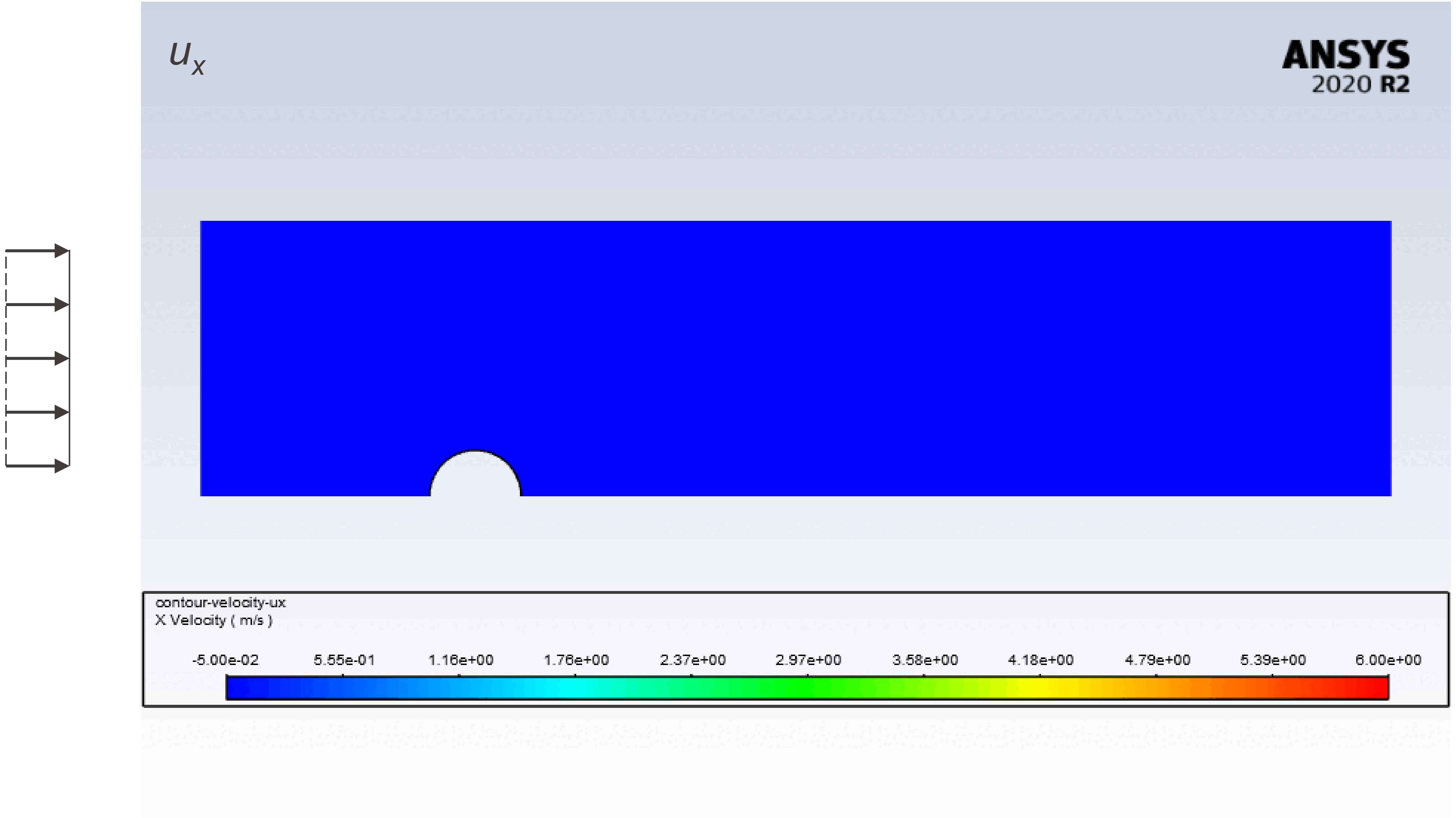
Parabolic inlet profile
u(y)



Boundary conditions

- Non-constant BCs that depend on space, on time, or on the solution.

Smoothly increasing inlet velocity *u(t)*

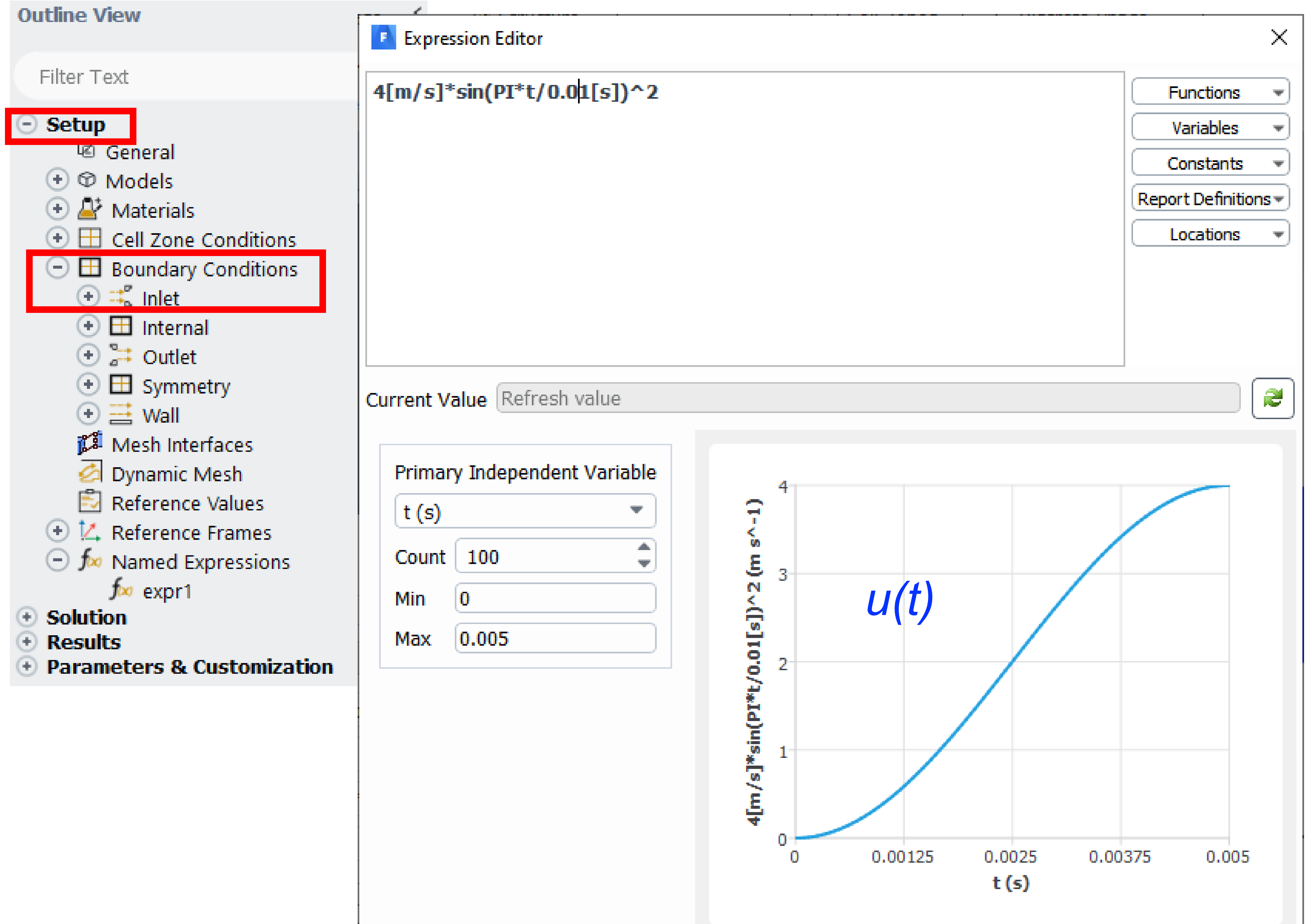


Quiz: is it equivalent to a cylinder smoothly accelerating in a fluid initially at rest?

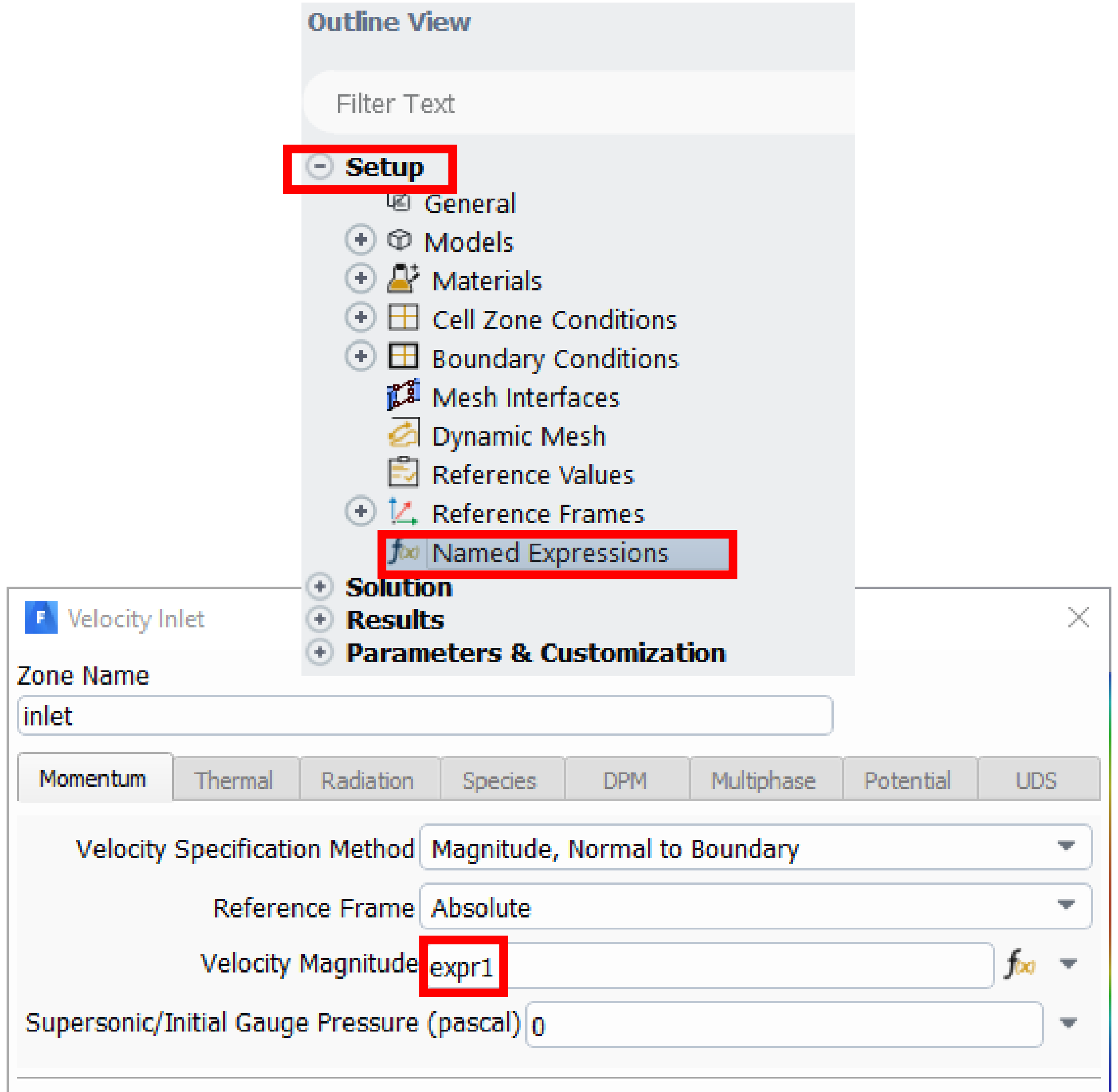
Boundary conditions

- Non-constant BCs: “**expressions**”. (When an analytical expression is available.)

Define expression directly in the BC:



Or first define a “named expression”, then apply it in the BC:

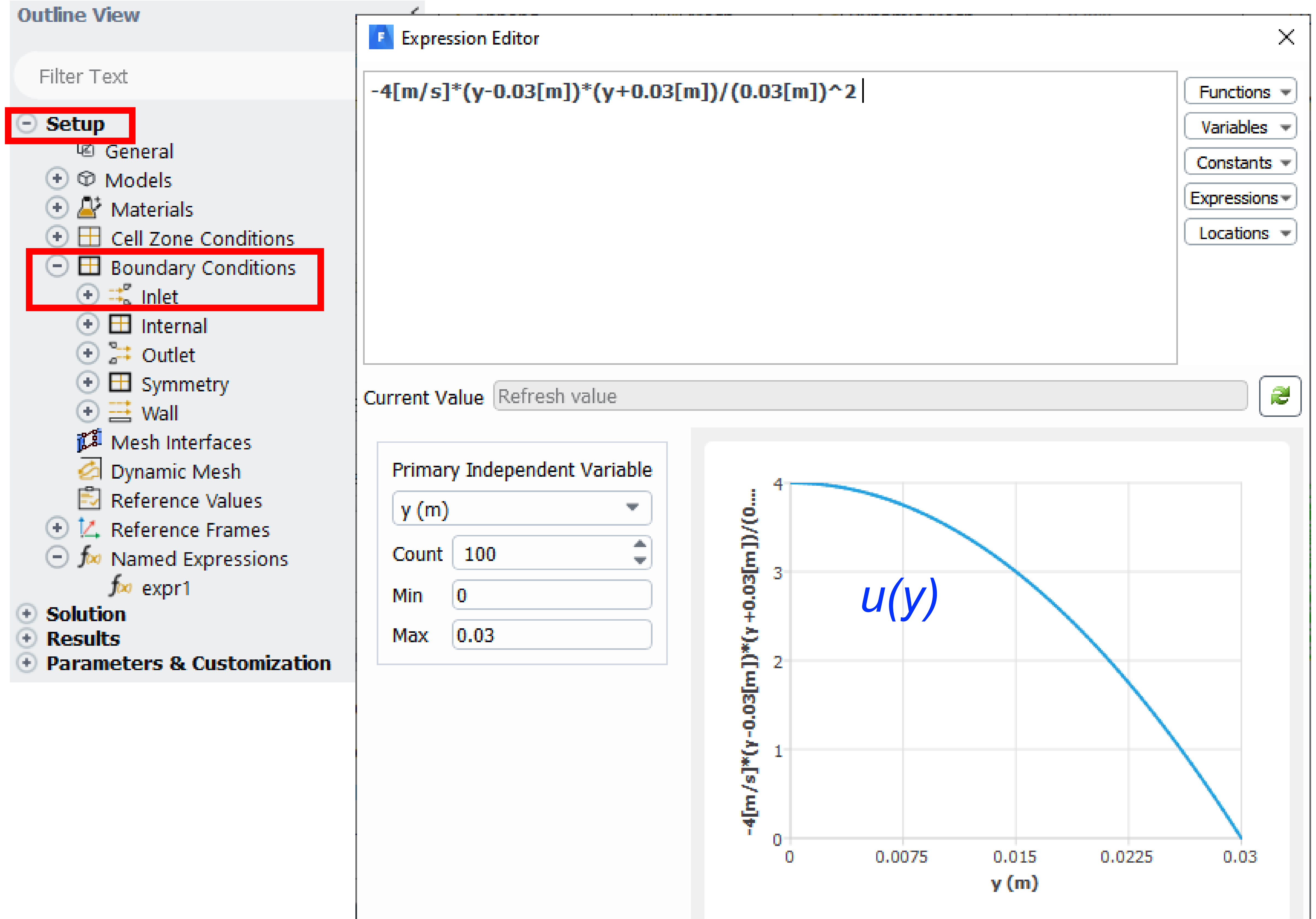


Numerical Flow Simulation

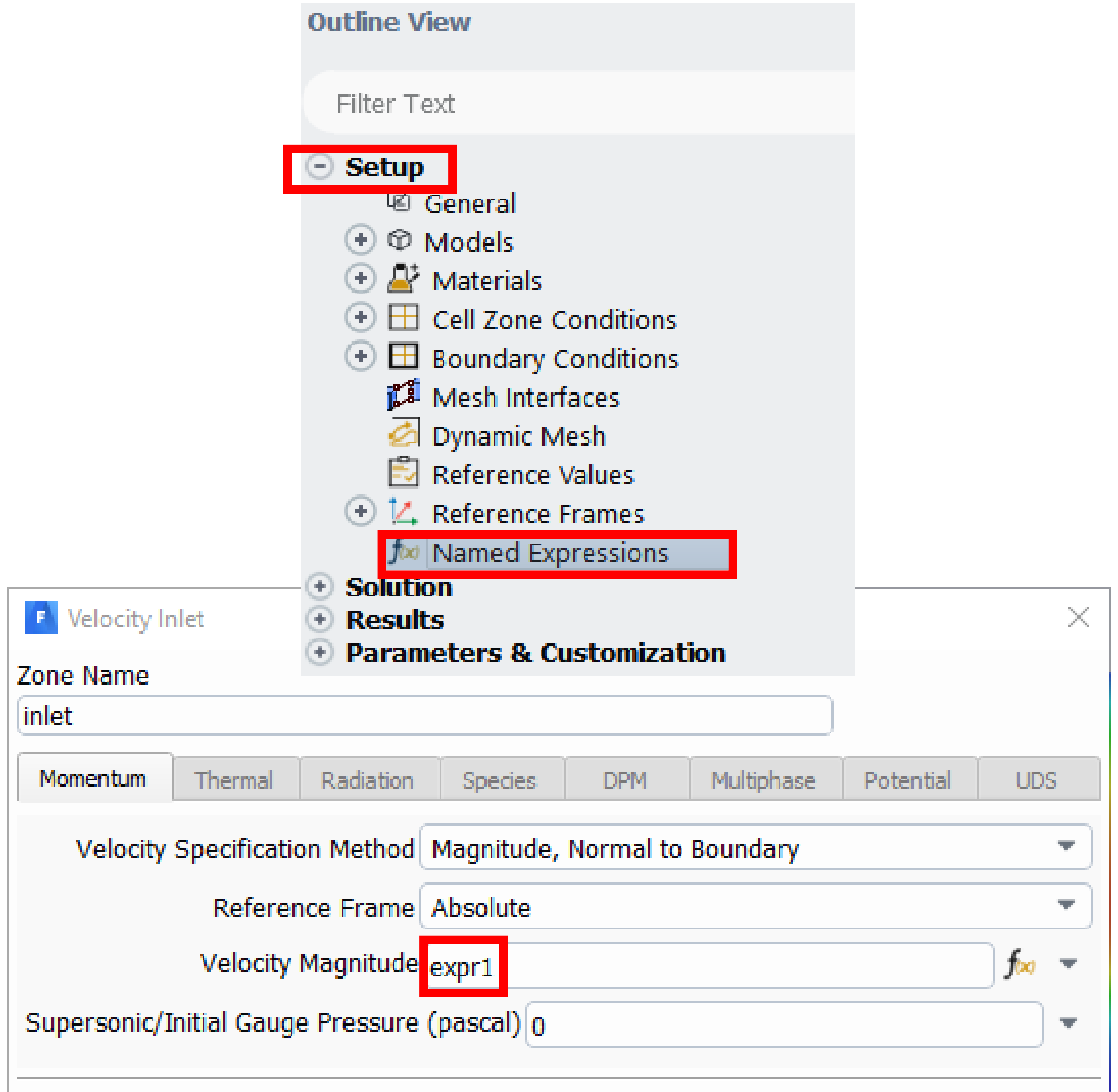
Boundary conditions

- Non-constant BCs: “**expressions**”. (When an analytical expression is available.)

Define expression directly in the BC:



Or first define a “named expression”, then apply it in the BC:

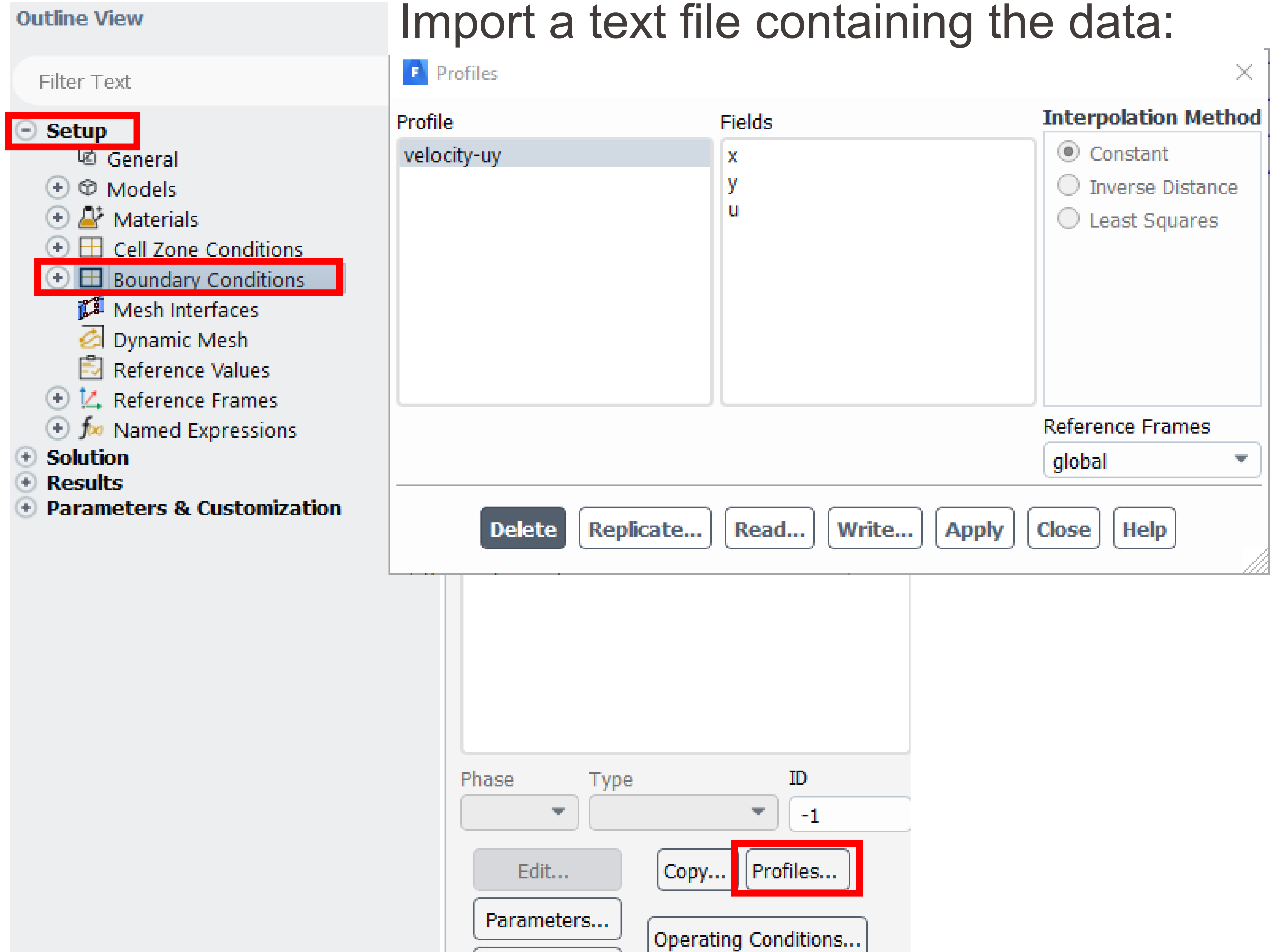


Numerical Flow Simulation

Boundary conditions

- Non-constant BCs: “**profiles**”. (To use data measured in an experiment, calculated by an external program, or written from a previous Fluent solution.)

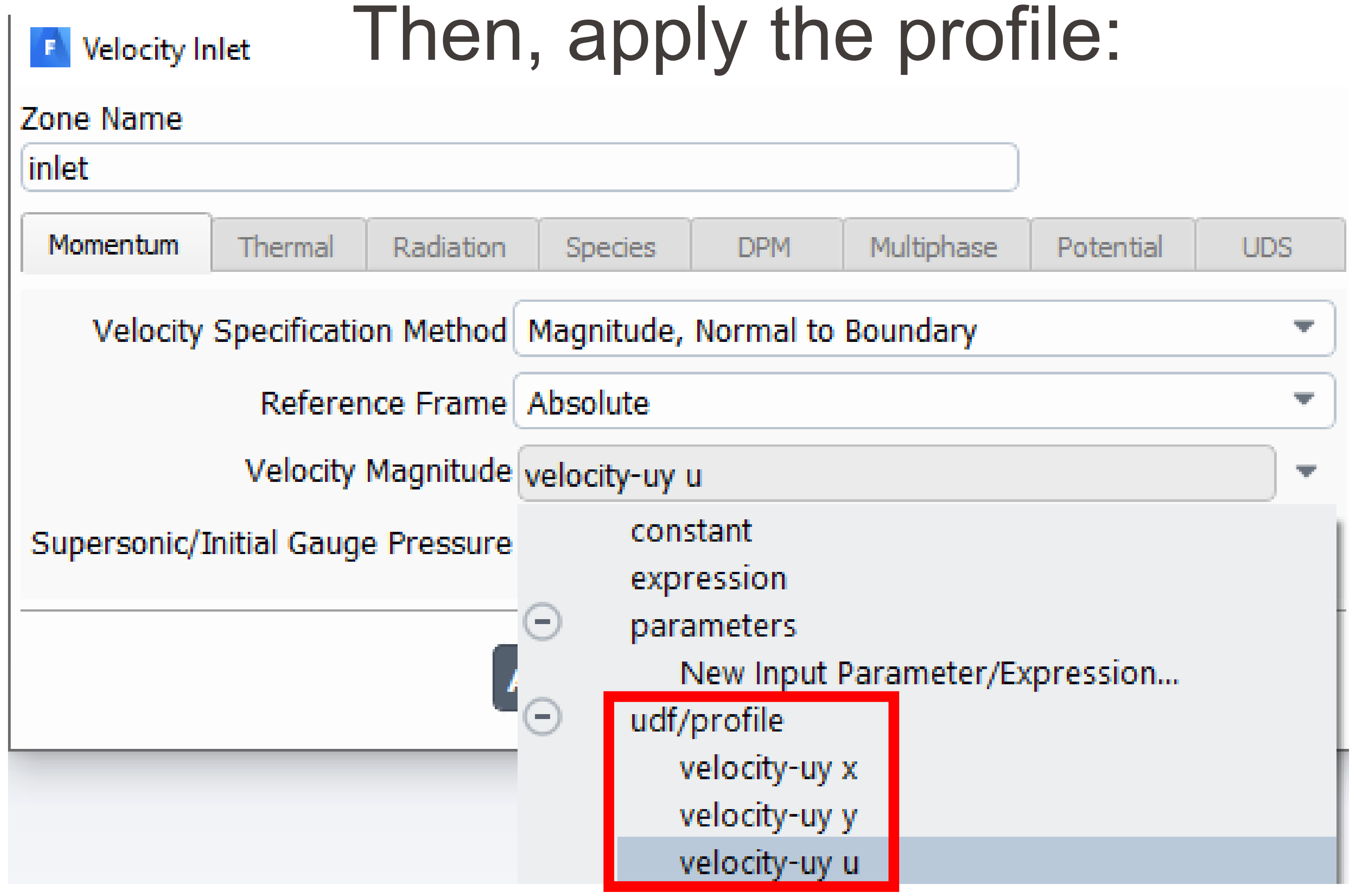
Import a text file containing the data:



For example (noisy parabolic profile):

```
((velocity-uy line 21)
(x
0        0        0        0        0        0        0        0        0        0
0        0        0        0        0        0        0        0        0        0
0 )
(y
0 0.0015  0.003 0.0045  0.006 0.0075  0.009 0.0105  0.012 0.0135
0.015 0.0165  0.018 0.0195  0.021 0.0225  0.024 0.0255  0.027
0.0285  0.03 )
(u
4.50355789823 4.63926709932 3.36317890607 4.57140136982
4.05177479396 3.106064648 3.28559715019 3.58501043073
4.09201093669 3.93382165632 2.45218093068 3.54294845082
3.29146711719 2.28660103796 2.52044875022 1.1770181418
1.3148180522 1.7751768403 1.2275317273 1.12518788223
0.249185118651 )
)
```

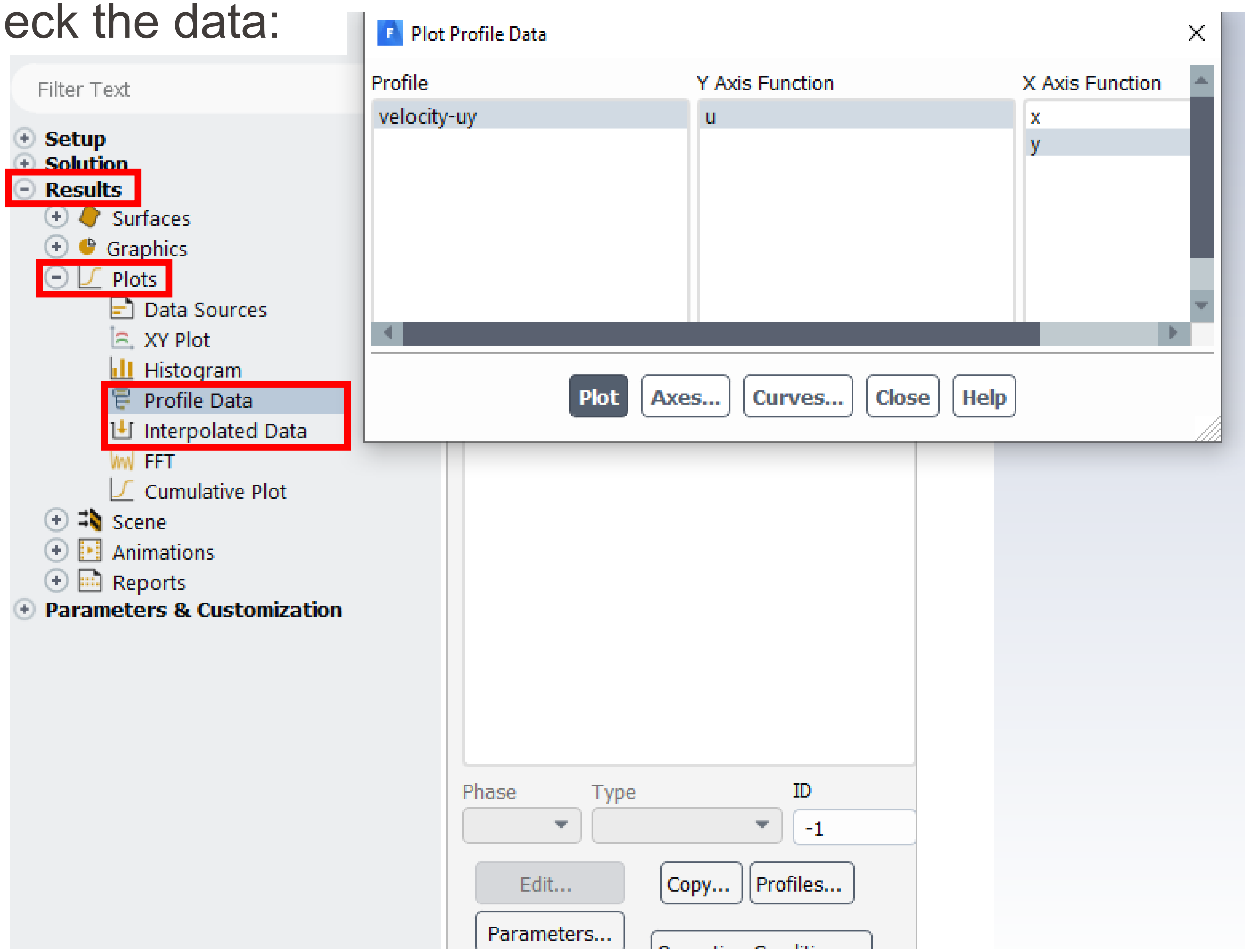
Then, apply the profile:



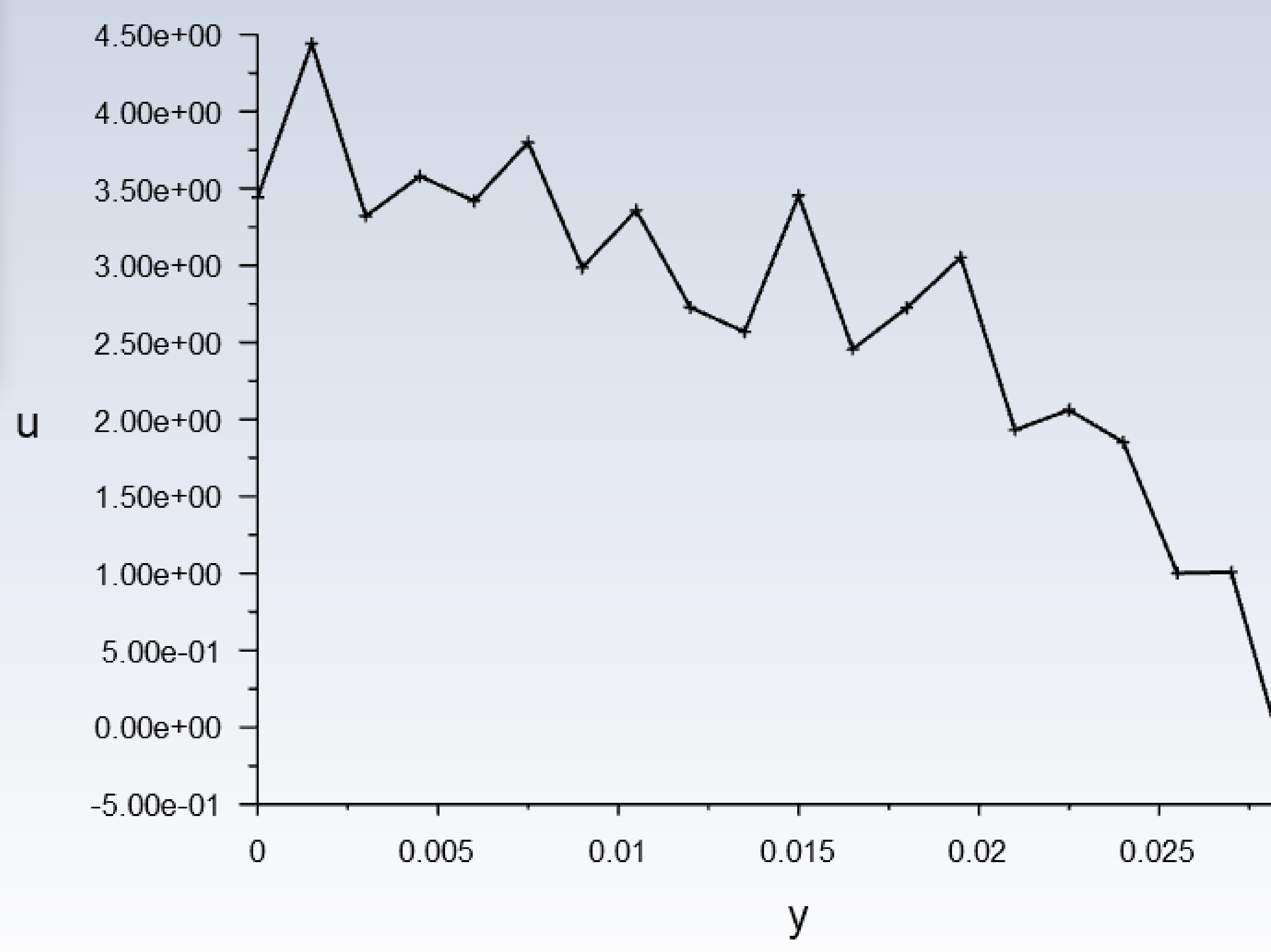
Boundary conditions

- Non-constant BCs: "**profiles**". (To use data measured in an experiment, calculated by an external program, or written from a previous Fluent solution.)

Check the data:



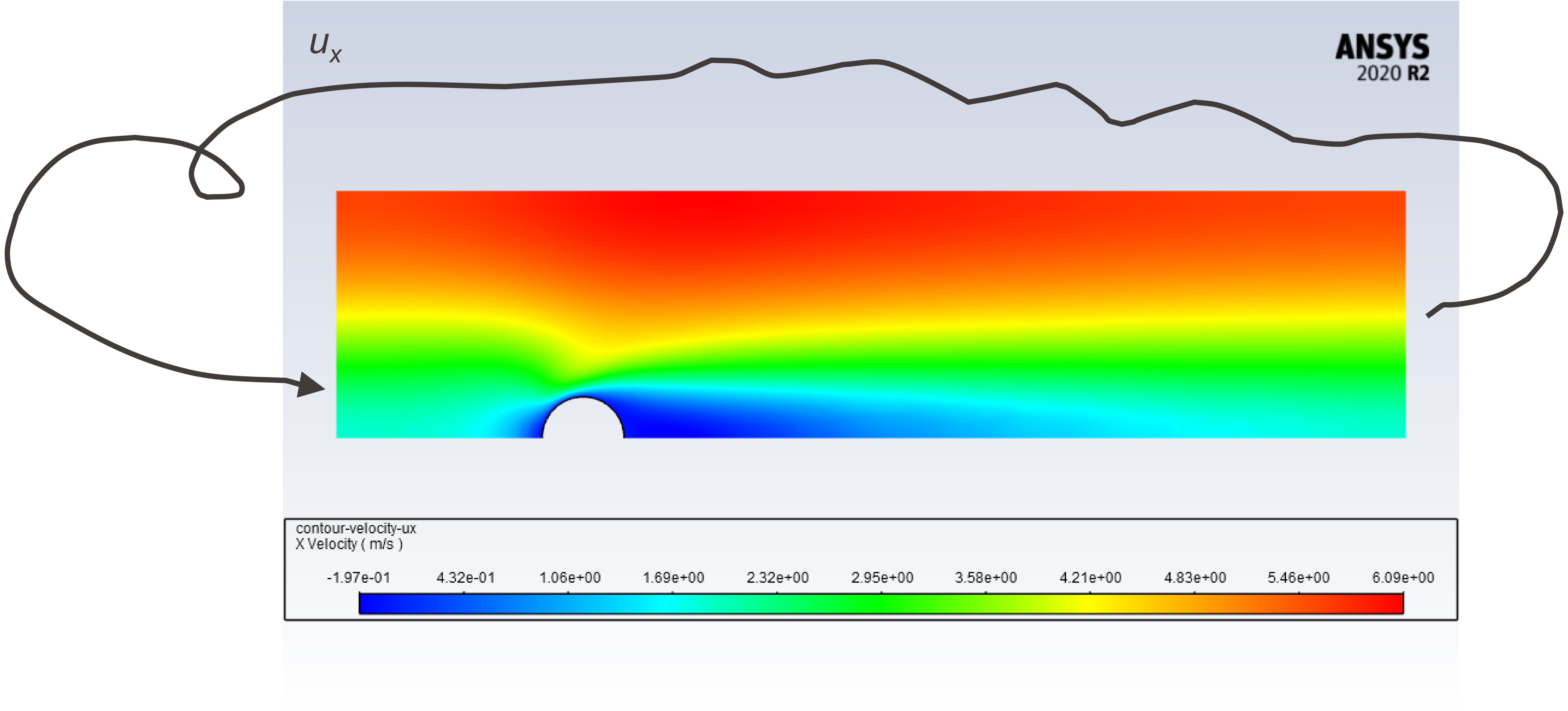
For example (noisy parabolic profile):



Numerical Flow Simulation

Boundary conditions

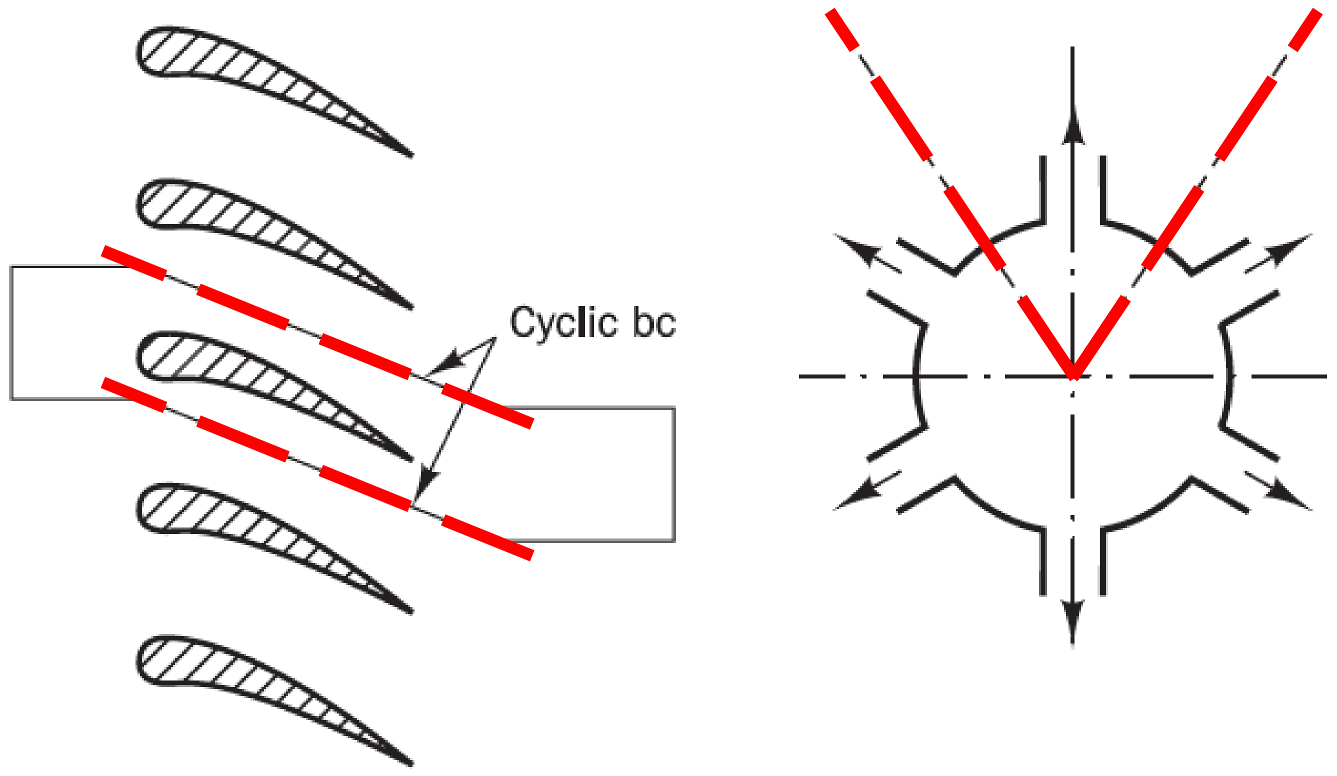
- Periodic BCs

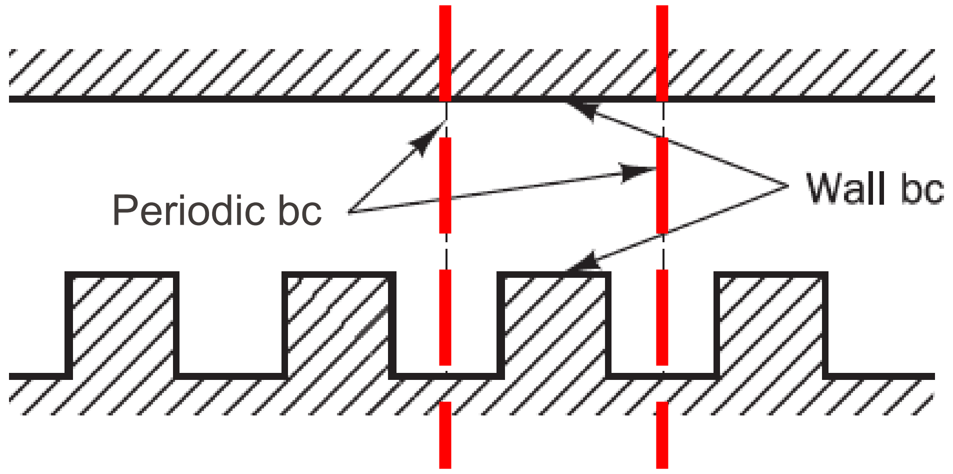


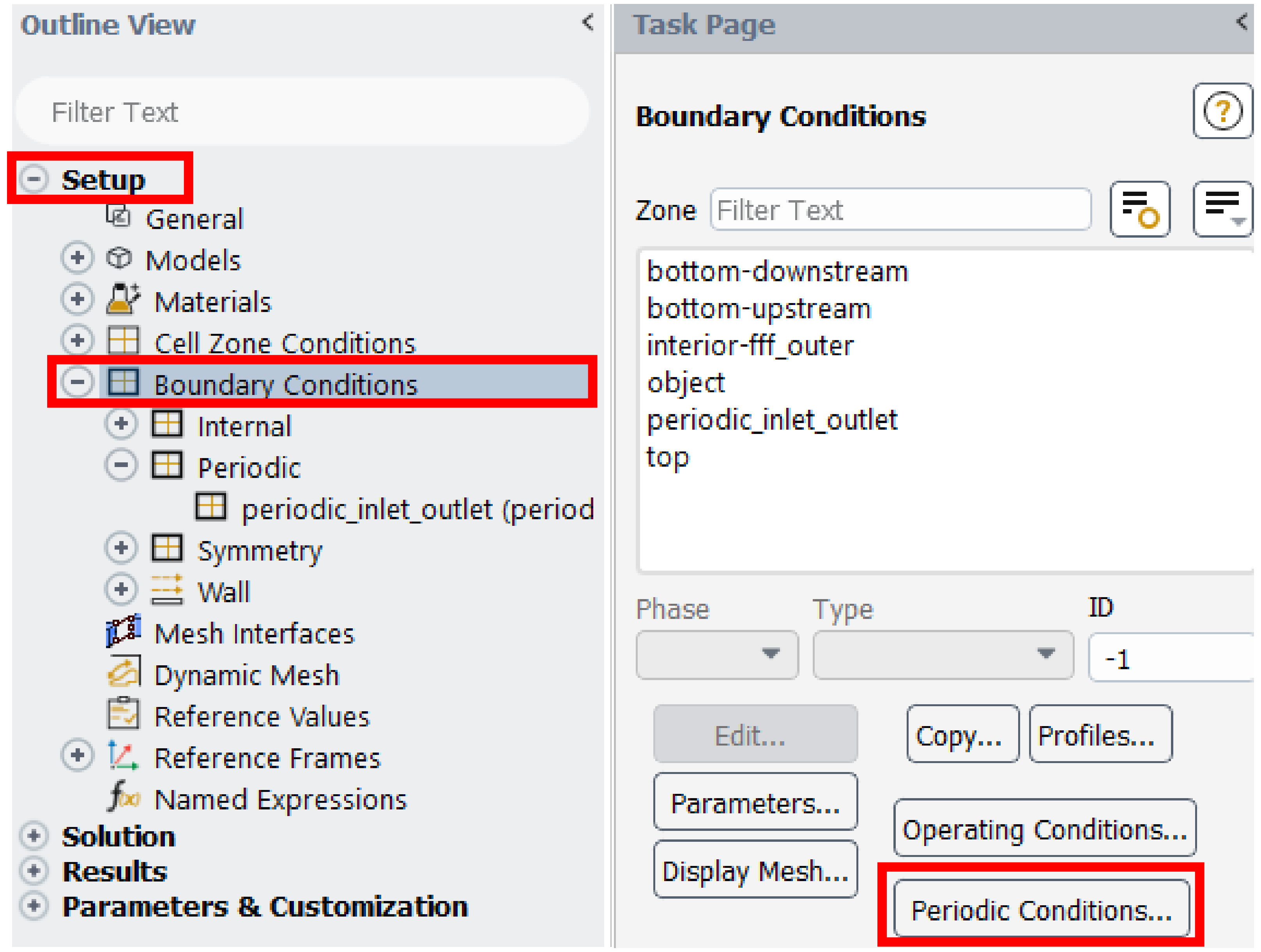
Boundary conditions

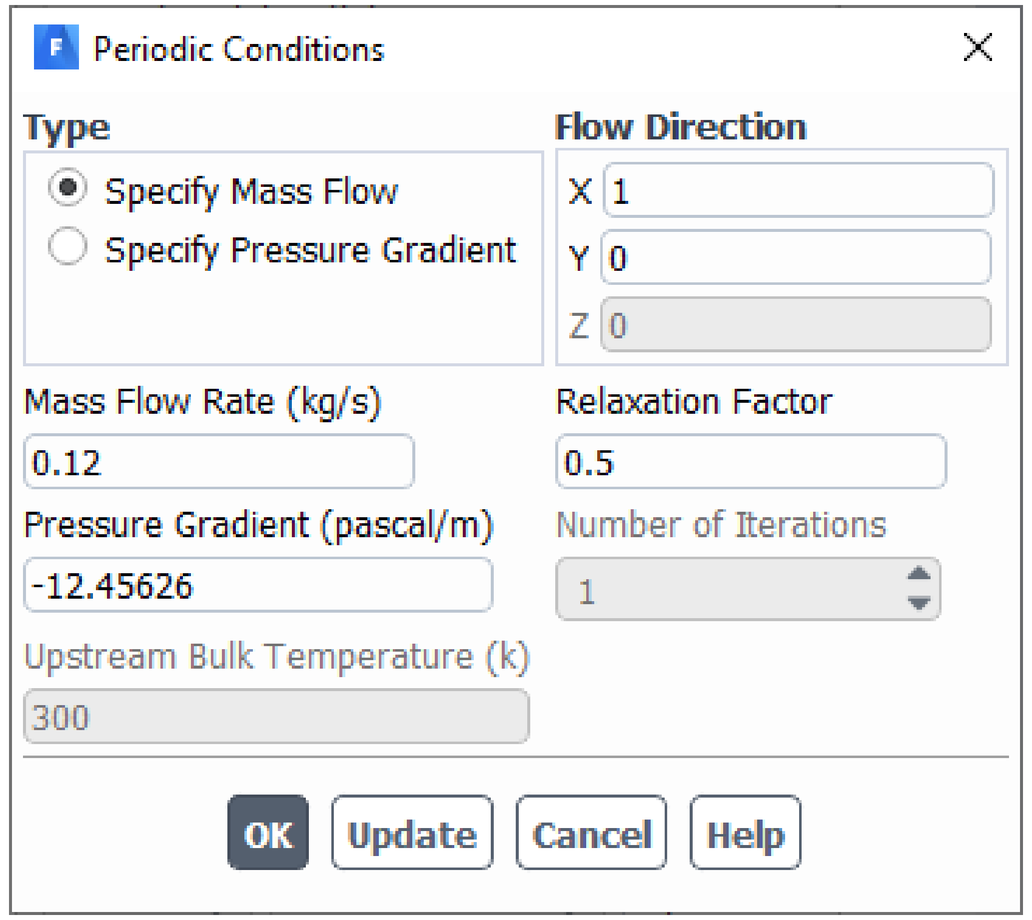
- Periodic BCs (translation or rotation)
  1. Without pressure drop (“cyclic”)

  2. With pressure drop (“fully developed”)
     → specify mass flow rate or pressure gradient





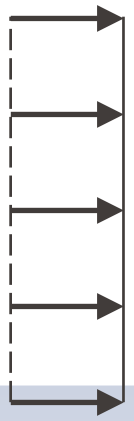


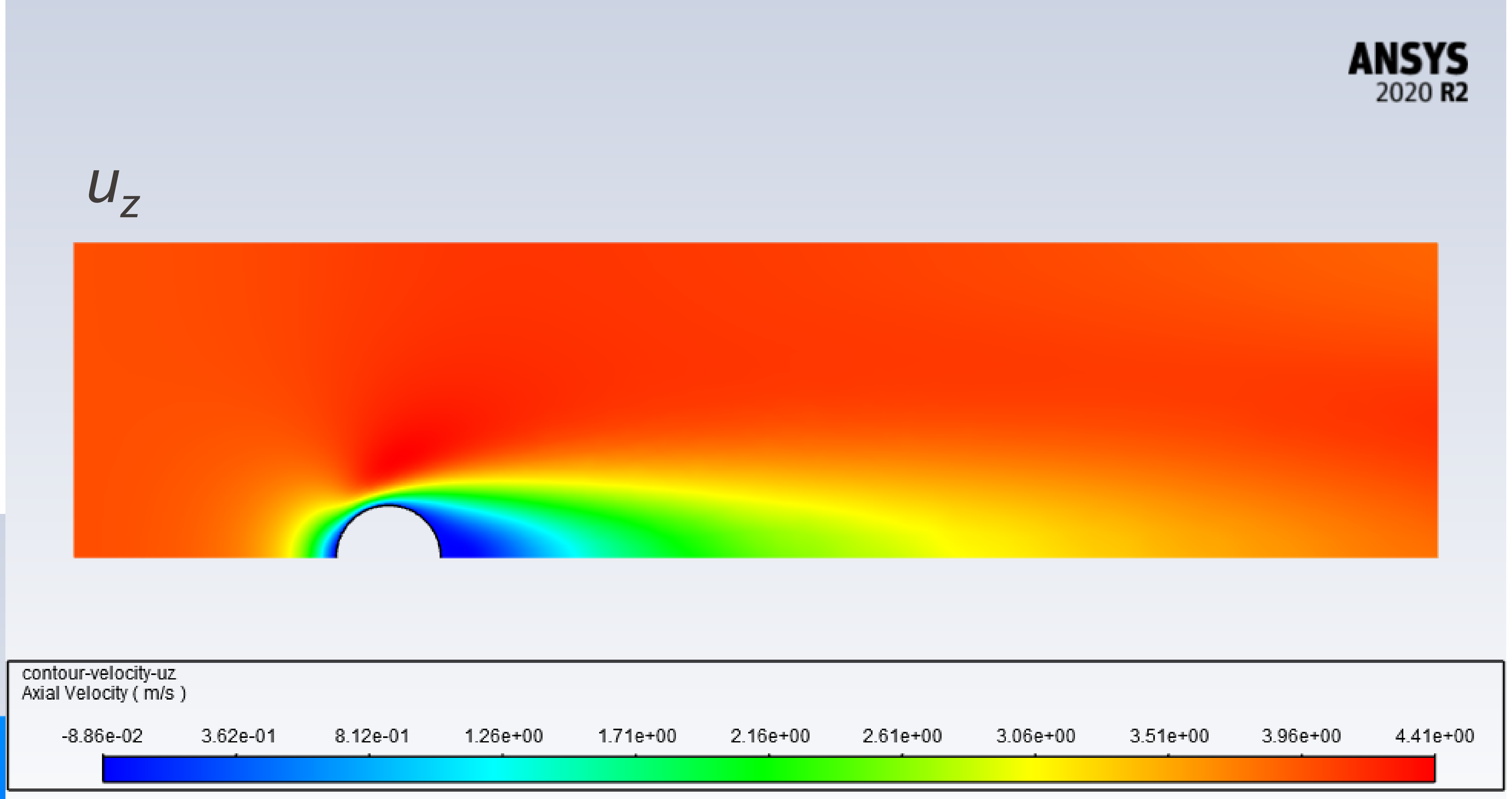


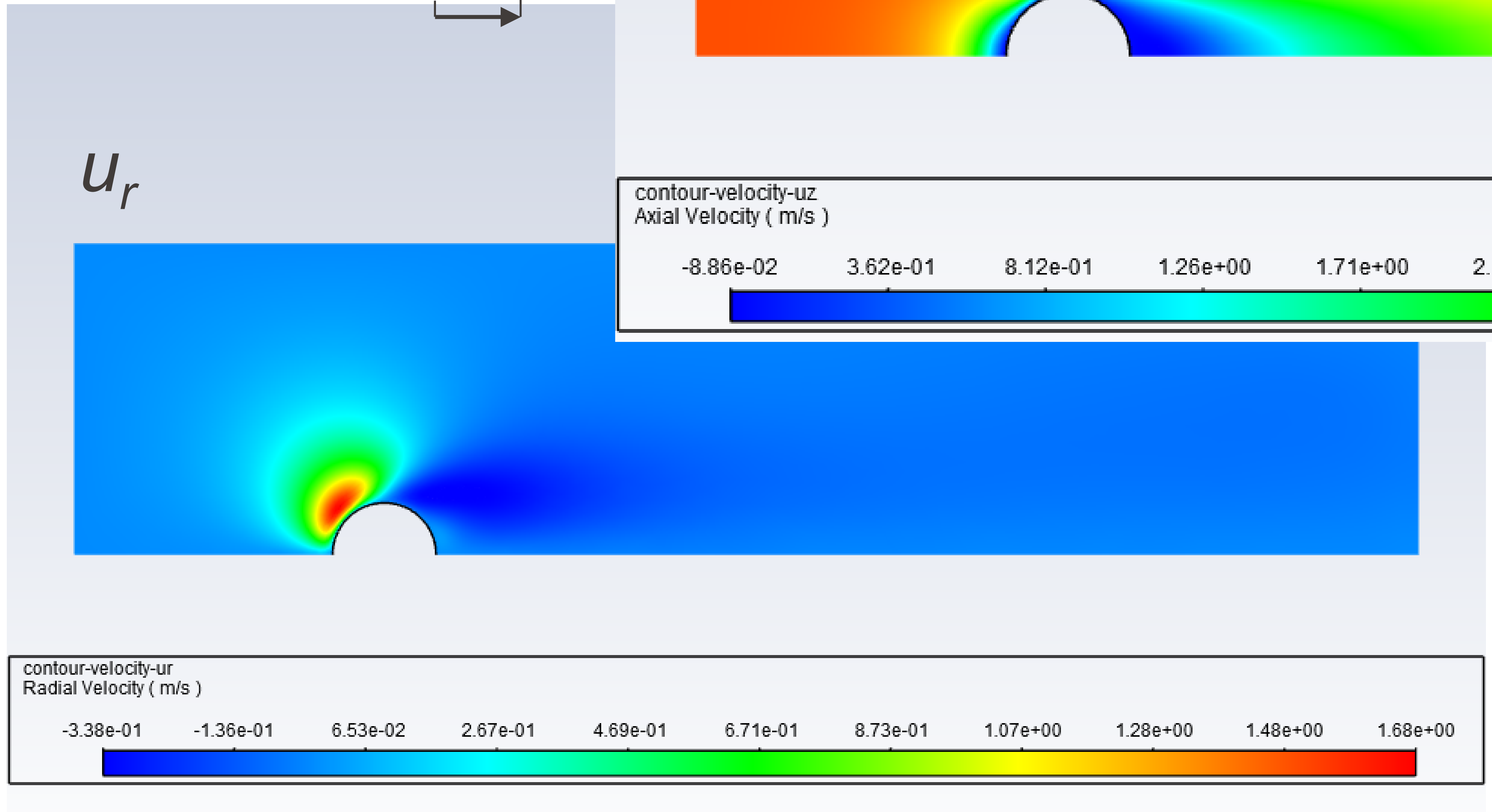
Boundary conditions

- Axis (for axisymmetric flows)

Without swirl:





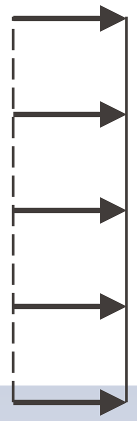


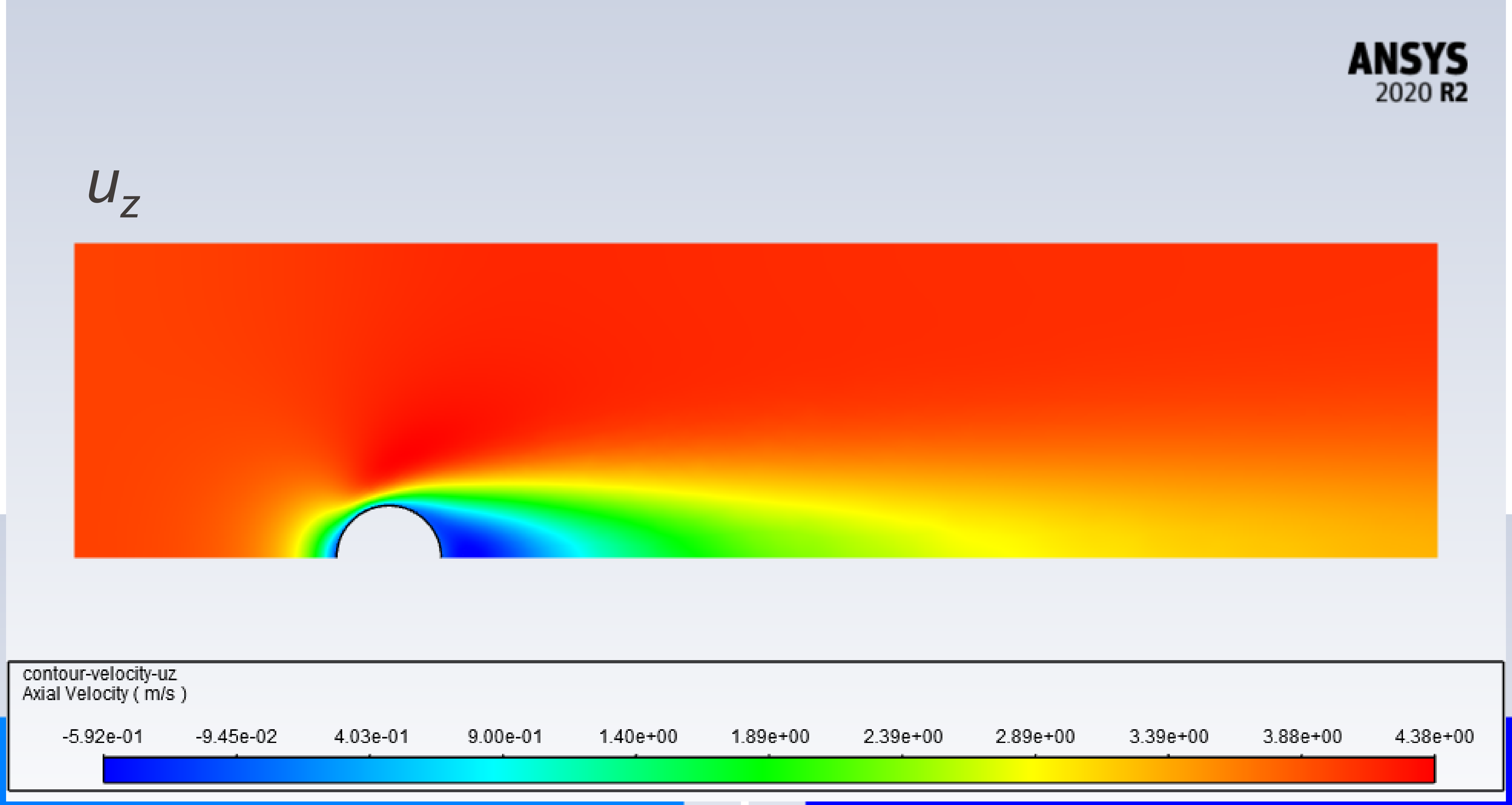
$u_\theta = 0$

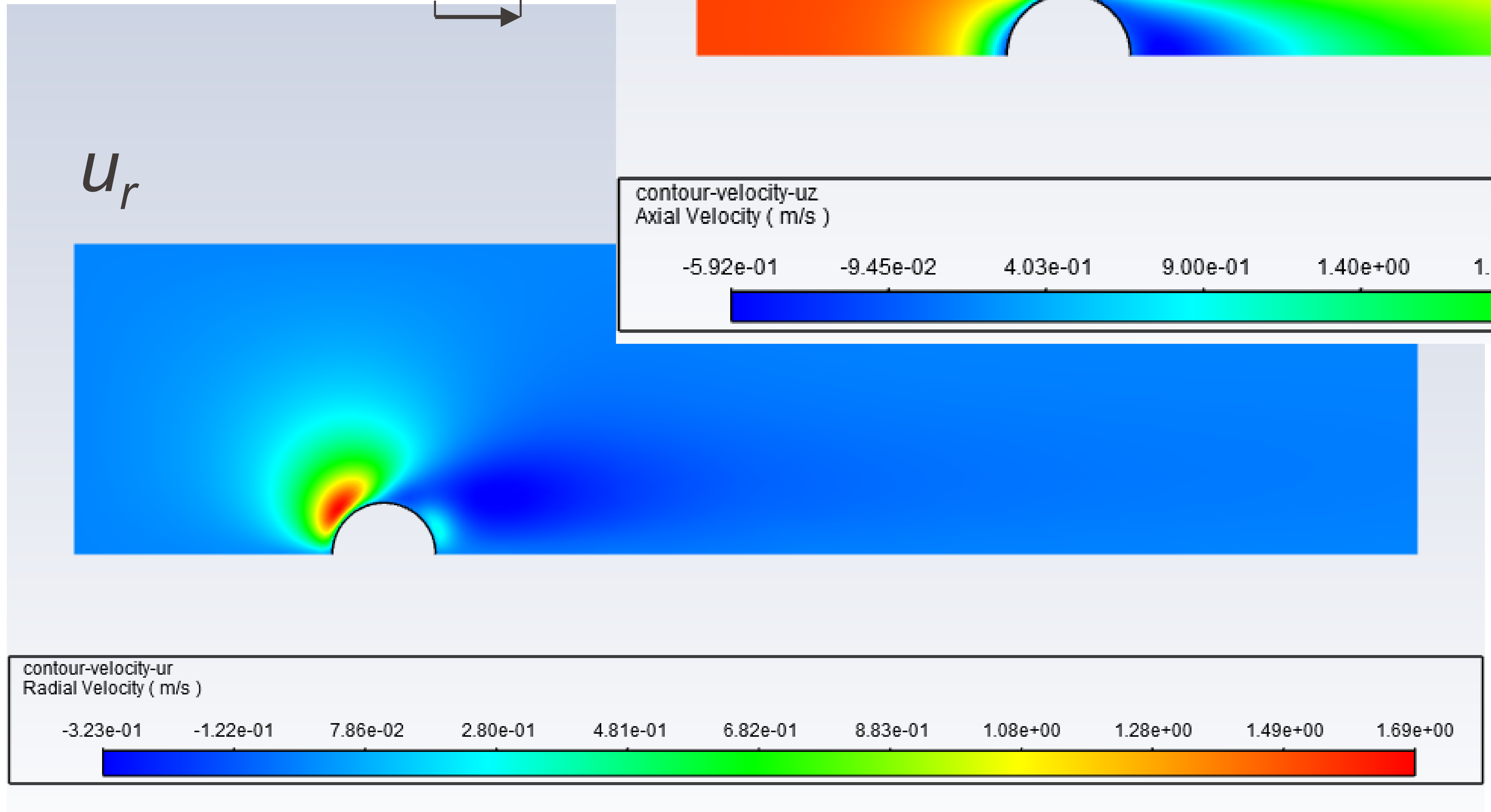
Boundary conditions

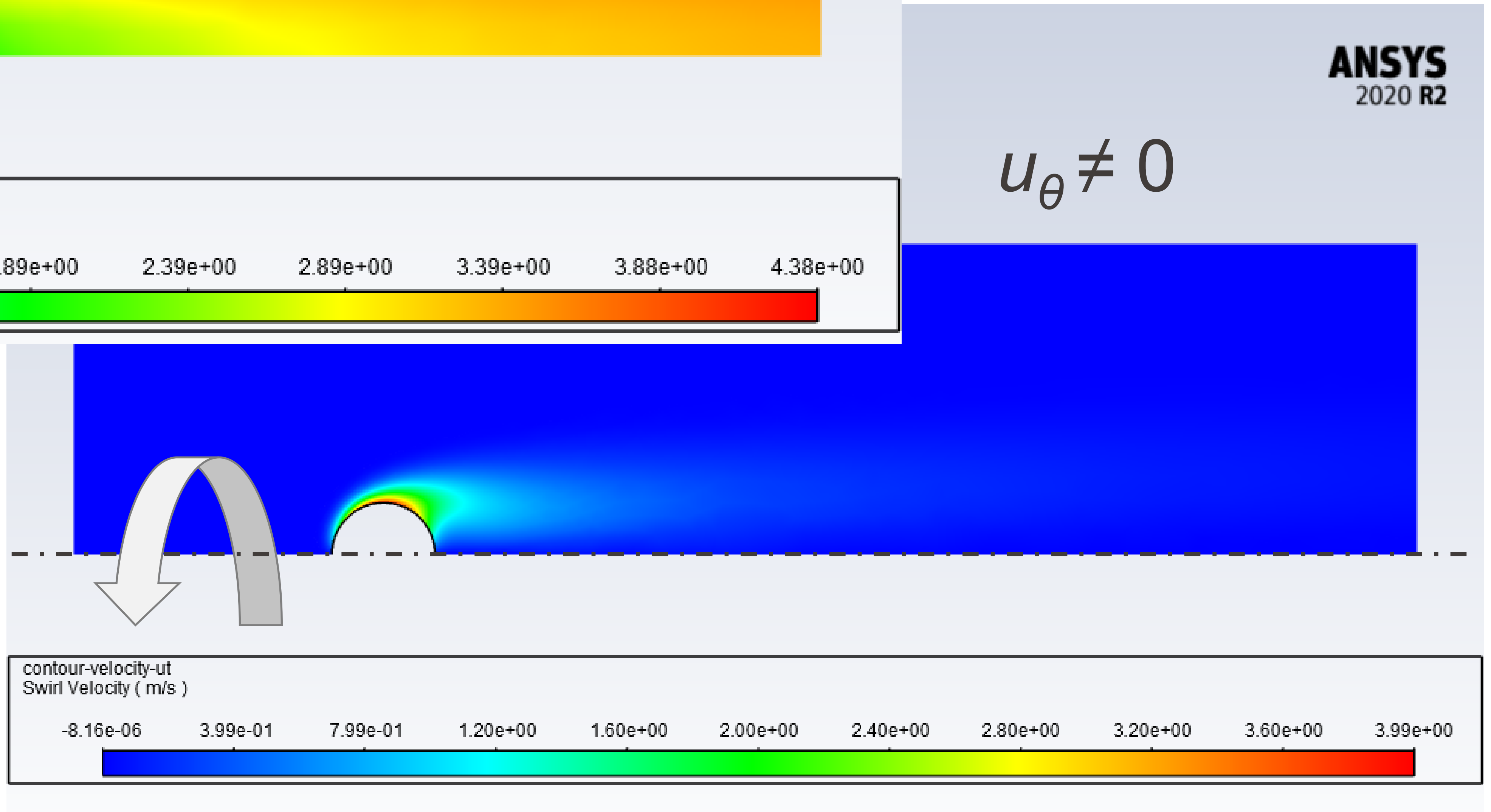
- Axis (for axisymmetric flows)

With swirl (here as a wall BC):



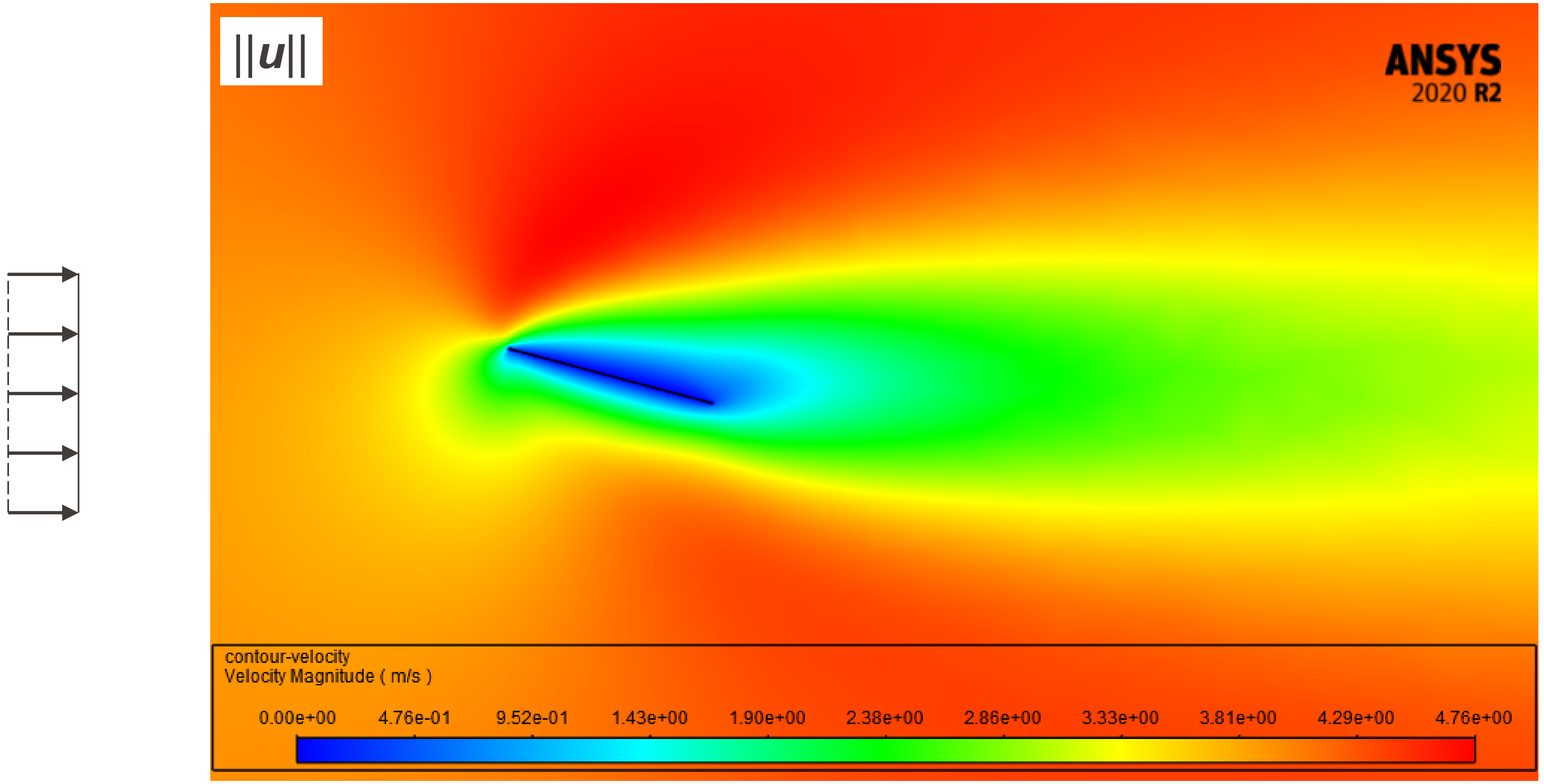






Boundary conditions

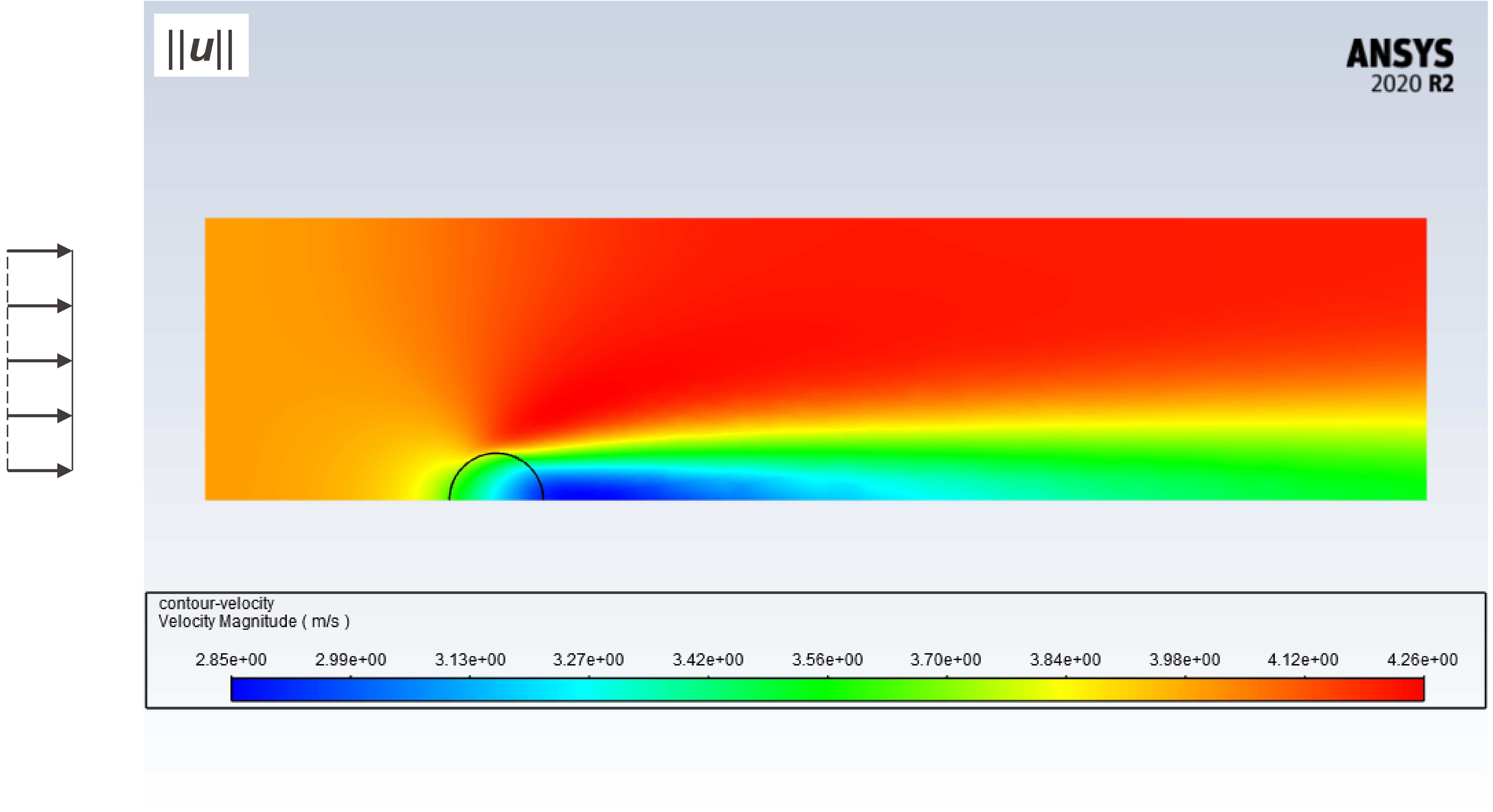
- Two-sided wall (zero-thickness wall)



Numerical Flow Simulation

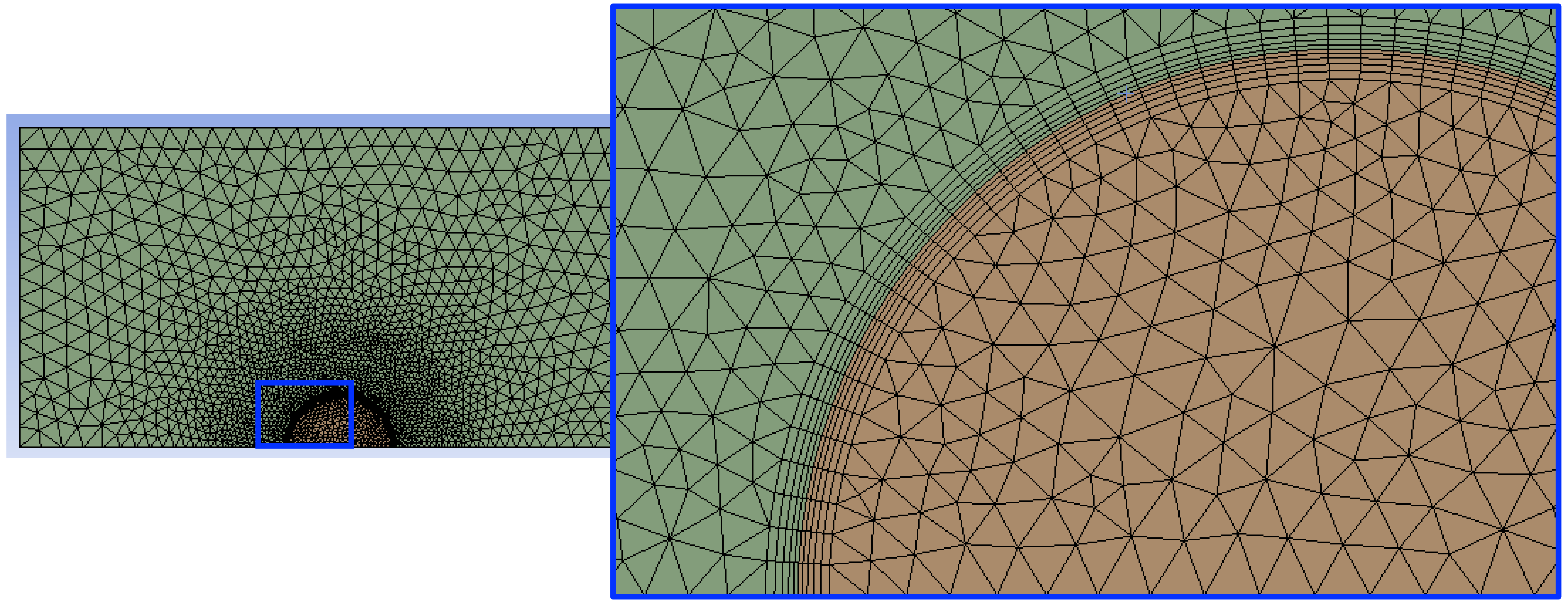
Zone conditions

- Porous medium ("porous zone")

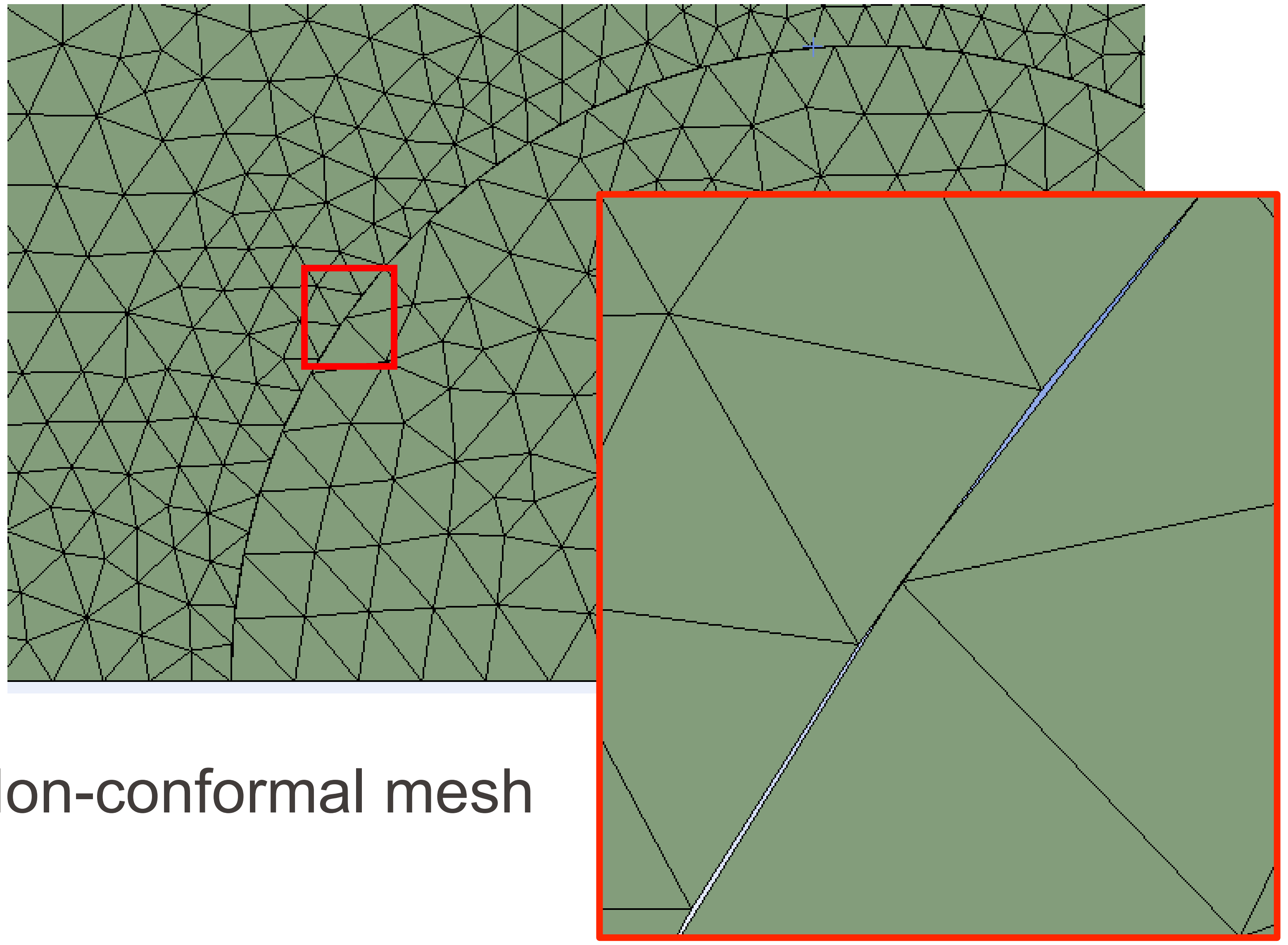


Zone conditions

- Note: if several zones, must create the geometry (and mesh) with care. Need either a **single body** with **several faces and “named selections”** (otherwise, get a single zone), or **several bodies** with a **“shared topology”** (otherwise, get a non-conformal mesh).



Conformal mesh, with 2 distinct zones



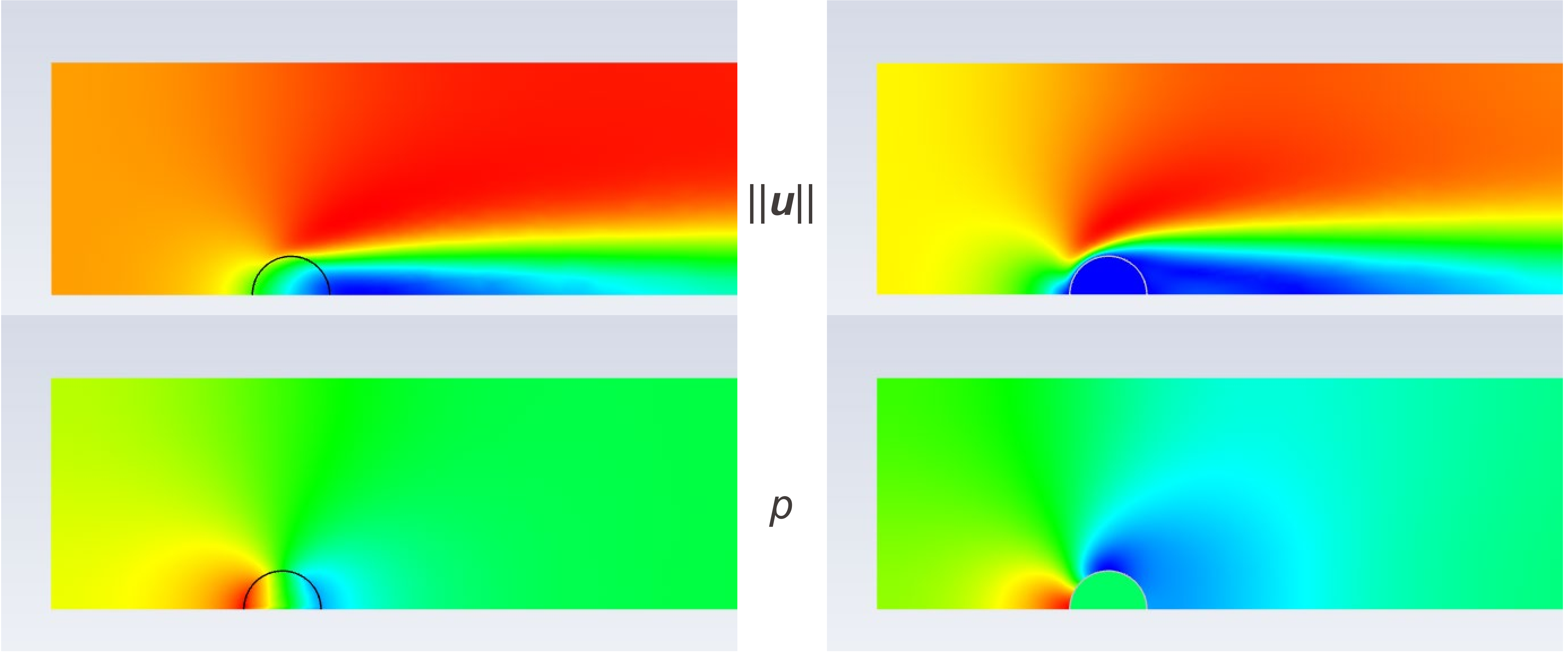
Non-conformal mesh

Zone conditions

- Note: if several zones, must create the geometry (and mesh) with care. Need either a **single body** with **several faces and “named selections”** (otherwise, get a single zone), or **several bodies** with a **“shared topology”** (otherwise, get a non-conformal mesh).

Example: flow though porous medium

Conformal mesh: OK



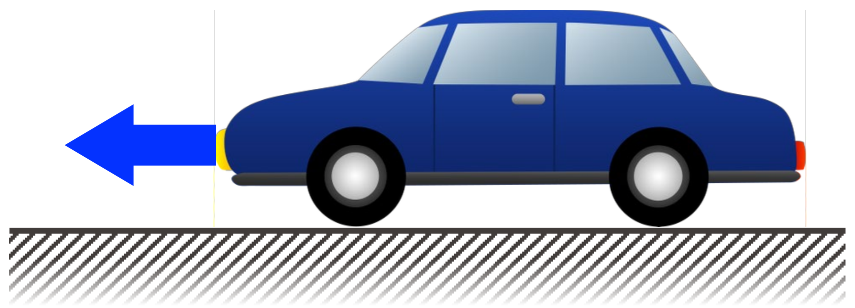
Non-conformal mesh: outer flow not “seen” by inner domain

(Tedious and less accurate alternative: set up a non-conformal mesh anyway in Fluent.)

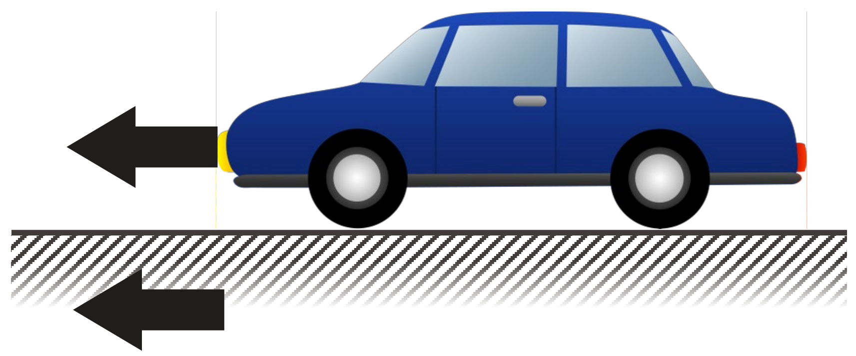
Unsteady problems with moving boundaries

- **Inertial reference frame** (uniform rectilinear motion) → **stationary** problem in the moving frame.

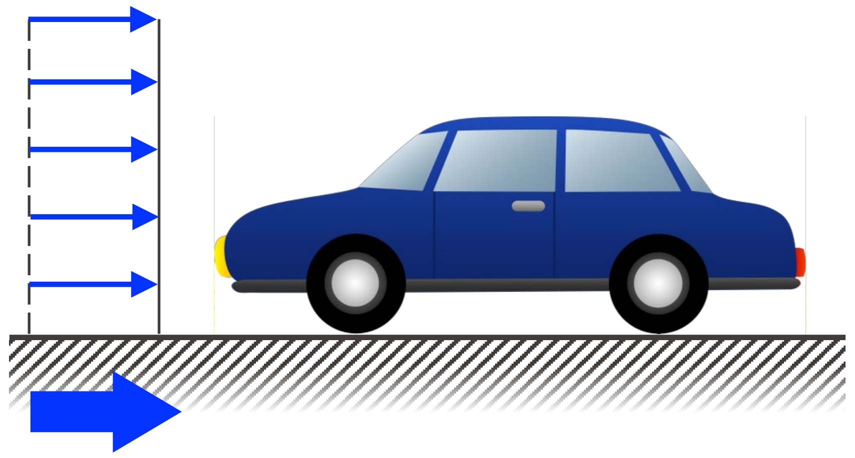
Moving car, still air, fixed road



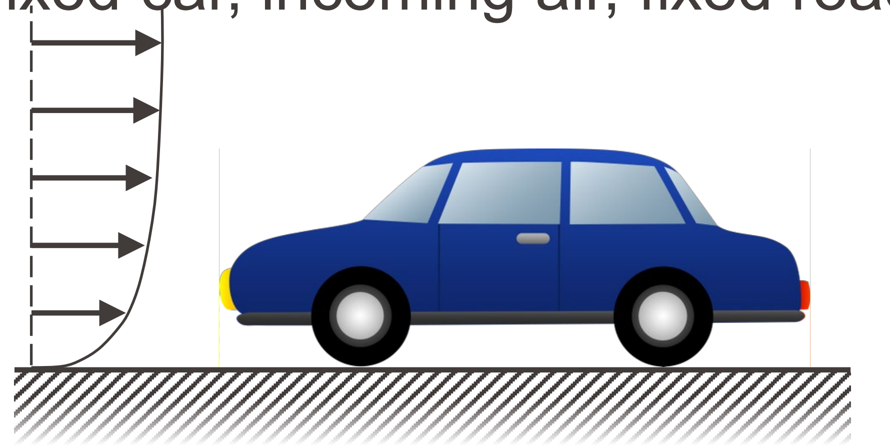
Moving car, still air, moving road



Fixed car, incoming air, moving road

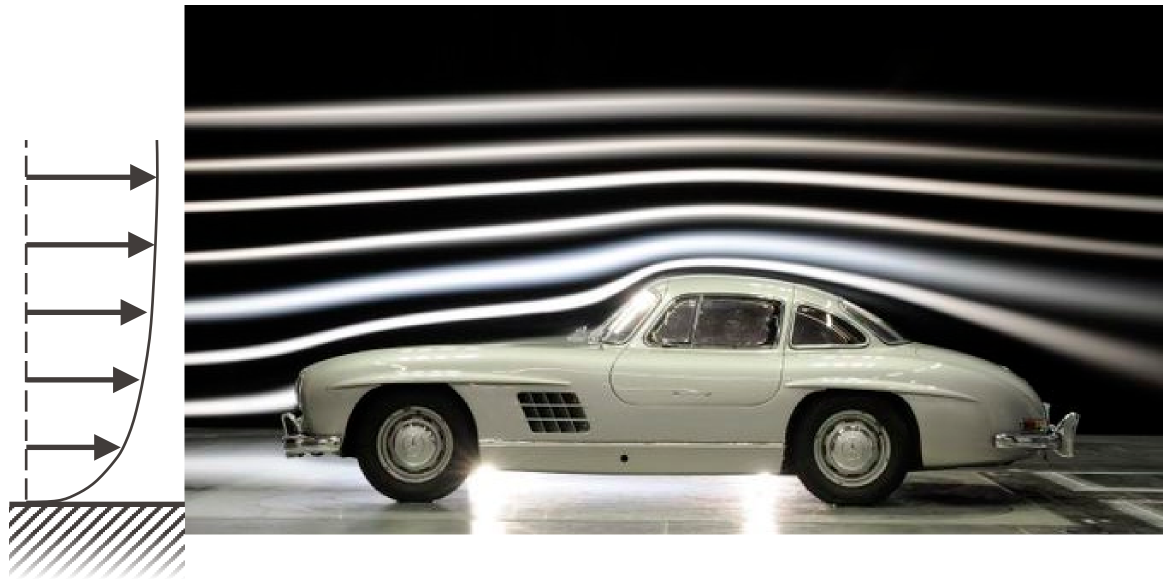


Fixed car, incoming air, fixed road



Modern wind tunnels with moving ground:

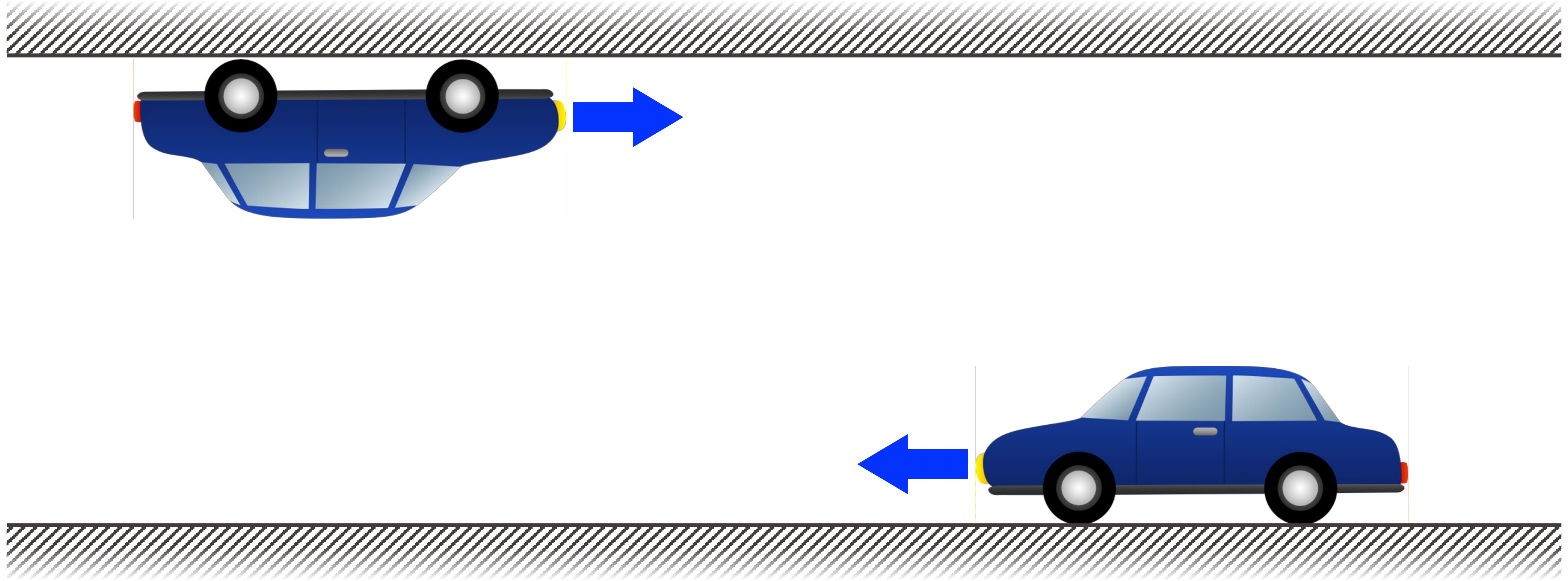
Traditional wind tunnels without moving ground:



Unsteady problems with moving boundaries

- **Inertial reference frame** (uniform rectilinear motion) → **stationary** problem in the moving frame.

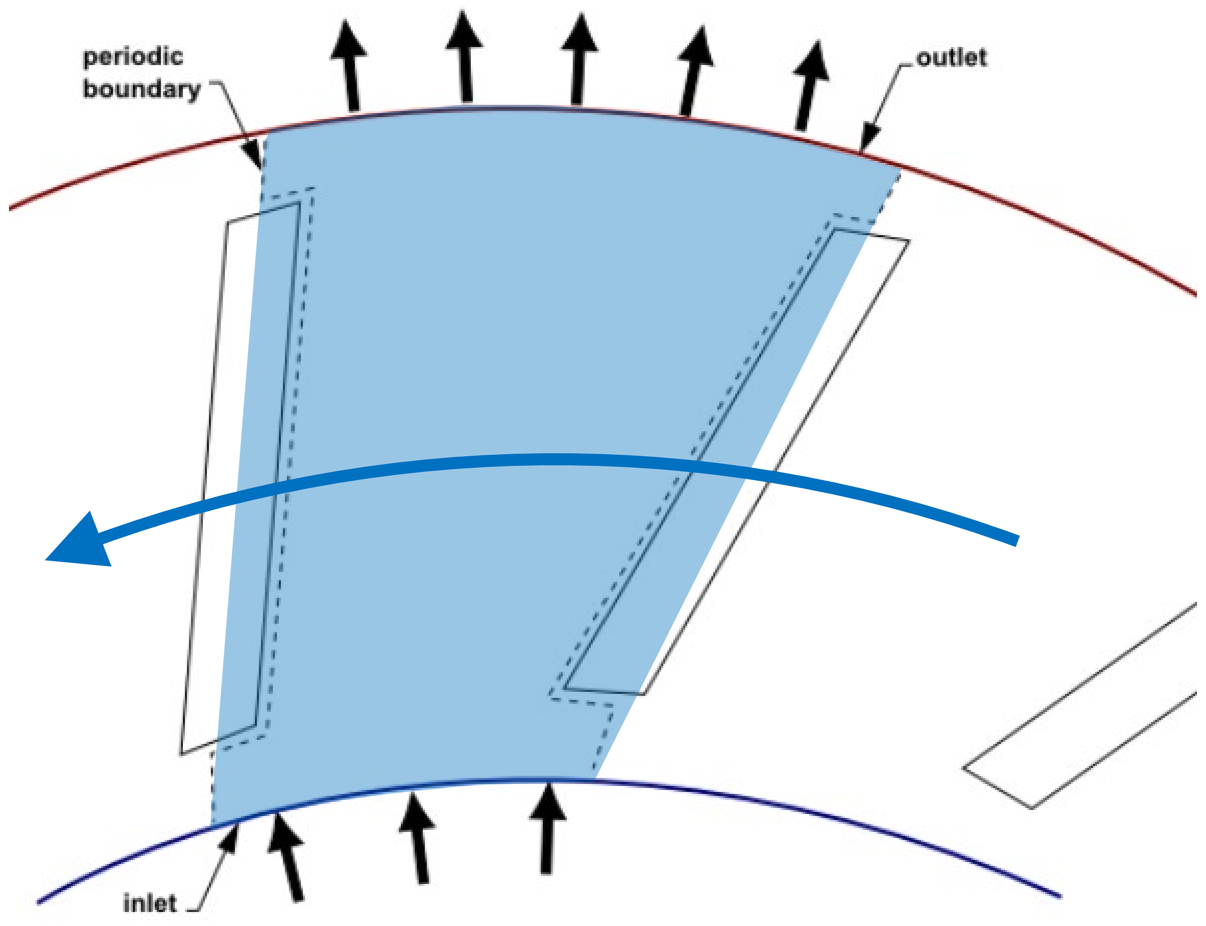
- Approach **not possible** if **multiple** inertial frames moving with different velocities (inherently unsteady problem):



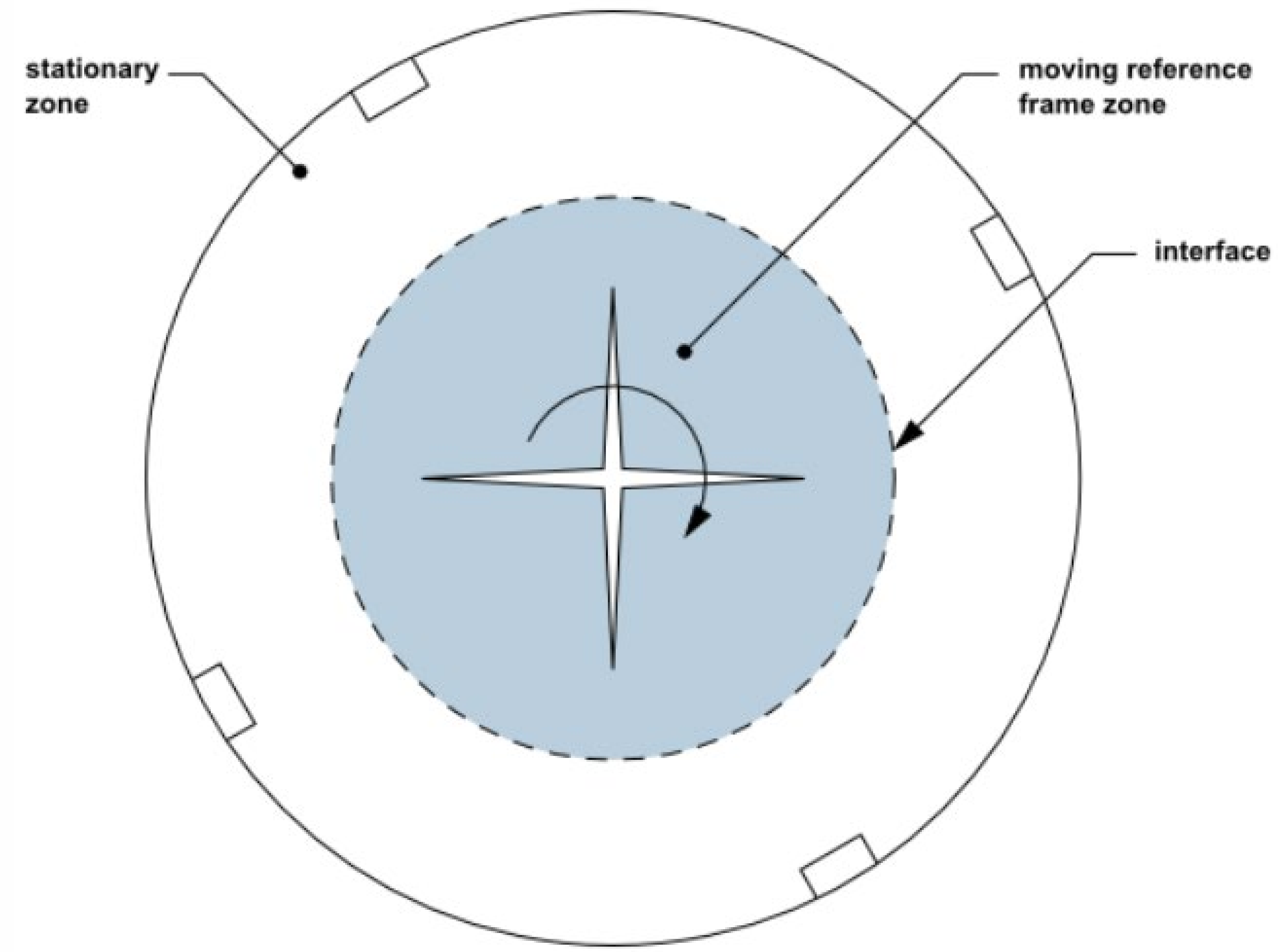
Unsteady problems with moving boundaries

- **Non-inertial reference frames**: non uniform (linear acceleration) or non rectilinear (rotational acceleration) → still a **stationary** problem in the moving frame, but need to add **inertial terms** to the eqs (e.g. centrifugal and Coriolis forces).
- In Fluent: “moving reference frame”

“Single reference frame” (SRF):
single fluid domain attached to a moving frame



“Multiple reference frame” (MRF):
several fluid domains, attached to different frames
(→ different equations solved in each domain)



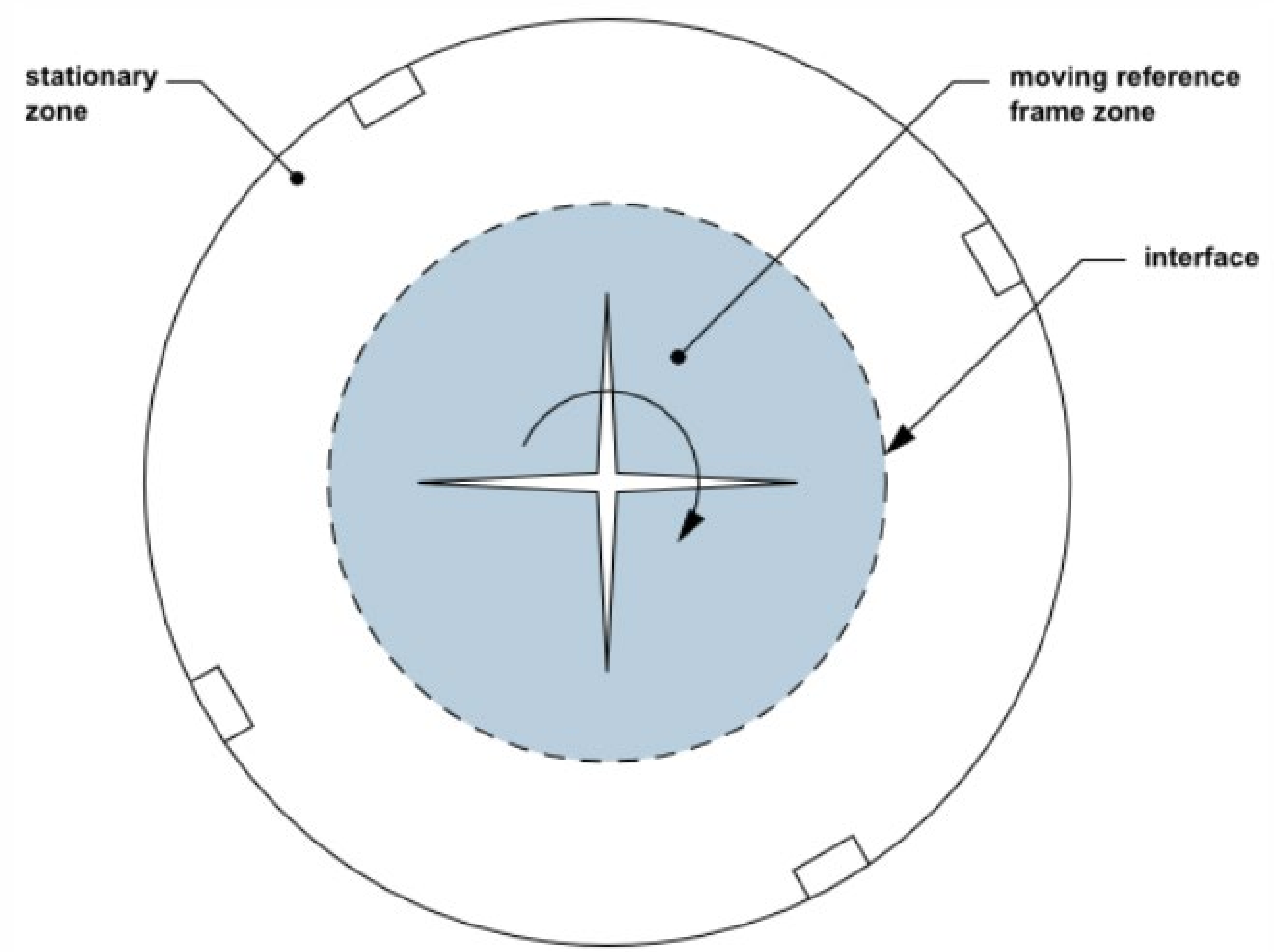
Each domain seen in its own moving frame → stationary problem, no mesh deformation.

Unsteady problems with moving boundaries

- **Non-inertial reference frames**: non uniform (linear acceleration) or non rectilinear (rotational acceleration) → still a **stationary** problem in the moving frame, but need to add **inertial terms** to the eqs (e.g. centrifugal and Coriolis forces).
- In Fluent: "moving reference frame"

"Multiple reference frame" (MRF):
several fluid domains, attached to different frames
(→ different equations solved in each domain)

Note: the MRF model is an **approximation**. The motion of the moving part is frozen.
It may be suitable for weak interaction between moving and stationary parts, and for uncomplicated flows at the interface.

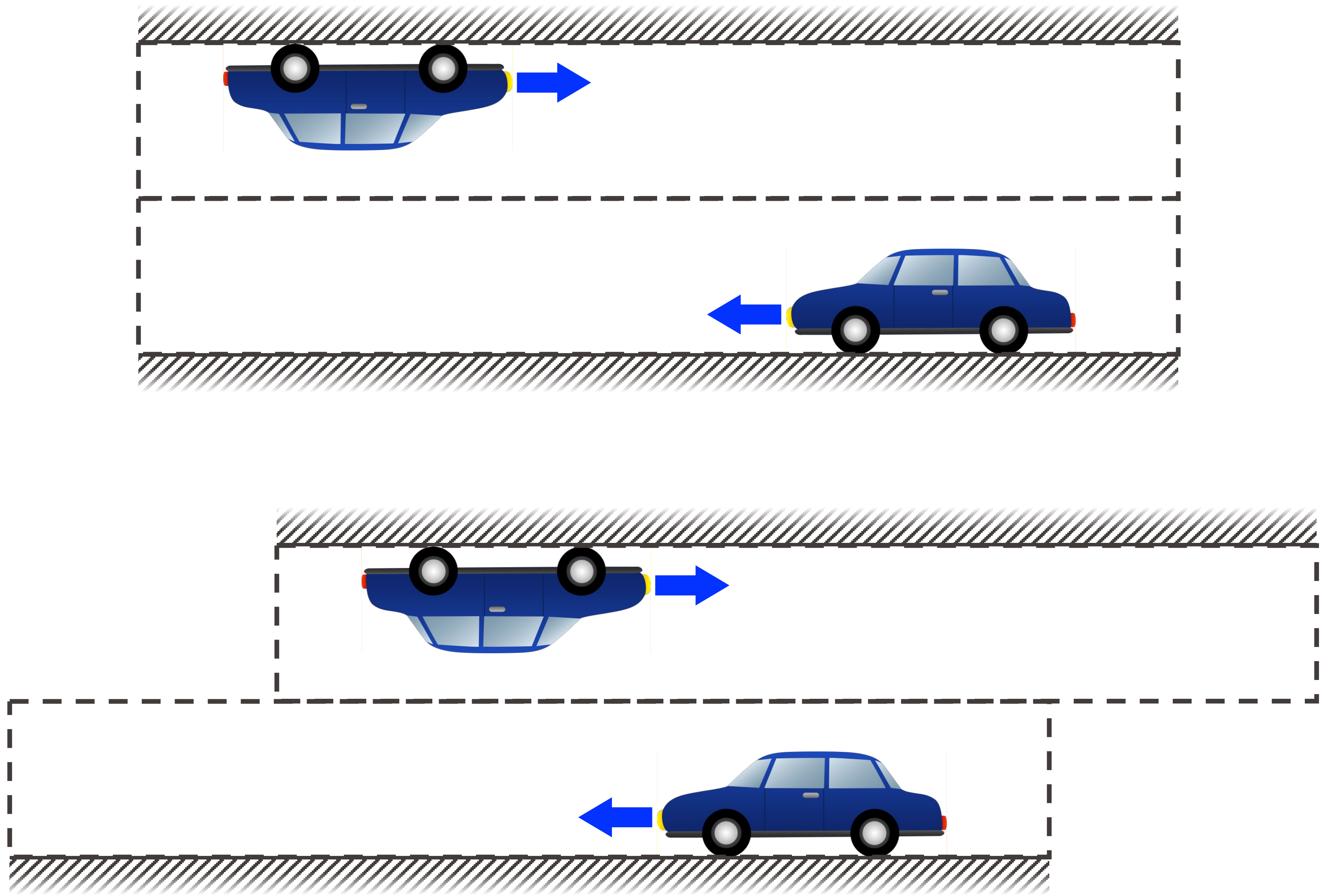


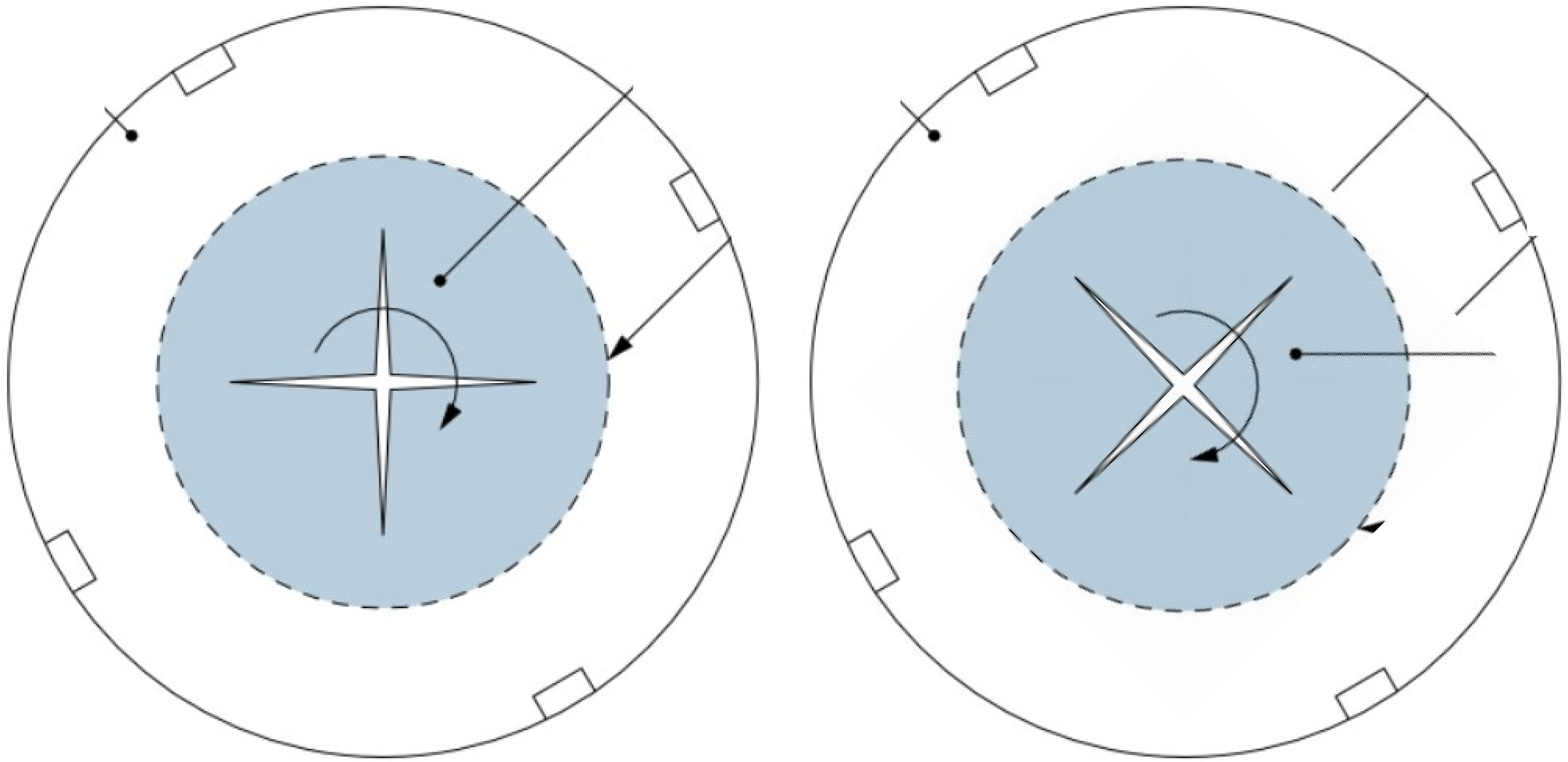
Unsteady problems with moving boundaries

- **Unsteady** flows with moving/deforming boundaries → adjust the mesh over time.
- In Fluent: "sliding mesh" and "dynamic mesh".

"Sliding mesh"

Rigid-body motion of the mesh in each domain. The interface can only slide (no normal motion).



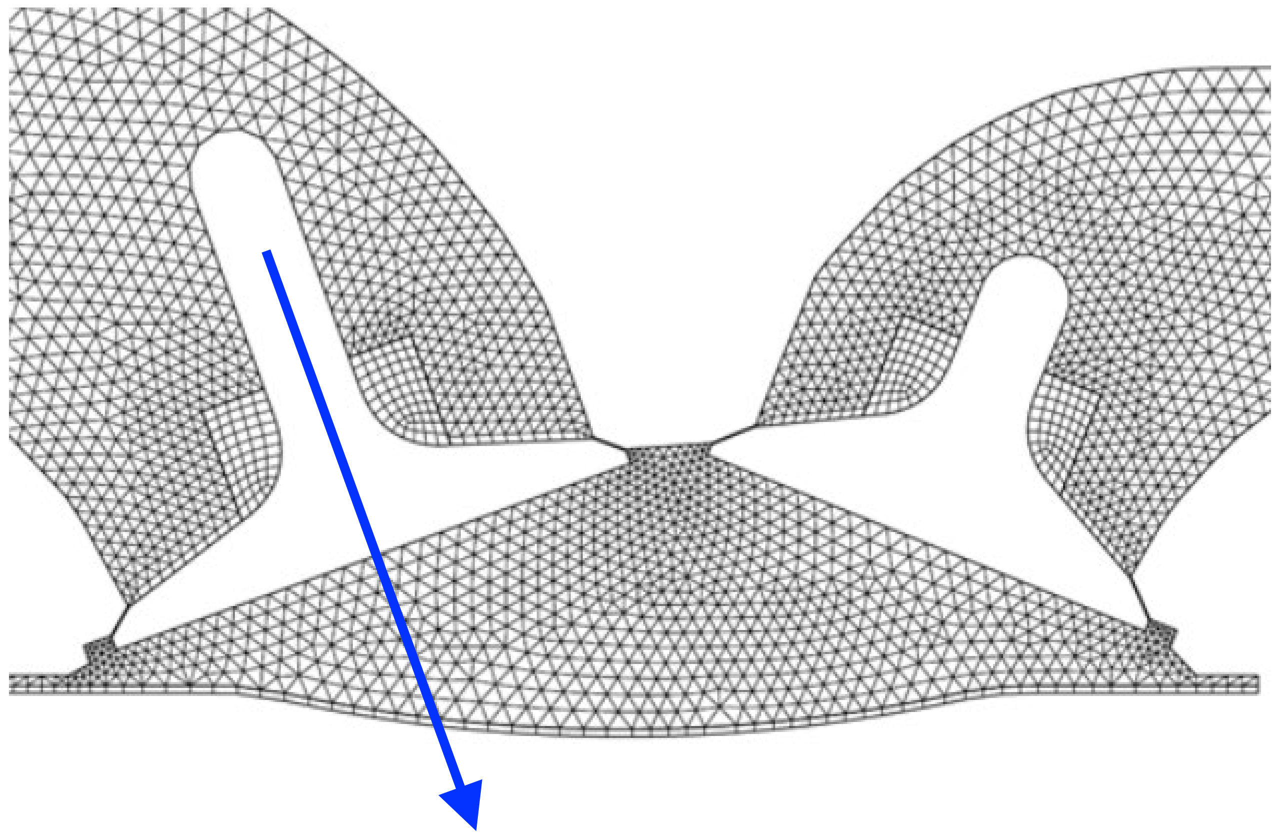


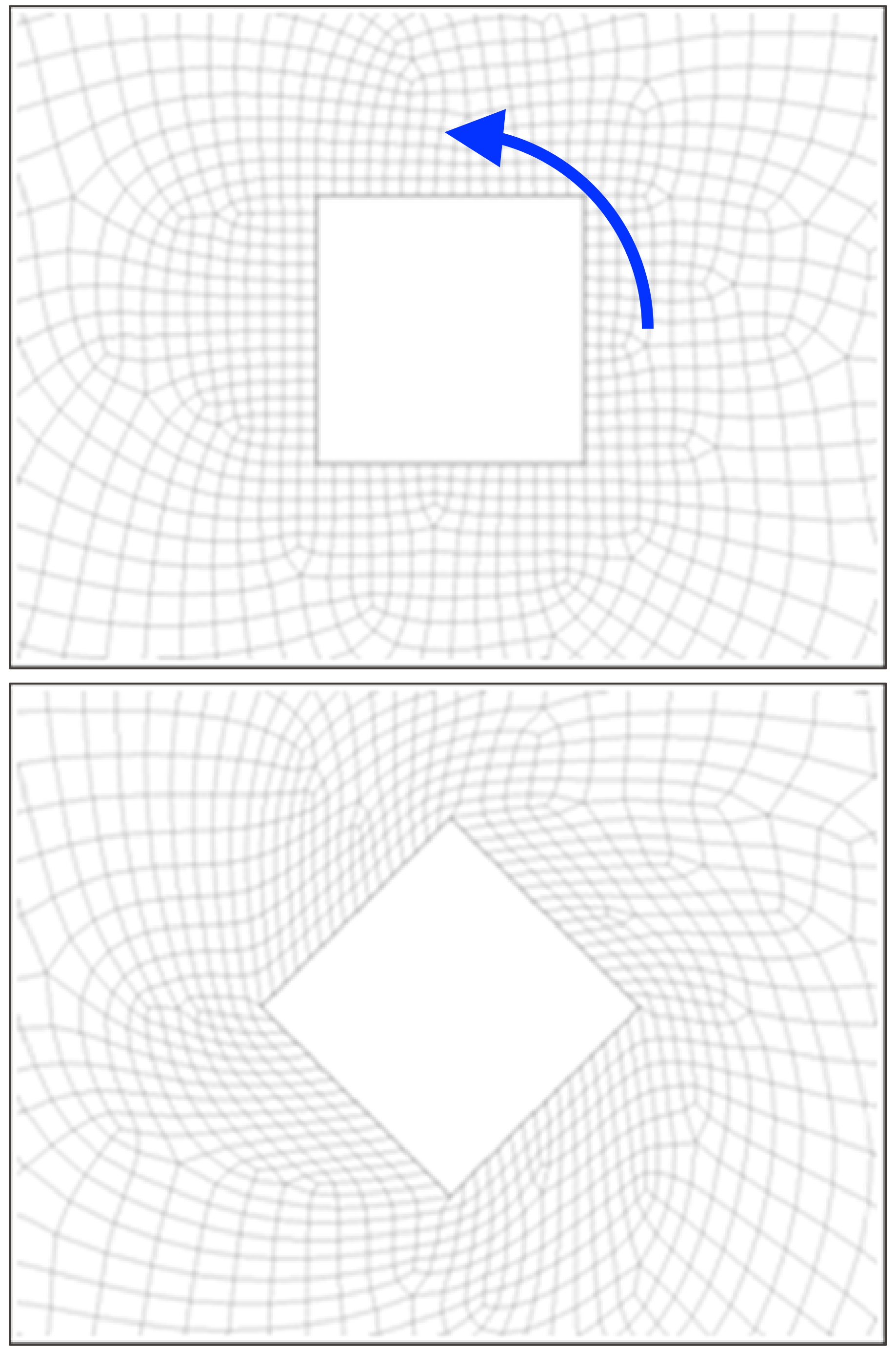
Unsteady problems with moving boundaries

- **Unsteady** flows with moving/deforming boundaries → adjust the mesh over time.
- In Fluent: “sliding mesh” and “dynamic mesh”.

“Dynamic mesh”

The shape of the domain is changing with time → the mesh must be updated



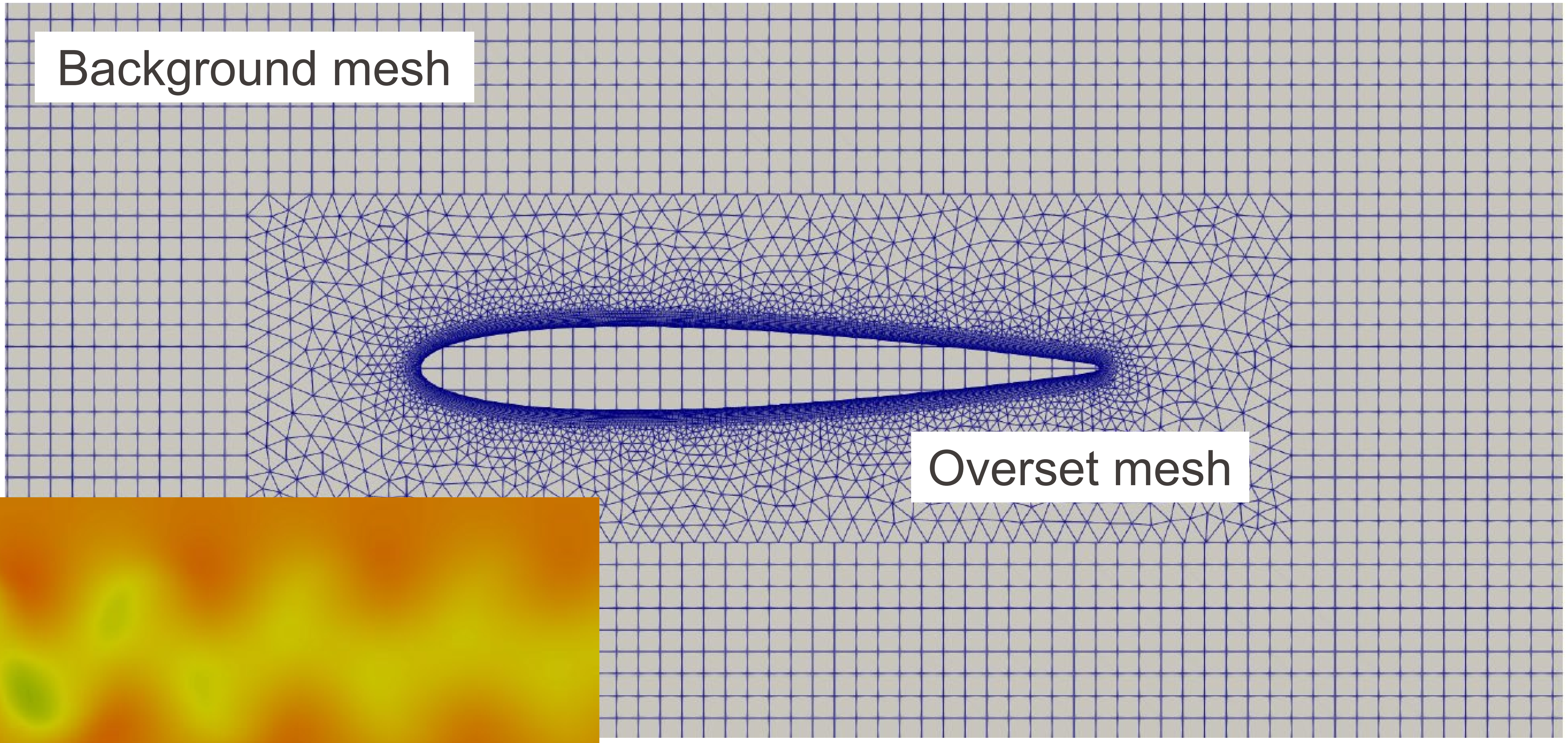


Update methods:

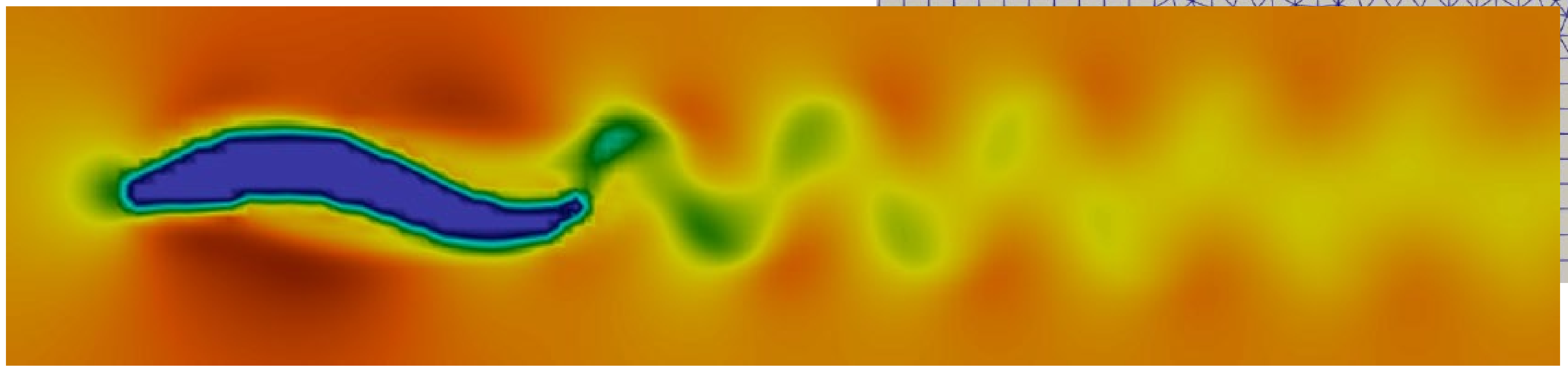
- **Smoothing**: move interior nodes to “absorb” the motion of the boundaries (several methods: diffusion, spring, elastic solid, etc.)
- **Dynamic layering**: add (split) or remove (merge) layers next to the boundaries (only for quads in 2D, and prisms/hexas in 3D)
- **Remeshing**: create a new mesh in some regions (based on mesh quality, cell zone, etc.)

Unsteady problems with moving boundaries

- **"Overset mesh"** method: use 2 (or more) overlapping meshes.
  - "Background mesh": usually fixed, and in the whole domain,
  - "Overset mesh" (or "component mesh"): around the moving/deforming boundary,
  - Overlapping region: interpolation between meshes (can have different element types).



Example: fish locomotion with large body deformation, *Re*=5000

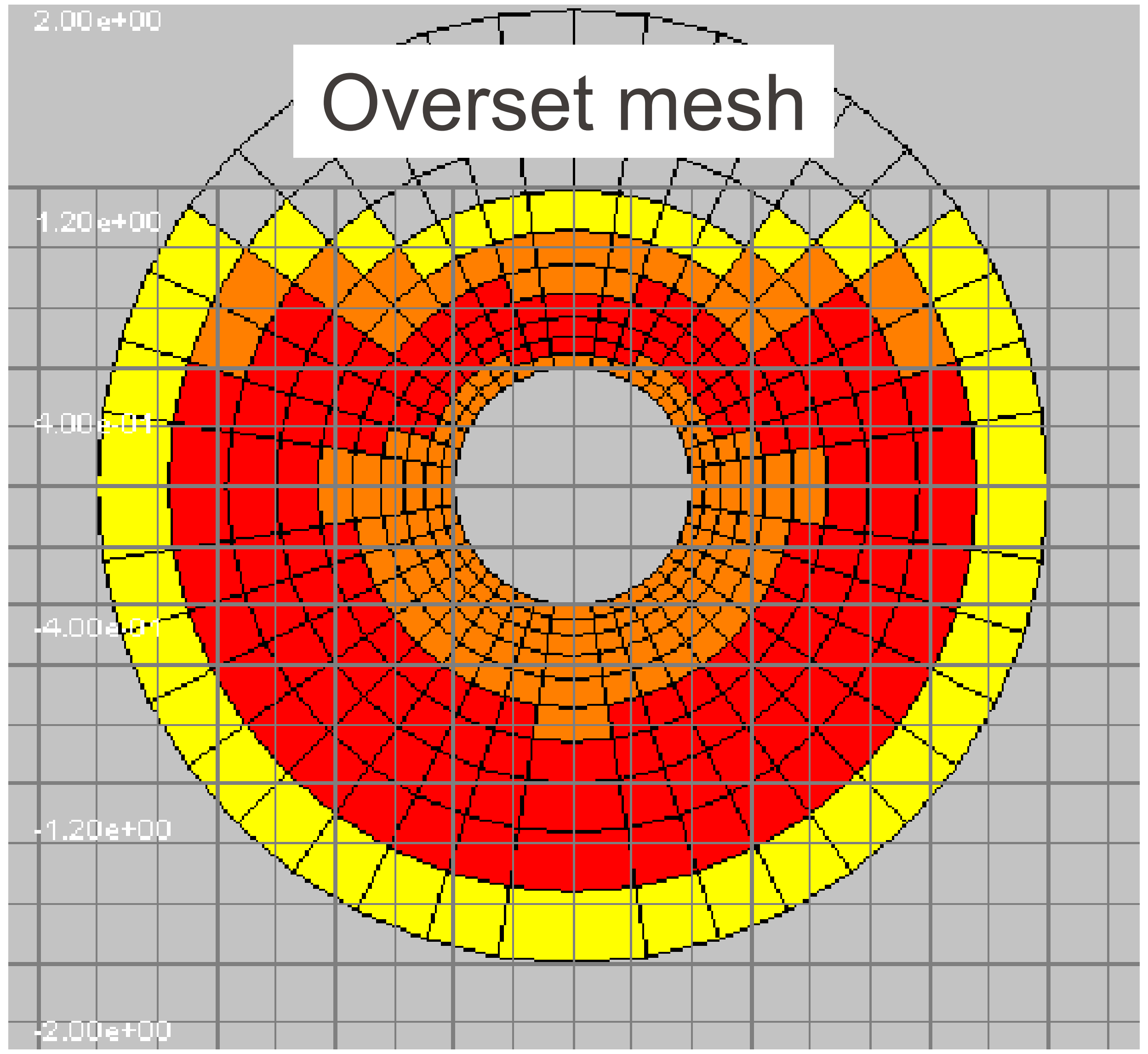


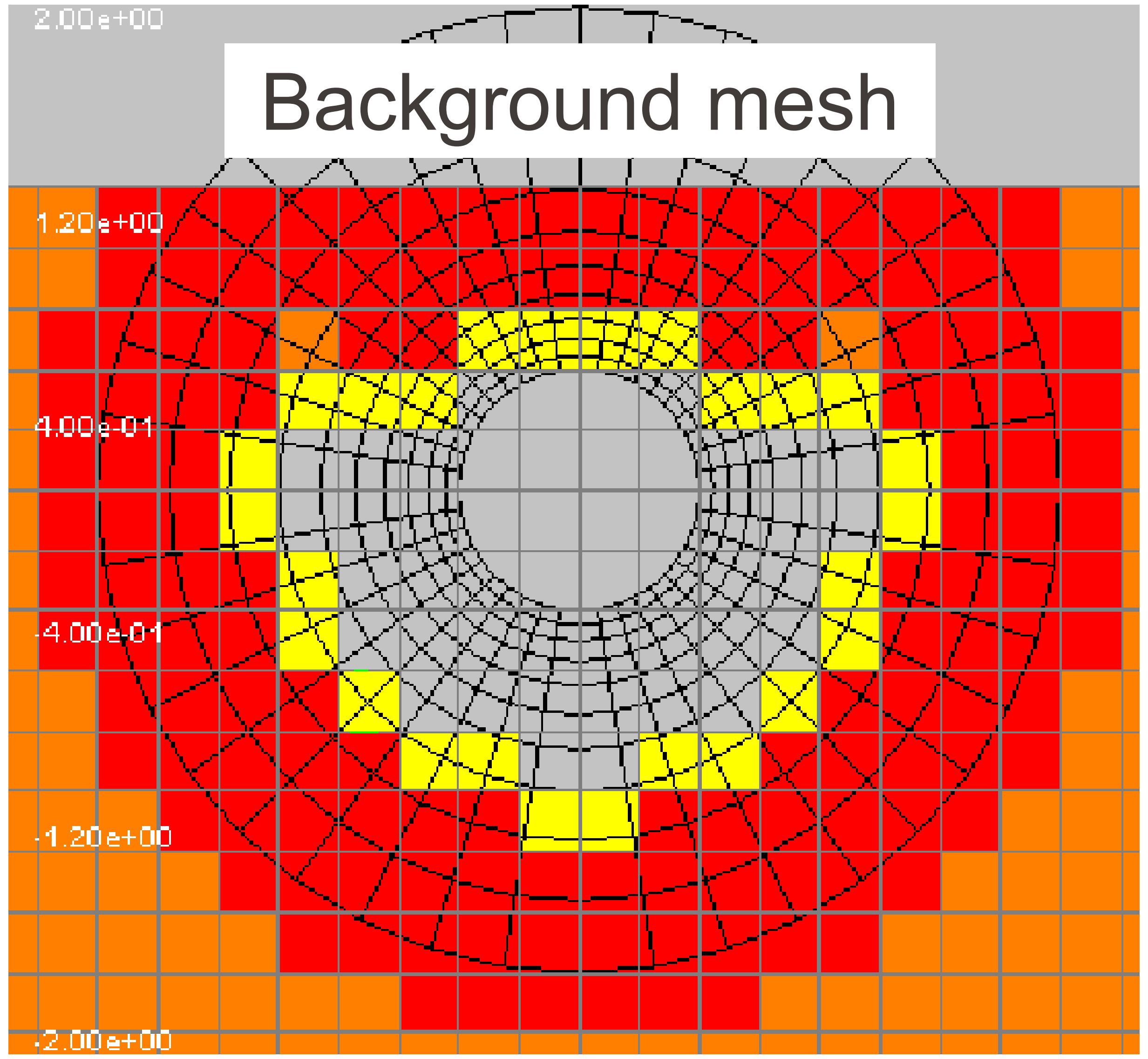
Coe & Gutschmidt (2023)

Unsteady problems with moving boundaries

- **"Overset mesh"** method: use 2 (or more) overlapping meshes.
  - "Background mesh": usually fixed, and in the whole domain,
  - "Overset mesh" (or "component mesh"): around the moving/deforming boundary,
  - Overlapping region: interpolation between meshes (can have different element types).

Example: 2D cylinder





Red/orange: solve cells, where equations are solved

Yellow: receptor cells, receive data interpolation from other mesh

Unsteady problems with moving boundaries

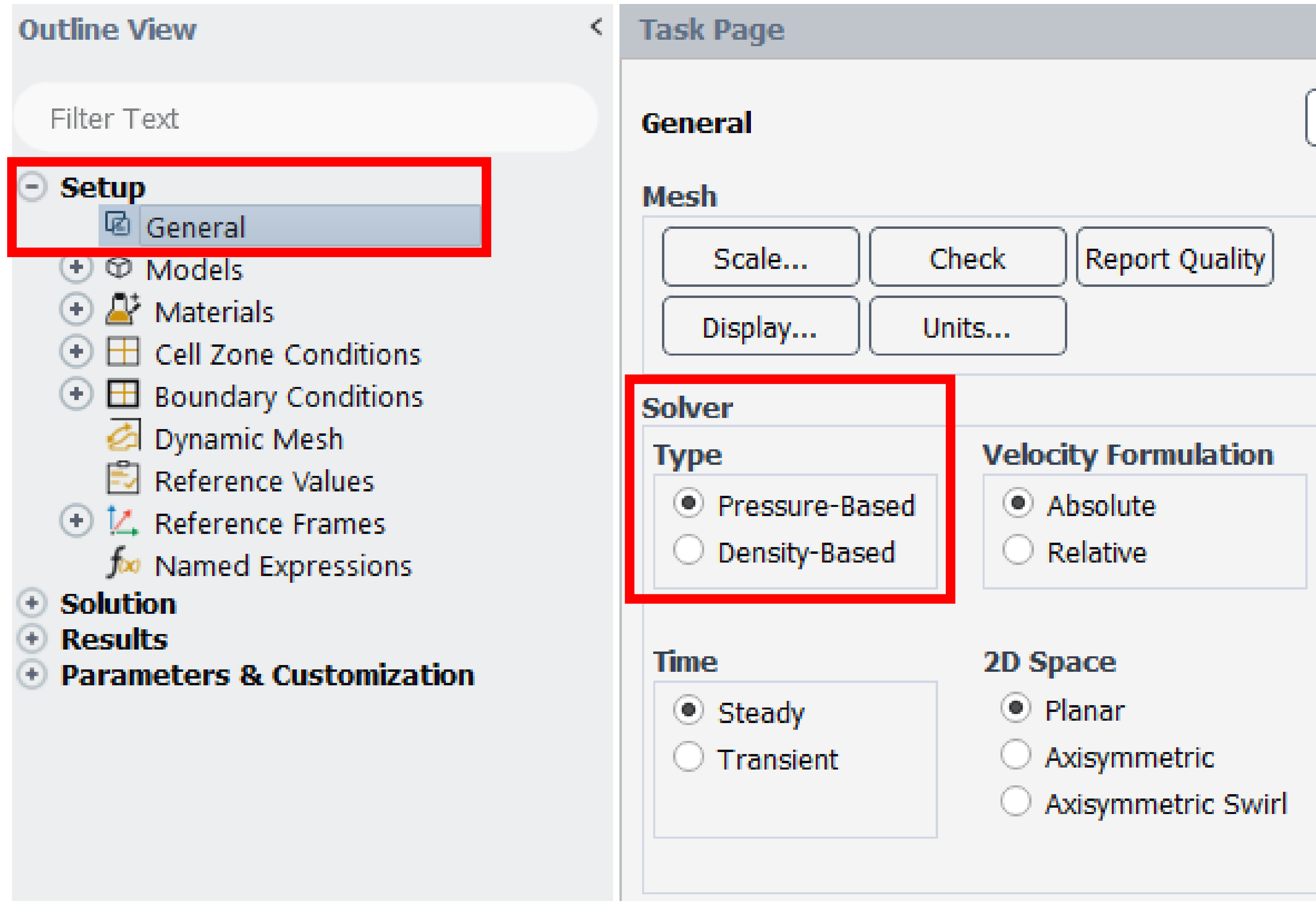
- **“Overset mesh”** method: use 2 (or more) overlapping meshes.
  - “Background mesh”: usually fixed, and in the whole domain.
  - “Overset mesh” (or “component mesh”): around the moving/deforming boundary.
  - Overlapping region: interpolation between meshes (can have different element types).

- Pros: allows for complex motion, larger time steps, less mesh quality problems.
- Cons: interpolation error.

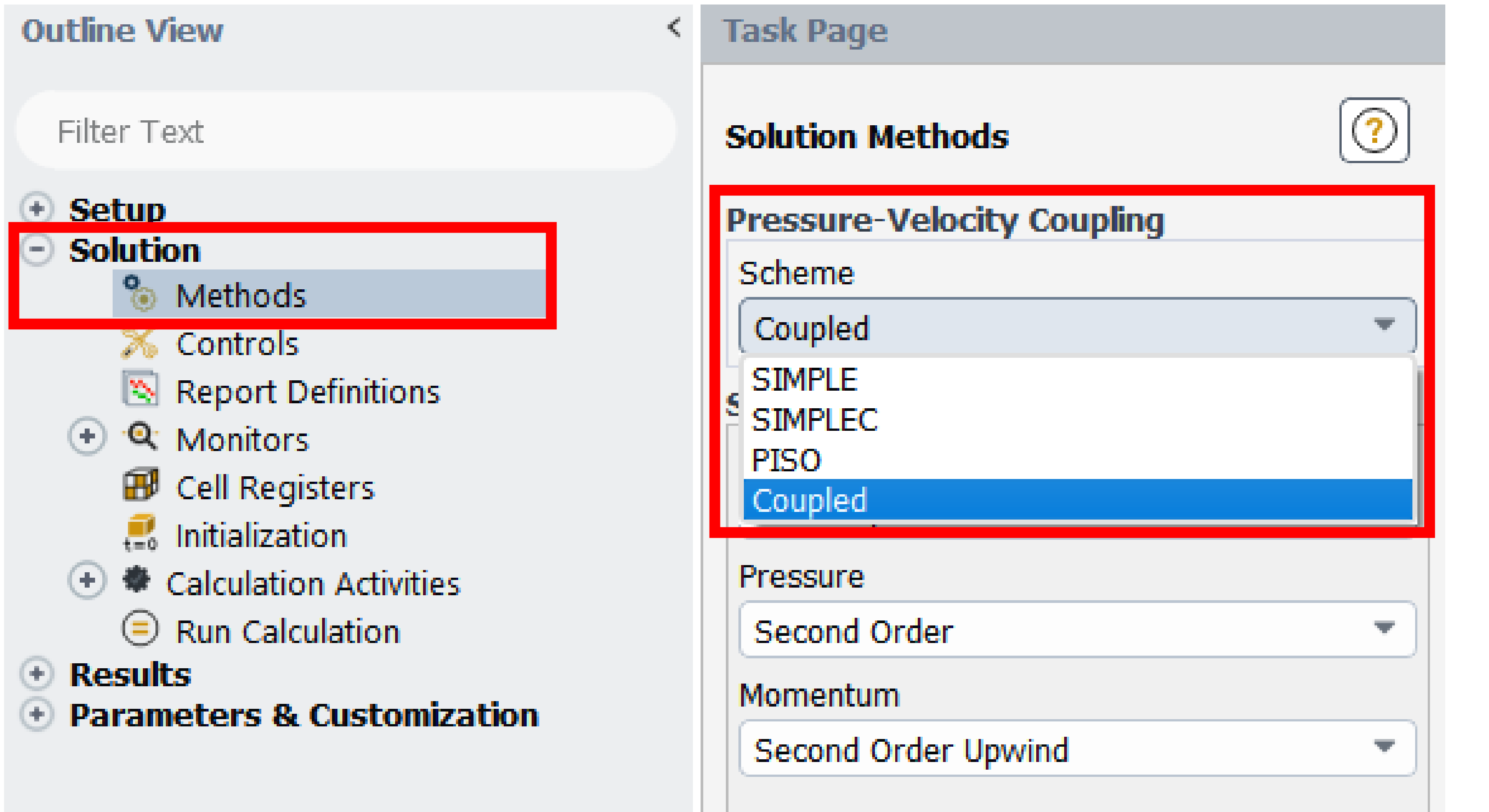
- Elements should have similar sizes in the overlapping region.
- The time step should limit the relative mesh motion to the size of the smallest cell in the overlapping region.

Choosing the numerical method

- Two basic solvers in Fluent:
  pressure-based (PB) and **density**-based (DB)
- Both can be used for a wide range of flows.
  - PB originally developed for **incompressible** flows. Indicated for incompressible / mildly compressible flows.
  - DB originally developed for **compressible** flows. Indicated for high-speed compressible flows.



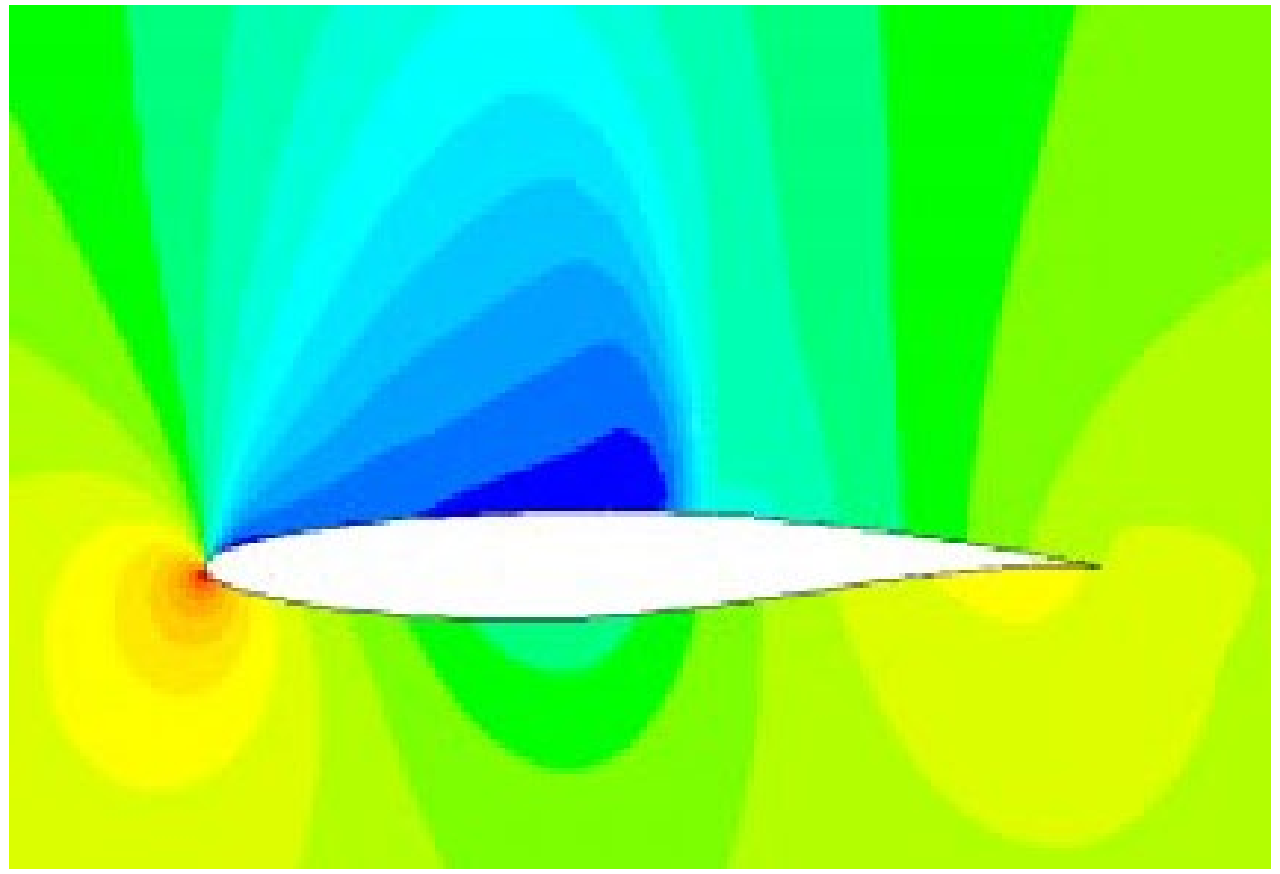
- Two main PB variants: **segregated** and **coupled** (see week 5).
  - Coupled solver generally **converges faster** than segregated solver, but uses approx. 2 times **more memory**.

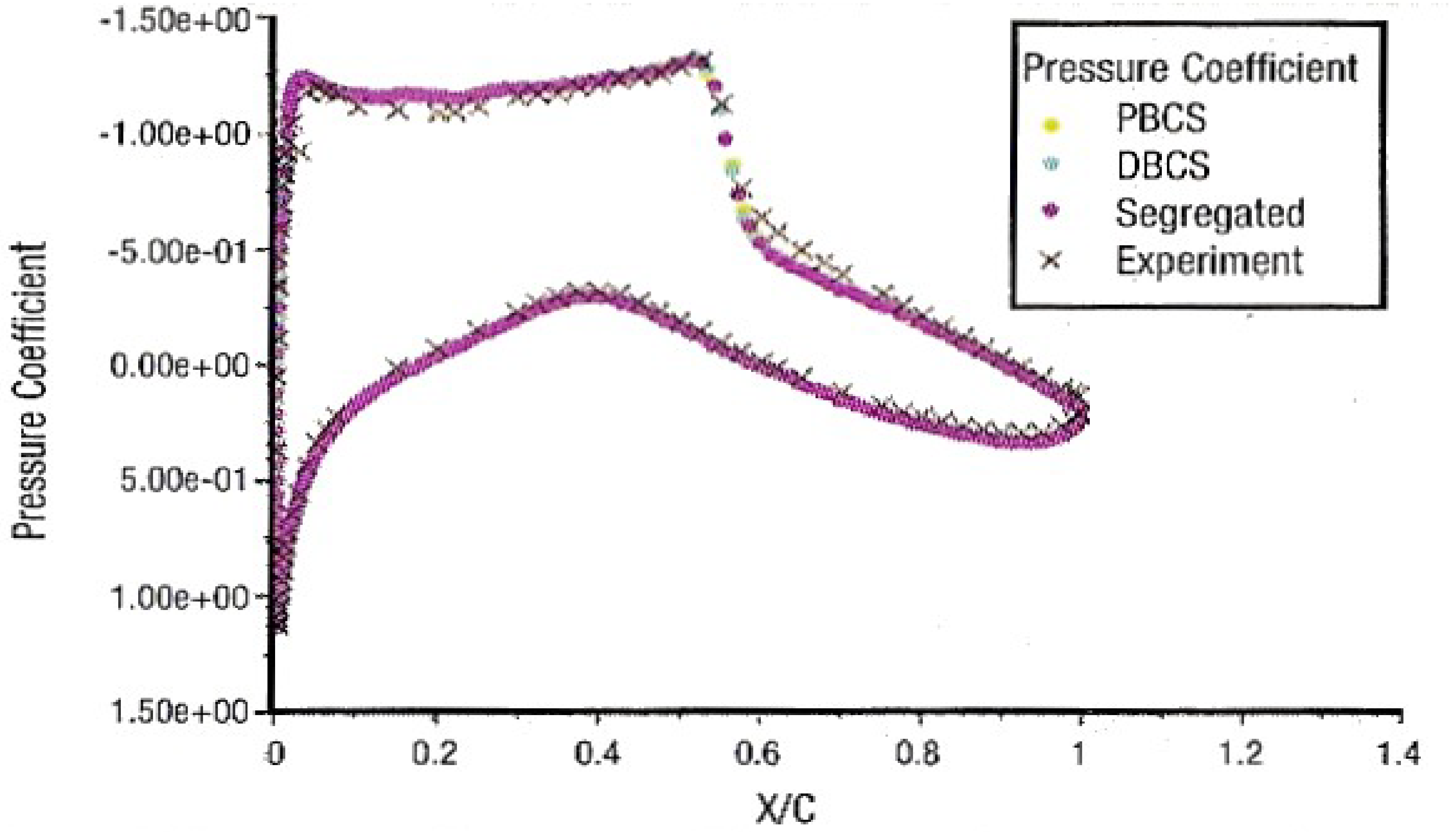


- DB always coupled. Not available for some physical models (e.g. multiphase, combustion…).

Choosing the numerical method

- Example: transonic flow over RAE 2822 airfoil (M=0.73, Re=6.5x10^6, α=2.8°)





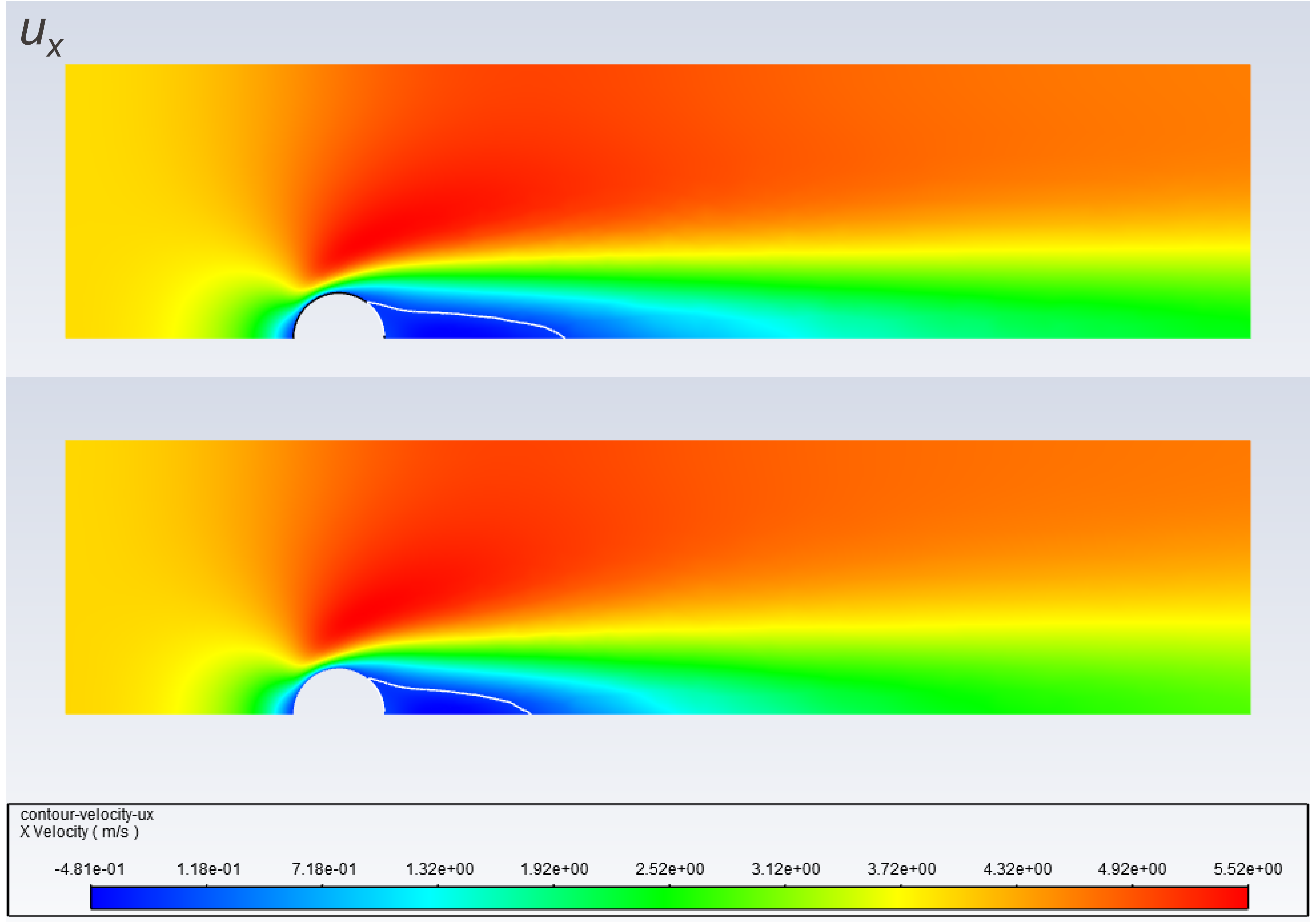
Solver	Memory [MB]	Time per iteration [sec]	Iterations to convergence	Time to convergence [min]
PB segregated	**172**	2.1	2570	90
PB coupled	259	3.3	**298**	16
DB coupled	317	3.8	**976**	62

Choosing the numerical method

- Spatial discretization: 1st vs 2nd order (see week 4)

2nd order

1st order



Numerical Flow Simulation

Choosing the numerical method

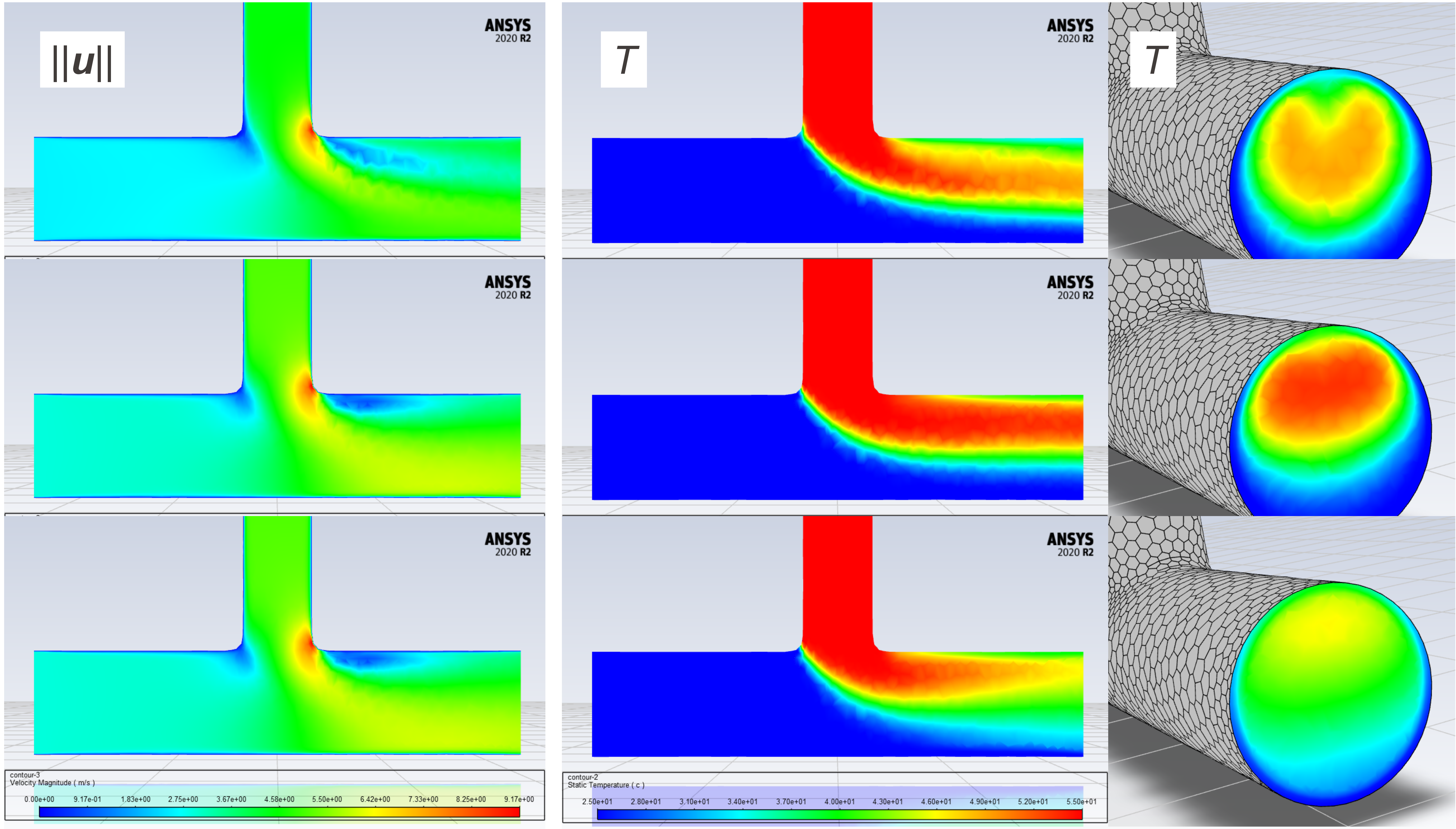
- Spatial discretization: 1st vs 2nd order (see week 4)

Momentum & energy eqs.:

2nd & 2nd orders

1st & 2nd orders

1st & 1st orders



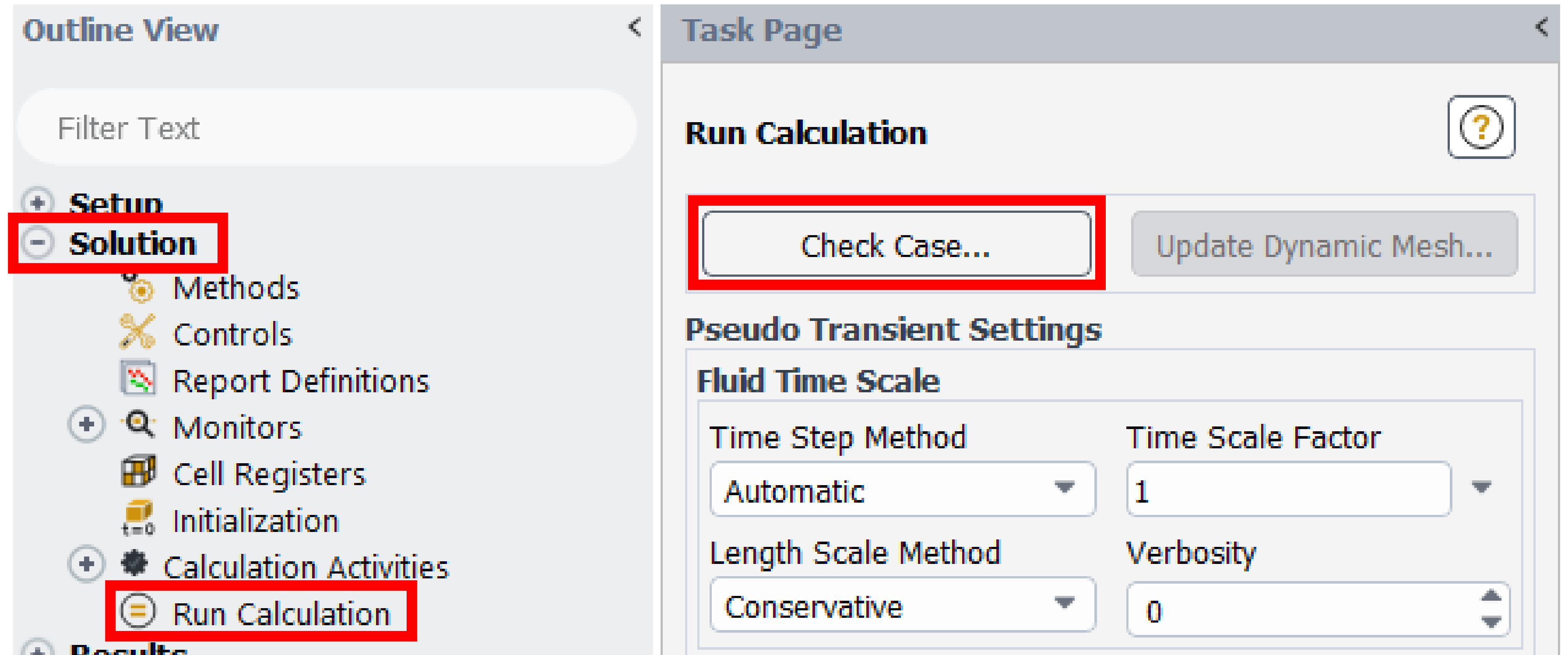
Running the solver

- **Initialize** the flow field. Various possibilities:
  - Constant values
  - Different constant values in different regions ("patches")
  - "Hybrid" initialization (solves Laplace equation for velocity potential and pressure)
  - Last computed values (if re-starting the computation)
  - Interpolation of solution computed on another mesh.

$$\nabla^2\varphi = 0$$
$$\nabla^2 p = 0$$

- Choose **monitor** quantities:
  - Residuals
  - Surface integrals (e.g. forces/moments, average value, standard deviation…)
  - Pointwise values (e.g. pressure, velocity, temperature…)

- Set max. **number of iterations**.
- **Check** the setup ("check case").
- **Calculate**.



Checking convergence

- At convergence:
  - All discretized equations are satisfied in all cells to a specified tolerance.
  - Solution no longer changes significantly with more iterations.
  - Overall mass, momentum, energy and scalar balances are obtained.

- Monitor convergence with residuals:
  - Generally, decrease in residuals by 10^3 indicates basic global convergence: major flow features have been established; may be sufficient for “industrial flows” (depending on required accuracy)
  - Scaled energy residual must decrease by 10^6 for segregated solver
  - Scaled species residual may need to decrease by 10^5 to achieve species balance

- Monitor convergence with physical quantities.

- Check conservation: mass and heat balances should be within 0.2% of net flux.

Checking convergence

- Convergence difficulties can arise due to:
  - Ill-posed problem (no physical solution, e.g. velocity inlet without outlet; pressure inlet and outflow)
  - Inappropriate boundary conditions
  - Poor-quality mesh
  - Inappropriate initial conditions
  - Inappropriate solver settings
  - …

- Trouble-shooting:
  - Ensure problem is physically realizable
  - Check boundary conditions. Move boundaries farther if needed.
  - Re-mesh or refine mesh in regions with high aspect ratio or highly skewed cells
  - Compute initial solution with 1st-order discretization scheme
  - Decrease under-relaxation for eqs. with convergence issues
  - Reduce time step / CFL number (unsteady flow)
  - …

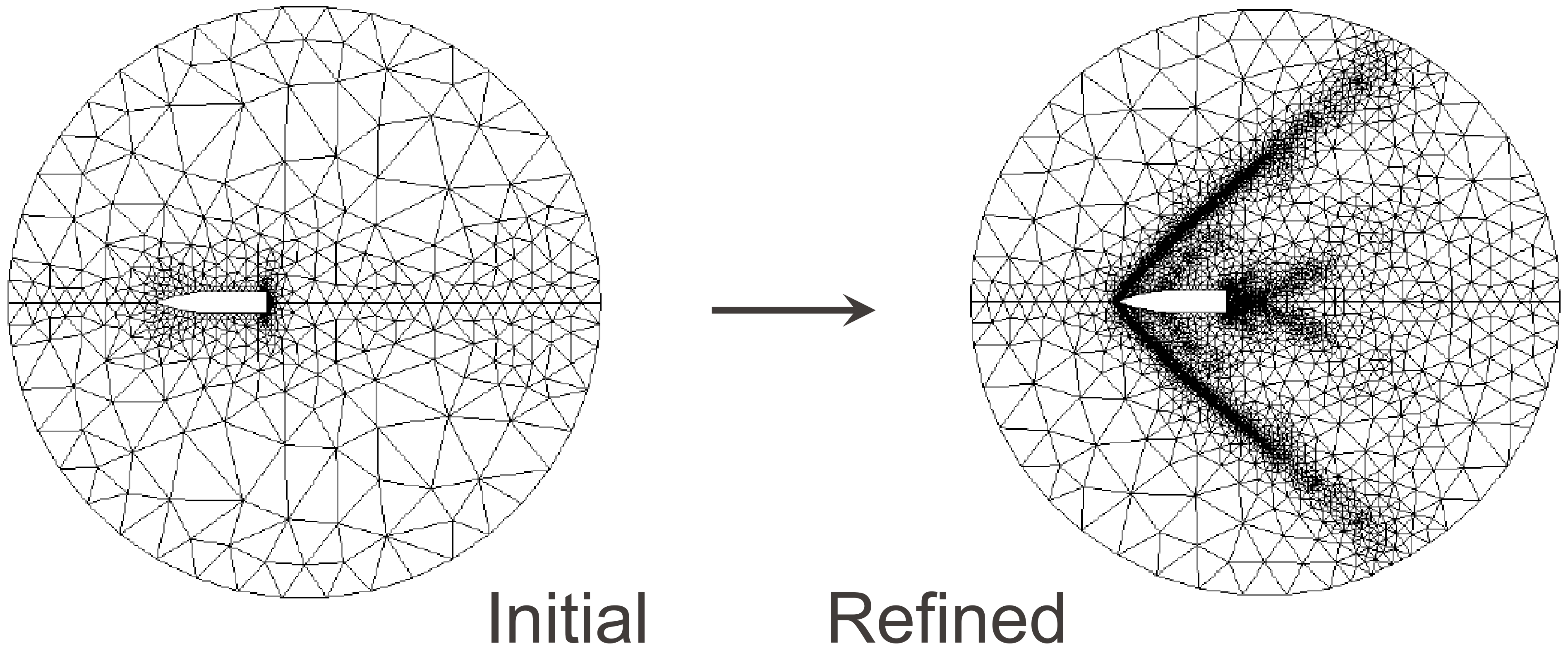
Analyzing the solution

- Qualitative analysis (visualization):
  - Displaying the mesh
  - Contours of flow fields (e.g. pressure, velocity, temperature, concentrations…)
  - Contours of derived field quantities (e.g. vorticity, shear stress…)
  - Velocity vectors
  - Animation

- Quantitative analysis:
  - XY plots (e.g. pressure, velocity, temperature… vs. position)
  - Forces and moments on surfaces
  - Surface and volume integrals (average, standard deviation…)

Improving the solution

- Adaptive mesh refinement
  - Mesh adaption adds more cells where needed to better resolve the flow field.
  - Cells to be adapted are listed in a register.
  - Registers can be defined based on:
    - Gradient of flow variables (or user-defined variables)
    - Iso-value of flow variables (or user-defined variables)
    - Whole boundary/region
    - Cell volume, or volume change
    - Distance from walls (y^+)
    - A combination of the above



Summary

- Problem set-up must contain important physical flow effects.

- Choice of boundary conditions influences:
  - Convergence of iterative procedure
  - Physical reality of converged solution

- Numerical method employed influences:
  - Convergence of iterative procedure
  - Computer time and memory storage