Summer School 2017

Computational Continuum Mechanics in Petroleum Engineering

29.06.2017 ▪ Petr Vita

DPE Department Petroleum Engineering
Montan Universität

Content

I. Computational Continuum Mechanics
What is Continuum Mechanics?

Continuum Mechanics is a study of the physics of continuous materials

<table>
<tr><th colspan="4">Continuum Mechanics</th></tr>
<tr><td colspan="2">Fluid Mechanics
Liquids, gases and plasmas</td><td colspan="2">Solid Mechanics
Materials with defined rest state</td></tr>
<tr><td rowspan="2">Newtonian</td><td>Non-Newtonian</td><td>Plasticity</td><td rowspan="2">Elasticity</td></tr>
<tr><td colspan="2">Rheology</td></tr>
</table>

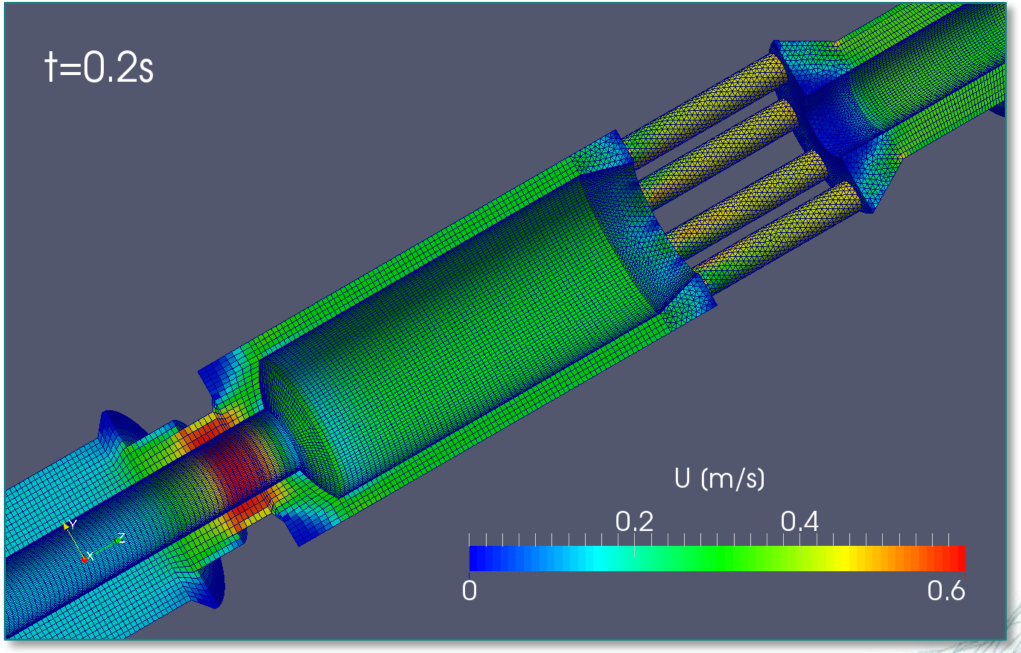
I. Computational Continuum Mechanics (2)
What about Computational?

Computational suggests use of numerical methods and algorithms to solve problems in the focus

- **Numerical analysis is usually not looking for exact solutions, but rather for approximate solutions with well defined error**
  - Finite Difference Method
  - Finite Volume Method
  - Finite Element Method

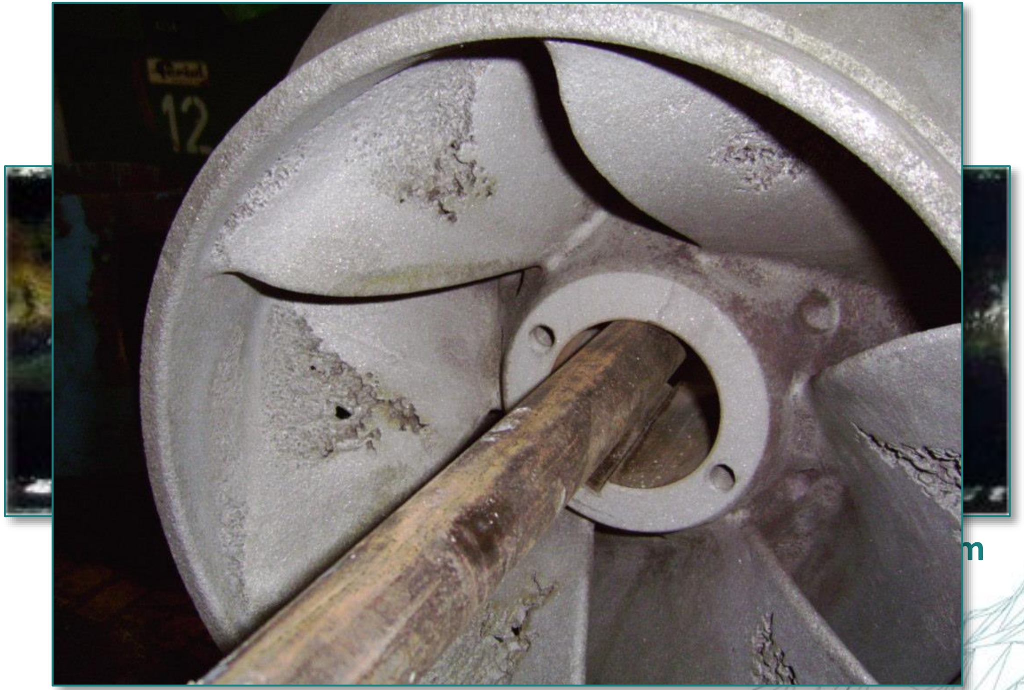
II. Motivation
SRABS Design Optimization

- **Sucker Rod Anti-Buckling System**
  - Optimization of pump design w/OpenFOAM
    - Possible abrasion of the internal components
  - Pre-processor Gambit
  - Dr. mont. Langbauer
    DPE, Montanuniversität Leoben



II. Motivation (2)
Pipelines and Pumps Design

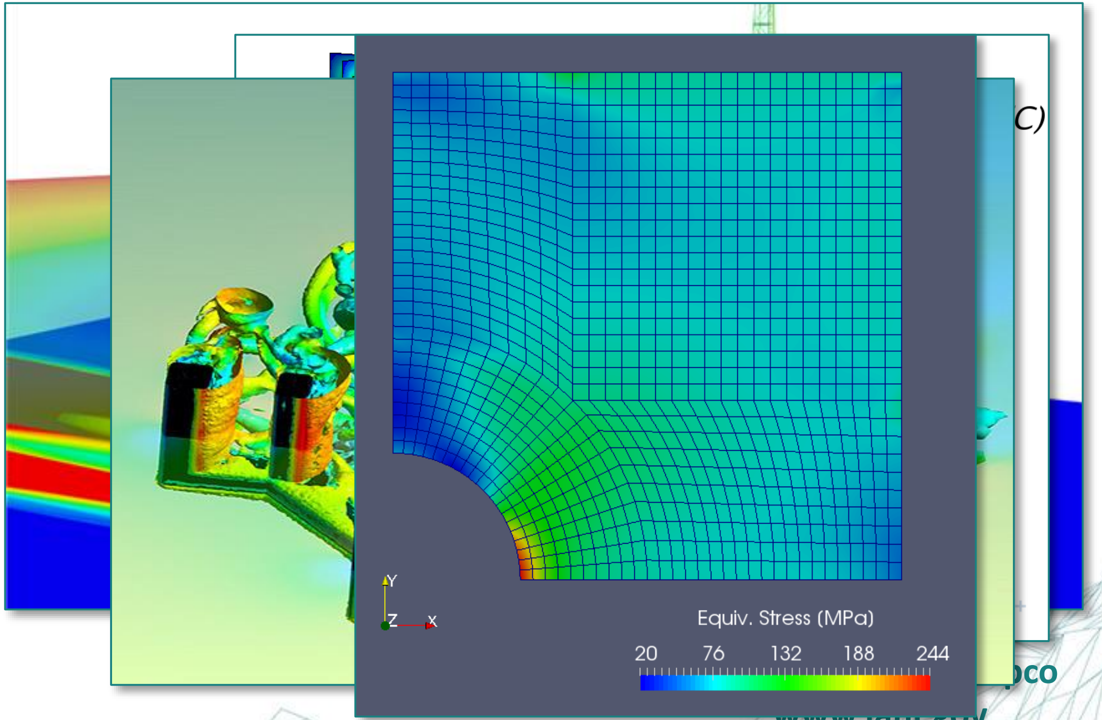
- **Pipeline internal erosion, and corrosion**
  - Cavitation at increased pipe elevations, bends
- **Internal damage of the centrifugal pumps**
  - Cavitation at blade edges, and discharge areas



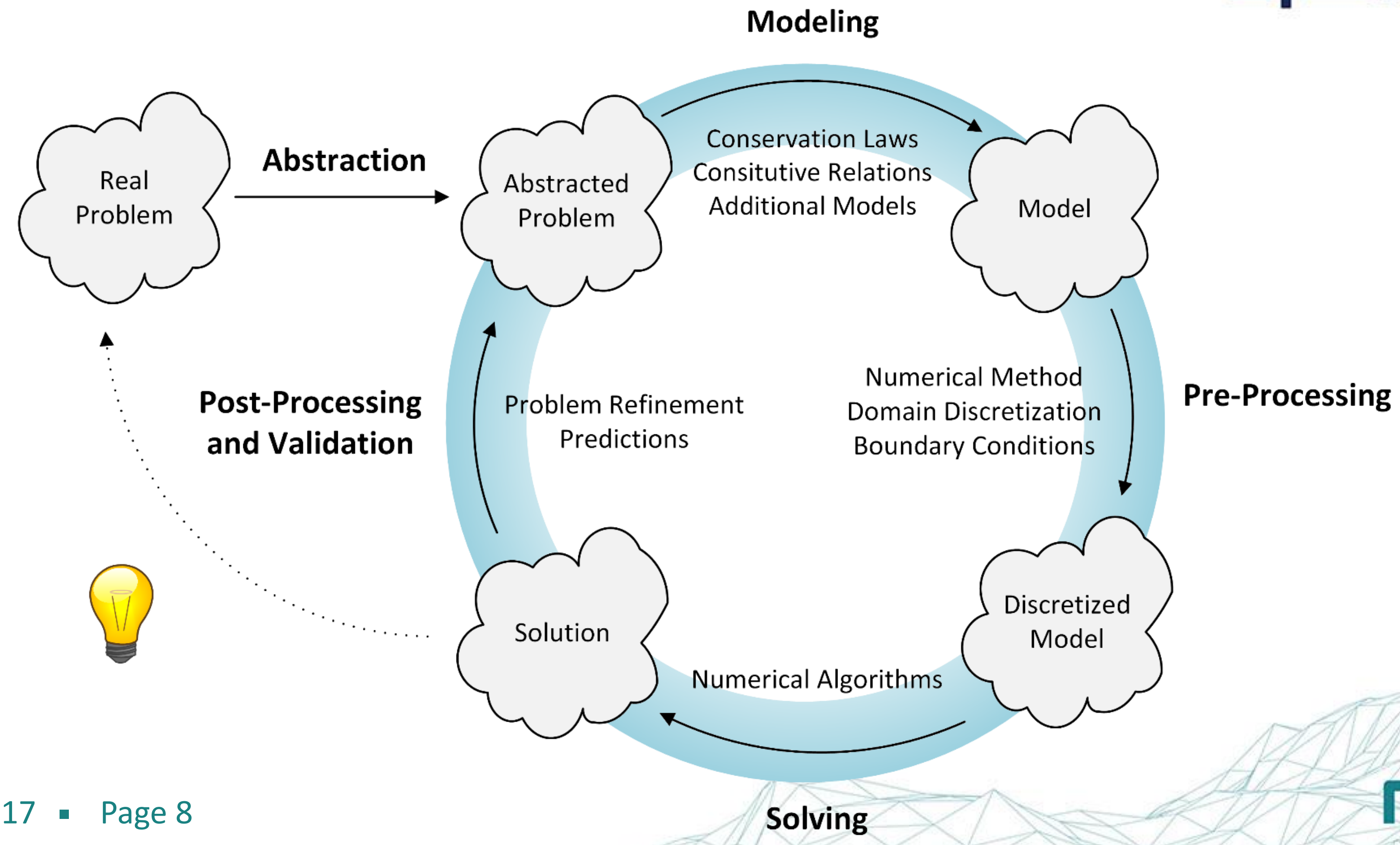
www.belzona.com

II. Motivation (3)
Wind Loading, FSI, CHT, and Stress Analysis

- Wind Loading
- Conjugate Heat Transfer
- Fluid-Structure Interaction
- Stress Analysis
- …and many others



III. Simulation Workflow



III. Simulation Workflow (2)
Abstraction

- **Problem dimensionality**
- **Transient vs Steady-state**
- **Fluid vs Solid**
- **Single-phase vs Multi-phase**
  - Phase changes
- **Compressible vs Incompressible**
- **Laminar vs Turbulent**
- **Inviscid vs Viscous**
  - Non-Newtonian
- **Plastic vs Elastic**
- **Heat Transfer**
- **Fluid-Solid Interaction**
- **Chemical reactions**
- **Additional forces**
- **Motion present**

III. Simulation Workflow (3)
Modeling

- **Conservation, or balance, laws**
  - Mass
  - Momentum (linear, and angular)
  - Energy
- **Constitutive relations**
  - Functional dependencies between quantities, usually specific to material or substance
  - Mathematical closure to governing conservation laws

$$\frac{\partial \rho E}{\partial t} + \nabla \cdot (\rho \mathbf{u} E) = \nabla \cdot \mathbf{q} + \rho Q + \nabla \cdot (\mathbf{T} \cdot \mathbf{u}) + \mathbf{F}_b \cdot \mathbf{u}$$

$$\mathbf{q} = -\kappa \nabla T$$

III. Simulation Workflow (4)
Finding the Model Solution

- **Pick a suitable discretization method**
  - Problem domain discretization
  - Discretize the equation system
- **Select suitable solution algorithms/techniques**
  - SIMPLE, PISO, PRESTO etc.
  - Cholesky, LU-decomposition, (Bi-)Conjugate Gradient, Geometric-Algebraic Multi-Grid
- **Analyze and validate results**

Simulation Software

IV. OpenFOAM

OpenFOAM® is a modern open source object-oriented library for the computational continuum mechanics in C++

- **Licensed under GNU GPL**
  - It is free to run, study, share, and modify
  - Vibrant community around OpenFOAM Extend project
  - It comes without any warranty
  - You have to provide source code to customer
- **Many basic solvers, and applications included**

IV. OpenFOAM (2)
Main Features

- **Collocated polyhedral unstructured meshes**
- **Finite Volume Method as a native discretization**
  - Second order in space and time
- **Many discretization schemes including higher order ones**
- **Algorithms for pressure-velocity coupling**
- **Mesh generation and manipulation tools**
- **Mesh import for all major mesh generators and CAD systems**
- **Physics implemented through an equation mimicking**

IV. OpenFOAM (3)
Equation Mimicking

- **Continuum Mechanics**

$$\frac{\partial k}{\partial t} + \nabla\bullet \mathbf{u}k = \nabla\bullet \left[(\nu + \nu_t)\nabla k\right] + \nu_t \left[\frac{1}{2}(\nabla\mathbf{u} + \nabla\mathbf{u}^T)\right]^2 - \frac{\epsilon_0}{k_0}k$$

- **OpenFOAM**

```
   fvm::ddt(k)
+  fvm::div(phi, k)
   ==
   fvm::laplacian(
     nu()+nut, k)
+  nut*magSqr(symm(
     fvc::grad(U)))
-  fvm::Sp(
     epsilon/k, k)
```

IV. OpenFOAM (4)
Pre- and Post-Processing

- **OpenFOAM has limited pre-processing, but can import**
  - `blockMesh` utility for structured multi-block hexahedral meshes
  - `snappyHexMesh` mesh generator for hex-dominant meshes on arbitrary geometry
- **OpenFOAM has no post-processor on its own, but can export**
  - Open source multi-platform data analysis and visualization tool ParaView reads OpenFOAM simulation cases natively
    - Supports distributed computation for large data processing
    - Uses VTK, and Qt libraries; OpenGL 2

V. Commercial Software vs OpenFOAM
Lecturer's View

Category	Commercial Software	OpenFOAM
Costs	−	+
Comfort of Use	+	−
Stability	+	o
Documentation	+	−
Solution Control	o	+
Extensibility	o	+
Disclosure of Numerics	−	+

Wallet

User

Power User

Research

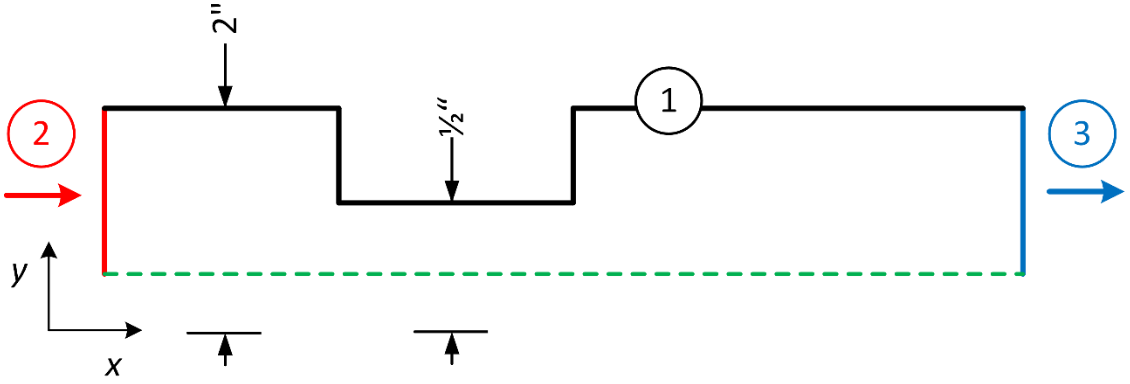
Education

What's the price?

VI. Example: Choke

DEPARTMENT PETROLEUM ENGINEERING

- **Compressible air flow through ½" choke**



$$
\begin{array}{llll}
① : & \nabla p = 0 & \mathbf{u} = 0 & \nabla T = 0 \\
② : & p = 1000\text{psi} & \nabla \mathbf{u} = 0 & T = 293\text{K} \\
③ : & p = 900\text{psi} & \nabla \mathbf{u} = 0 & \nabla T = 0
\end{array}
$$

MONTAN UNIVERSITÄT

VI. Example: Choke (2)
Compressible Navier-Stokes Equations

DEPARTMENT PETROLEUM ENGINEERING

$$1: \frac{\partial \rho}{\partial t} + \nabla \bullet (\rho \mathbf{u}) = 0$$

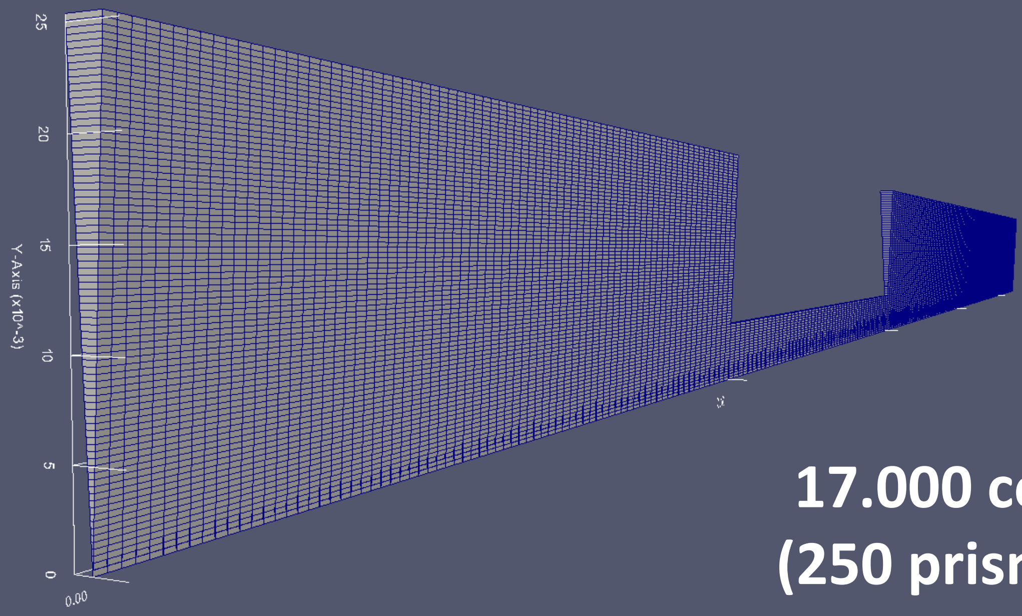
$$2: \frac{D}{Dt}(\rho \mathbf{u}) = -\nabla \left(p + \frac{2}{3}\mu \nabla \bullet \mathbf{u} \right) + \nabla \bullet \left(\mu \left[\nabla \mathbf{u} + (\nabla \mathbf{u})^T \right] \right)$$

$$3: \frac{\partial}{\partial t}(\rho h) + \nabla \bullet (\rho \mathbf{u} h) = \underbrace{\frac{\partial p}{\partial t} + \nabla \bullet (\mathbf{u} p)}_{\frac{D}{Dt} p} - p \nabla \bullet \mathbf{u} + \nabla \bullet (\alpha \nabla h)$$

$$4: \rho = \rho(p, T) \quad 5: \mu = \mu(p, T),\ \alpha = \alpha(p, T),\ \ldots$$

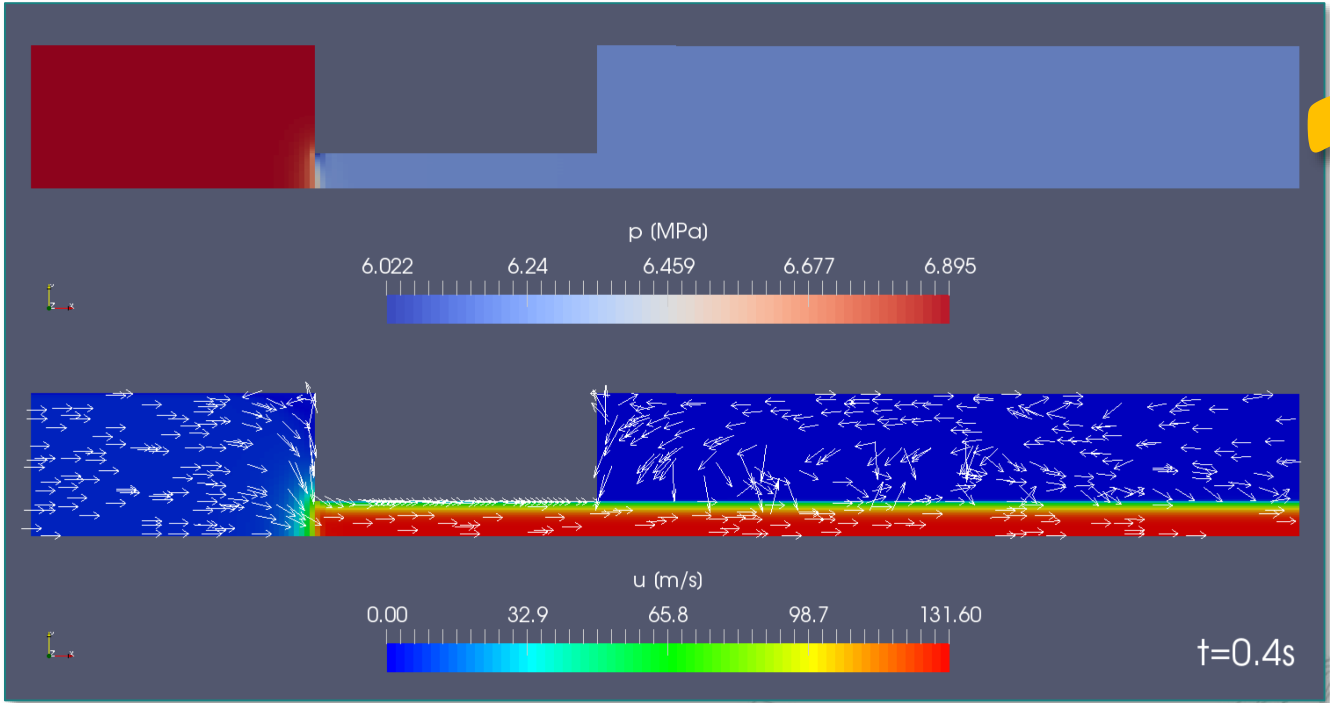
MONTAN UNIVERSITÄT

VI. Example: Choke (3)
2D Axial-Symmetric Mesh

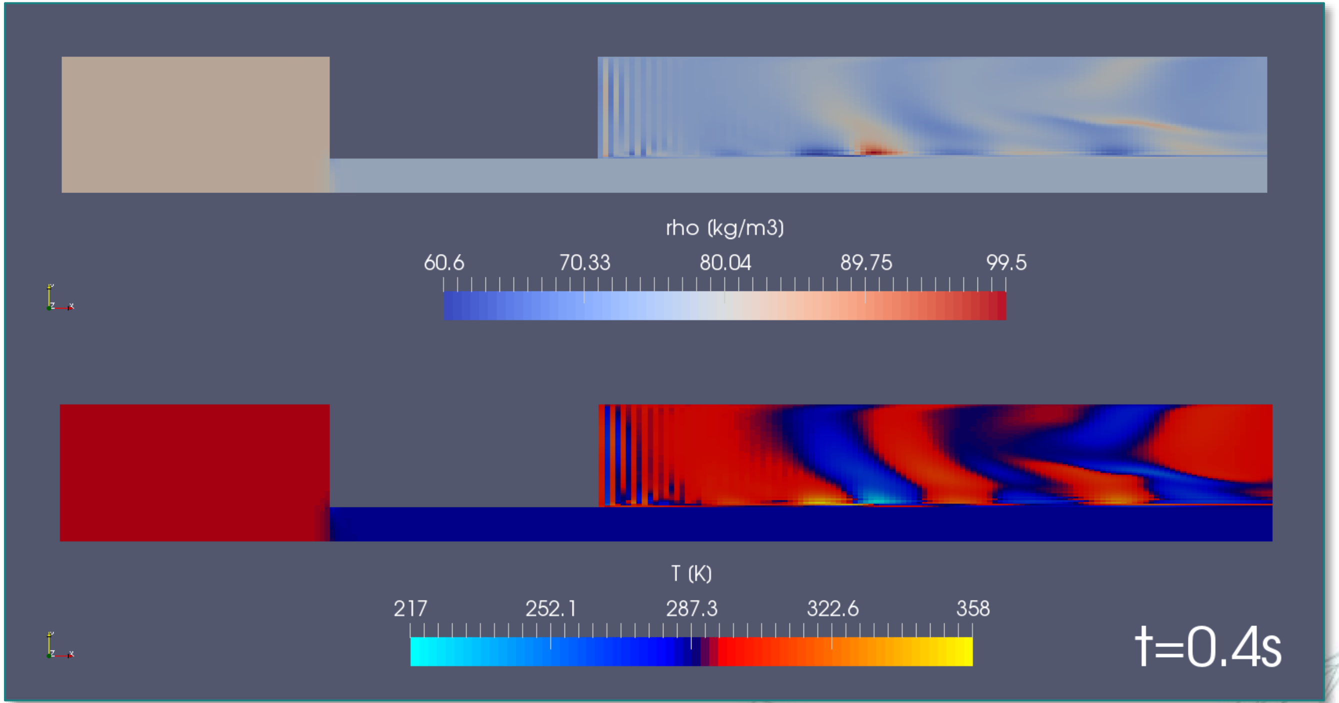


17.000 cells
(250 prisms)

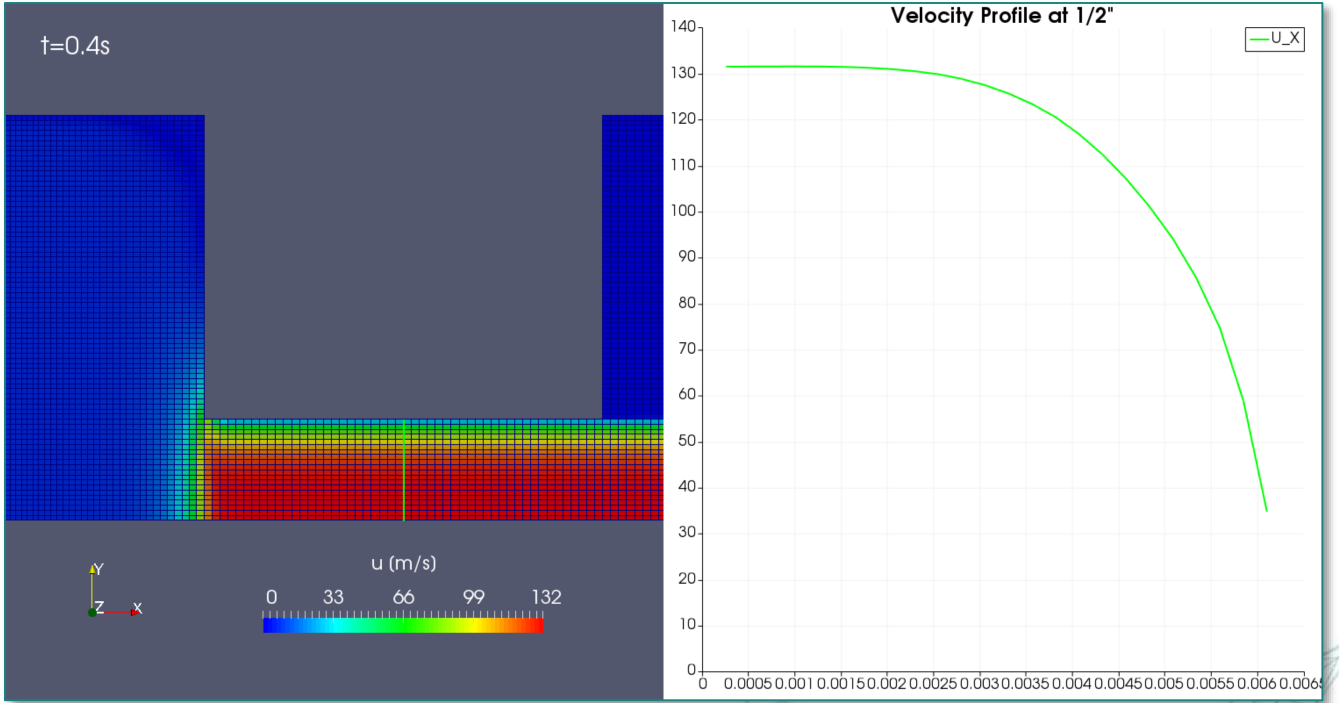
VI. Example: Choke (4)
Transient Compressible Laminar Air Flow



VI. Example: Choke (5)
Problems with Energy



VI. Example: Choke (6)
Realizing the Turbulence



VI. Example: Choke (7)
Standard *k-ε* Turbulence Model

- **Eddy viscosity**

$$\mu_\mathrm{t} = C_\mu \frac{k^2}{\varepsilon}, C_\mu = 0.09$$

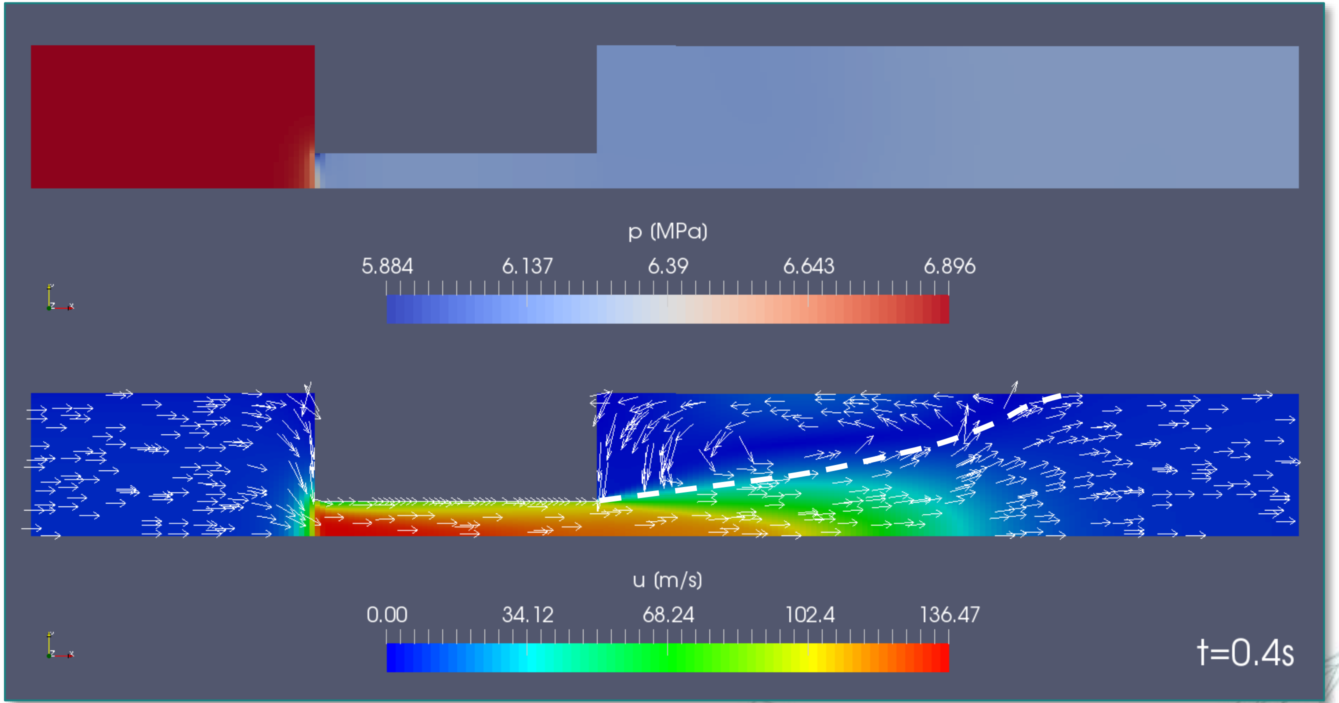
- ***k*- and *ε*-equation**

$$1: \frac{\partial}{\partial t}(\overline{\rho} k) + \nabla\bullet(\overline{\rho \mathbf{u}} k) = \nabla\bullet\left(\frac{\mu_\mathrm{t}}{\sigma_k}\nabla k\right) + P_k - \overline{\rho}\varepsilon + S_k$$

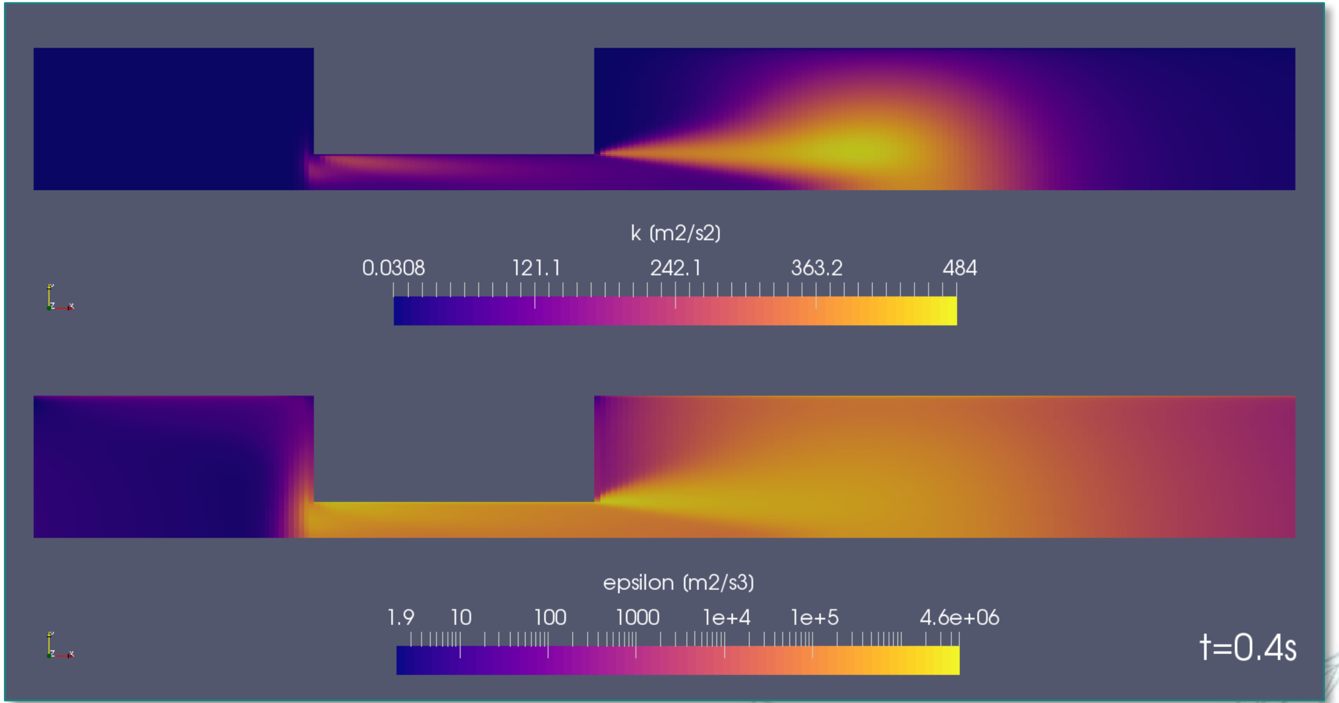
$$2: \frac{\partial}{\partial t}(\overline{\rho} \varepsilon) + \nabla\bullet(\overline{\rho \mathbf{u}} \varepsilon) = \nabla\bullet\left(\frac{\mu_\mathrm{t}}{\sigma_\varepsilon}\nabla \varepsilon\right) + C_{1\varepsilon}\frac{\varepsilon}{k}P_k - C_{2\varepsilon}\overline{\rho}\frac{\varepsilon^2}{k} + S_\varepsilon$$

$$3: P_k = 2\mu_t \epsilon_{ij}\epsilon_{ij}, C_{1\varepsilon} = 1.44, C_{2\varepsilon} = 1.92, \sigma_k = 1.0, \sigma_\varepsilon = 1.3$$

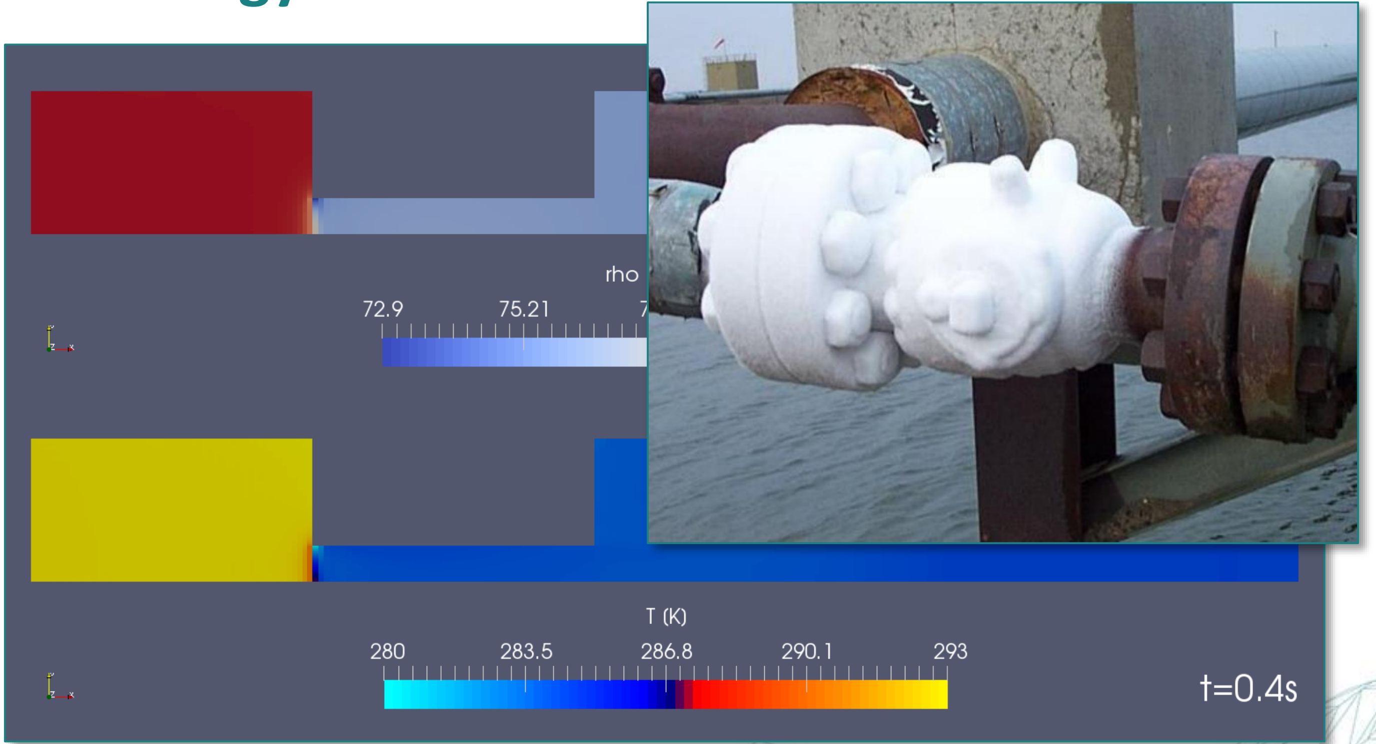
VI. Example: Choke (8)
Transient Compressible Turbulent Air Flow



VI. Example: Choke (9)
Turbulent Kinetic Energy, and Dissipation



VI. Example: Choke (10)
What about Energy?



VII. Discussion

DEPARTMENT PETROLEUM ENGINEERING

MONTAN UNIVERSITÄT