TAMPERE UNIVERSITY OF TECHNOLOGY

KEB-45250
Numerical Techniques for Process Modeling

Spring 2018

Lecture 3

Introduction to Computational Fluid Dynamics

Antti Mikkonen

Last lecture

- Review of relevant subjects
- Math
  - Equation types, matrices
- Physics
  - Advection, Convection, Diffusion, Navier-Stokes
  - Dimensionless numbers
- Chemistry
  - Arrhenius equation

Lectures		Exercises
Introduction		Python basics and libraries
Basics. Matrix, NS,…		Lecture topic
CFD Basics		Lecture topic
ANSYS intensive course		ANSYS intensive course
Heat convection, FVM		Lecture topic with Python
Advection		Lecture topic with Python
Navier-Stokes		Navier-Stokes with ANSYS
Mesh		Mesh with ANSYS
Turbulence		Turbulence with ANSYS
Differential equations		Lecture topic
Linear systems		Lecture topic
Easter Holiday		Easter Holiday
Linear/Non-linear systems		Lecture topic
Non-linear systems		Lecture topic
Reacting systems		Lecture topic
Reacting systems		Lecture topic

This lecture

- Introduction Computational Fluid Dynamics (CFD)
- What is CFD?
- Relevant math
- Preprocessing

Lectures	Exercises
Introduction	Python basics and libraries
Basics. Matrix, NS,…	Lecture topic
CFD Basics	Lecture topic
ANSYS intensive course	ANSYS intensive course
Heat convection, FVM	Lecture topic with Python
Advection	Lecture topic with Python
Navier-Stokes	Navier-Stokes with ANSYS
Mesh	Mesh with ANSYS
Turbulence	Turbulence with ANSYS
Differential equations	Lecture topic
Linear systems	Lecture topic
Easter Holiday	Easter Holiday
Linear/Non-linear systems	Lecture topic
Non-linear systems	Lecture topic
Reacting systems	Lecture topic
Reacting systems	Lecture topic

Near future

- ANSYS intensive course next week
  - **Tuesday** 9-16 SB202
  - **Wednesday** 9-15 RG100C
  - NOTE THE NONSTANDARD DAYS

Lectures	Exercises
Introduction	Python basics and libraries
Basics. Matrix, NS,…	Lecture topic
CFD Basics	Lecture topic
ANSYS intensive course	ANSYS intensive course
Heat convection, FVM	Lecture topic with Python
Advection	Lecture topic with Python
Navier-Stokes	Navier-Stokes with ANSYS
Mesh	Mesh with ANSYS
Turbulence	Turbulence with ANSYS
Differential equations	Lecture topic
Linear systems	Lecture topic
Easter Holiday	Easter Holiday
Linear/Non-linear systems	Lecture topic
Non-linear systems	Lecture topic
Reacting systems	Lecture topic
Reacting systems	Lecture topic

Computational Fluid Dynamics

- Fluid flow calculation is the basis of all CFD simulations
- Other physics and chemistry can be included
- Navier-Stokes equations govern fluid flow phenomena

Navier-Stokes

https://www.youtube.com/watch?v=qgKVxOWpi9U

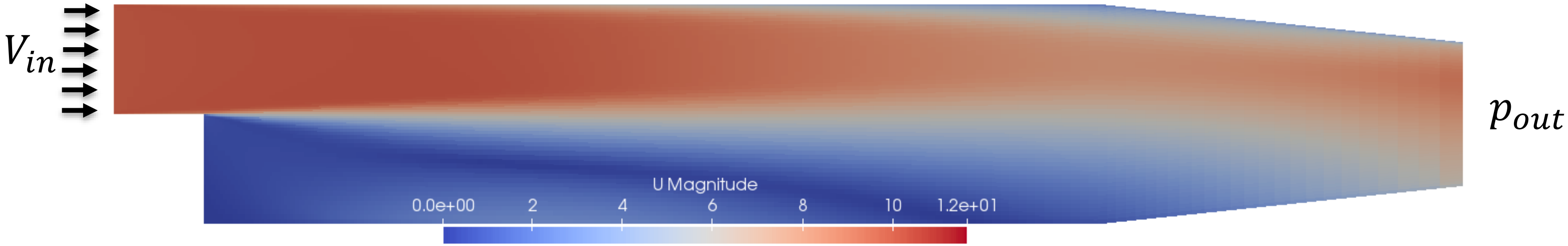
- Incompressible flow, x-direction
  - $$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + w\frac{\partial u}{\partial z} = -\frac{1}{\rho}\frac{\partial p}{\partial x} + \nu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}\right)$$
- All directions
  - $$\frac{\partial u_i}{\partial t} + u_j\frac{\partial u_i}{\partial x_j} = -\frac{1}{\rho}\frac{\partial p}{\partial x_i} + \nu\left(\frac{\partial^2 u_i}{\partial x_j^2}\right)$$
- Today, we only discuss the equations in a general sense, details come later

Scope of this course

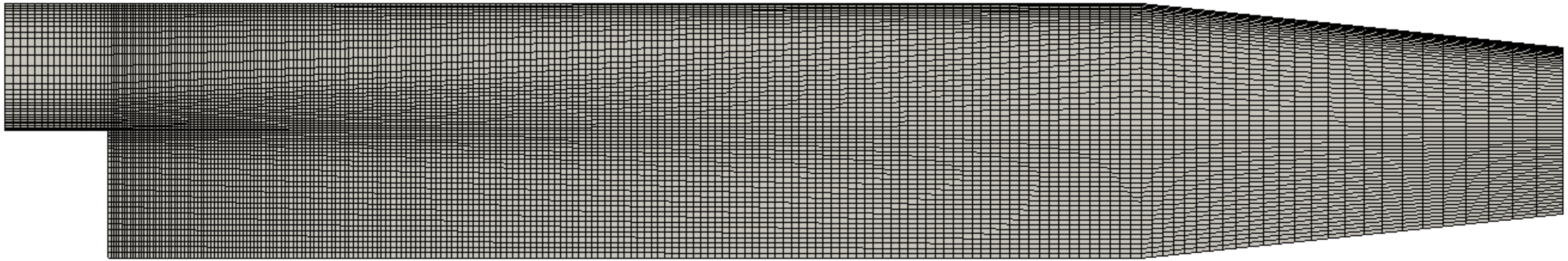
- Solution of NS is difficult
- We will not program our own NS solver on this course
  - Unless you want to learn deeper understanding of CFD and program your own solver as the CFD assignment
  - The case is chosen so that is relatively easy to solve
- We will program our own solver for simpler cases (heat conduction etc.)
- We will use ANSYS for NS
  - You still need to understand the basics
  - You can do the CFD assignment with ANSYS

Example case

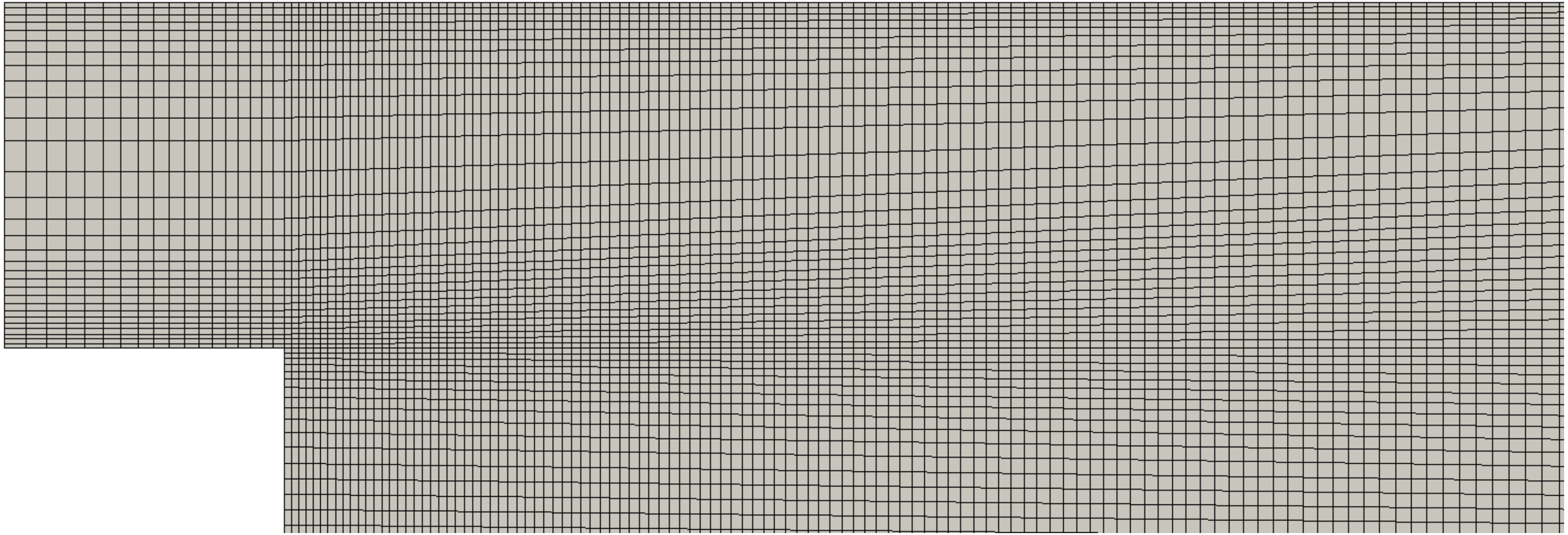
- Classical test case by Pitz and Daily (1981)



Mesh



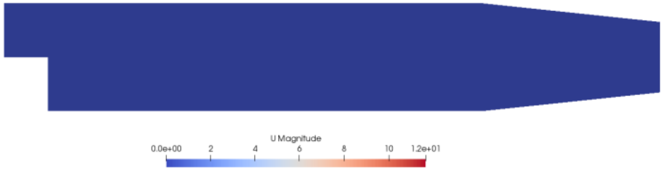
Mesh close up



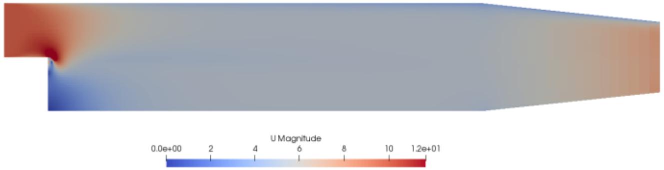
Unsteady

$$\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} = -\frac{1}{\rho}\frac{\partial p}{\partial x_i} + \nu\left(\frac{\partial^2 u_i}{\partial x_j^2}\right)$$

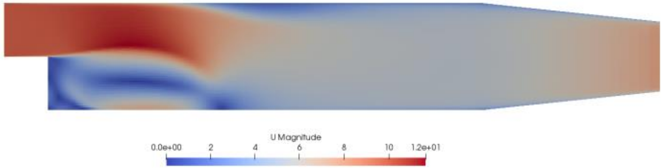
$t = 0s$



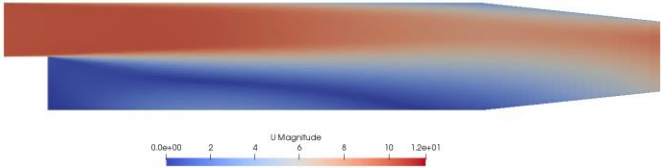
$t = 0.001s$



$t = 0.03s$



$t = 0.3s$

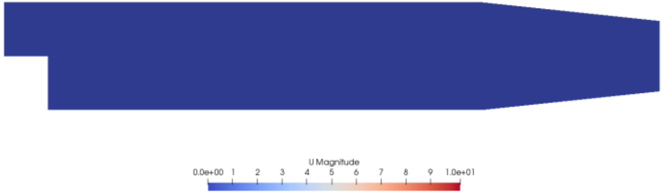


Steady state

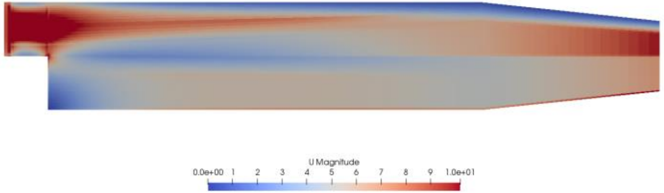
$$\cancelto{0}{\frac{\partial u_i}{\partial t}} + u_j \frac{\partial u_i}{\partial x_j} = -\frac{1}{\rho}\frac{\partial p}{\partial x_i} + \nu \left(\frac{\partial^2 u_i}{\partial x_j^2} \right)$$

k is iteration step

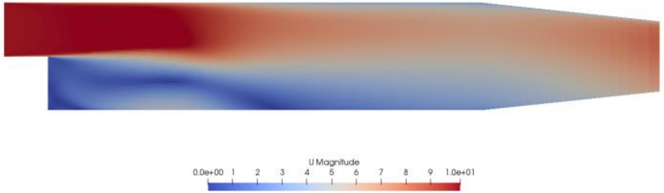
$k = 0$



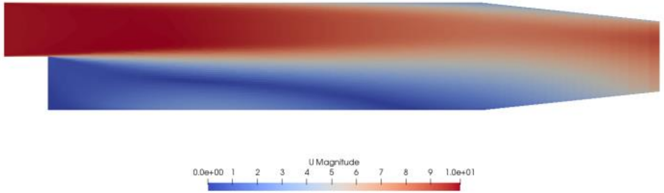
$k = 1$



$k = 70$



$k = 287$



Not logged in Talk Contributions Create account Log in

Article Talk Read Edit View history Search Wikipedia

WIKIPEDIA
The Free Encyclopedia

Main page
Contents
Featured content
Current events
Random article
Donate to Wikipedia
Wikipedia store

Interaction

Help
About Wikipedia
Community portal
Recent changes
Contact page

Tools

What links here
Related changes
Upload file
Special pages
Permanent link
Page information
Wikidata item
Cite this page

Print/export

Create a book
Download as PDF
Printable version

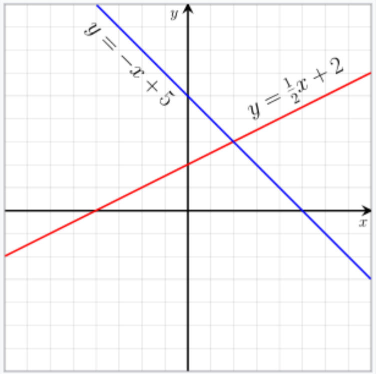
Linear equation

From Wikipedia, the free encyclopedia

This article **relies largely or entirely on a single source**. Relevant discussion may be found on the talk page. Please help improve this article by introducing citations to additional sources. *(January 2016)*

This article **needs additional citations for verification**. Please help improve this article by adding citations to reliable sources. Unsourced material may be challenged and removed. *(January 2016) (Learn how and when to remove this template message)*

A **linear equation** is an algebraic equation in which each term is either a constant or the product of a constant and (the first power of) a single variable (however, different variables may occur in different terms). A simple example of a linear equation with only one variable, x, may be written in the form: $ax + b = 0$, where a and b are constants and $a \neq 0$. The constants may be numbers, parameters, or even non-linear functions of parameters, and the distinction between variables and parameters may depend on the problem (for an example, see linear regression).



Graph sample of linear equations.

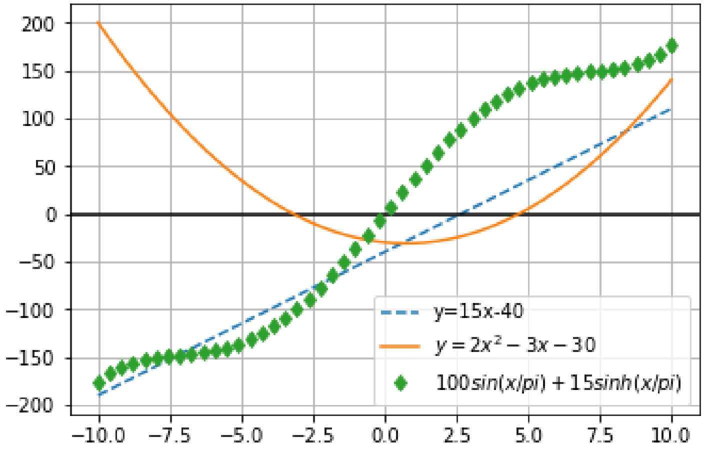
Linear equations can have one or more variables. An example of a linear equation with three variables, x, y, and z, is given by: $ax + by + cz + d = 0$, where a, b, c, and d are constants and a, b, and c are non-zero. Linear equations occur frequently in most subareas of mathematics and especially in applied mathematics. While they arise quite naturally when modeling many phenomena, they are particularly useful since many non-linear equations may be reduced to linear equations by assuming that quantities of interest vary to only a small extent from some "background" state. An equation is linear if the sum of the exponents of the variables of each term is one.

Equations with exponents greater than one are non-linear. An example of a non-linear equation of two variables is $axy + b = 0$, where a and b are constants and $a \neq 0$. It has two variables, x and y, and is non-linear because the sum of the exponents of the variables in the first term, axy, is two.

This article considers the case of a single equation for which one searches the real solutions. All its content applies for complex solutions and, more generally for linear equations with coefficients and solutions in any field.

Linear vs. nonlinear

- …all else is nonlinear
- Nonlinear equations don’t always have analytical solutions
- May be difficult to solve numerically
- The same ideas can be extended to systems of equations



Not logged in Talk Contributions Create account Log in

Article Talk Read Edit View history Search Wikipedia

Main page
Contents
Featured content
Current events
Random article
Donate to Wikipedia
Wikipedia store

Interaction

Help
About Wikipedia
Community portal
Recent changes
Contact page

Tools

What links here
Related changes
Upload file
Special pages
Permanent link

System of linear equations

From Wikipedia, the free encyclopedia

This article includes a list of references, but **its sources remain unclear** because it has **insufficient inline citations**. Please help to improve this article by introducing more precise citations. *(October 2015)* *(Learn how and when to remove this template message)*

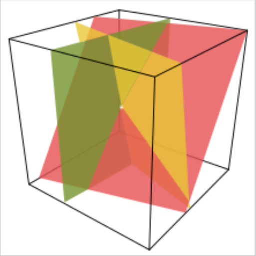
In mathematics, a **system of linear equations** (or **linear system**) is a collection of two or more linear equations involving the same set of variables.[1] For example,

$$\begin{aligned} 3x + 2y - z &= 1 \\ 2x - 2y + 4z &= -2 \\ -x + \tfrac{1}{2}y - z &= 0 \end{aligned}$$

is a system of three equations in the three variables x, y, z. A **solution** to a linear system is an assignment of values to the variables such that all the equations are simultaneously satisfied. A solution to the system above is given by

$$\begin{aligned} x &= 1 \\ y &= -2 \\ z &= -2 \end{aligned}$$

since it makes all three equations valid. The word "*system*" indicates that the equations are to be considered collectively, rather than individually.



A linear system in three variables determines a collection of planes. The intersection point is the solution.

$$\begin{bmatrix} 3 & 2 & -1 \\ 2 & -2 & 4 \\ -1 & 1/2 & -1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 1 \\ -2 \\ 0 \end{bmatrix} \Leftrightarrow \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 1 \\ -2 \\ -2 \end{bmatrix}$$

Why are we talking about matrices?

- The governing equations of fluid flow can be approximated with systems of algebraic equations
- Linear matrix systems are easily solved with computer algorithms
- Understanding the methods of how to do this is understanding CFD

Nonlinear system

$$\begin{aligned} 3xz + 2y - z &= 1 \\ 2x - 2y - 4 &= -2 \\ -x + 0.5y - z &= 0 \end{aligned}$$

Or in matrix form

$$\begin{bmatrix} 3z & 2 & -1 \\ 2 & -2 & 4 \\ -1 & 0.5 & -1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 1 \\ -2 \\ 0 \end{bmatrix}$$

Iterative solution

- Example non-linear equation
  - $-\frac{1}{\sqrt{f}} = -2\log_{10}\left(\frac{\epsilon}{3.7d_h} + \frac{2.51}{Re\sqrt{f}}\right)$
- A naive iteration example on the right
- In CFD we use case specific methods

```
Re  = 10000
d   = 1e-2
eps = 0

f = 1e-2
n = 10
for k in range(n):
    print(f)
    f = (-2*sp.log10(eps/3.7/d
          +2.51/Re*f**0.5)
         )**-2
```

Out:

```
0.01
0.0118130689375
0.0120011028994
0.0120191569871
0.0120208776662
0.0120210415431
0.0120210571496
0.0120210586359
0.0120210587774
0.0120210587909
```

Nonlinear system solution

- Naive iteration for matrix system
$$\begin{bmatrix} 3z & 2 & -1 \\ 2 & -2 & 4 \\ -1 & 0.5 & -1 \end{bmatrix}\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 1 \\ -2 \\ 0 \end{bmatrix}$$
- Gives x = [1 4 1]
- Error

  A**x**-**b**=[0 0 -2e-16]

```python
b = sp.array([1,-2,0])
x = sp.array([2,1,1])

n = 100
for k in range(n):
    A = sp.array([[3*x[2],2,-1],
                  [2,-2, 4],
                  [-1,0.5,-1],
                  ])
    x = linalg.solve(A,b)

# Print solution
print(x)
# Test solution, @ is matrix multiplication
print(A@x-b)
```

Disclaimer

- It might be difficult to find the answer
- Better methods for general nonlinear equations can be found from nearly any programming language
- The naive method is used here for simplicity
- In CFD, case special methods are used

Nonlinear PDE

- In CFD, we mostly deal with nonlinear partial differential equations

- Steady state incompressible flow N-S, x-direction

$$u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + w\frac{\partial u}{\partial z} = -\frac{1}{\rho}\frac{\partial p}{\partial x} + \nu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}\right)$$

- All directions

$$u_j\frac{\partial u_i}{\partial x_j} = -\frac{1}{\rho}\frac{\partial p}{\partial x_i} + \nu\left(\frac{\partial^2 u_i}{\partial x_j^2}\right)$$

Solution strategy

- Linearize the PDE
- Approximate PDE with a system of algebraic equations
- Solve iteratively

Simple linearization

- Original equation

$$u_j \frac{\partial u_i}{\partial x_j} = -\frac{1}{\rho}\frac{\partial p}{\partial x_i} + \nu\left(\frac{\partial^2 u_i}{\partial x_j^2}\right)$$

- Linearized equation

$$u_j^* \frac{\partial u_i}{\partial x_j} = -\frac{1}{\rho}\frac{\partial p}{\partial x_i} + \nu\left(\frac{\partial^2 u_i}{\partial x_j^2}\right)$$

where u_j^* is velocity from previous iteration

Iterative solution of NS

$$u_j^* \frac{\partial u_i}{\partial x_j} = -\frac{1}{\rho}\frac{\partial p}{\partial x_i} + \nu\left(\frac{\partial^2 u_i}{\partial x_j^2}\right)$$

- The above equation is approximated using a system of algebraic equations

$$A\boldsymbol{x} = \boldsymbol{b}$$

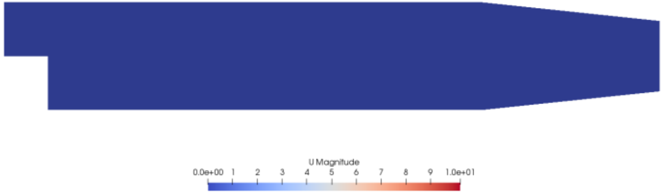
using methods we will learn later on this course. The system is solved iteratively

Iterations

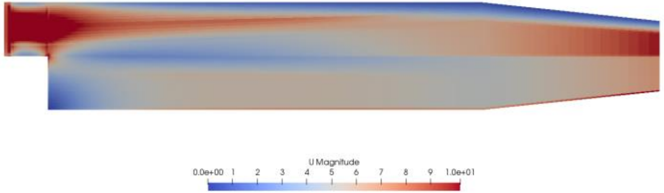
$$u_j^* \frac{\partial u_i}{\partial x_j} = -\frac{1}{\rho}\frac{\partial p}{\partial x_i} + \nu\left(\frac{\partial^2 u_i}{\partial x_j^2}\right)$$

k is iteration step

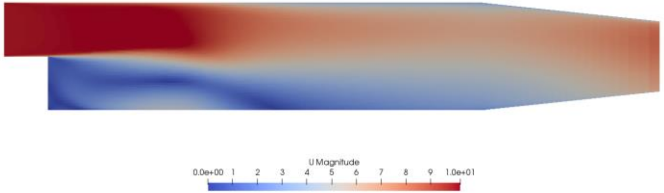
$k = 0$



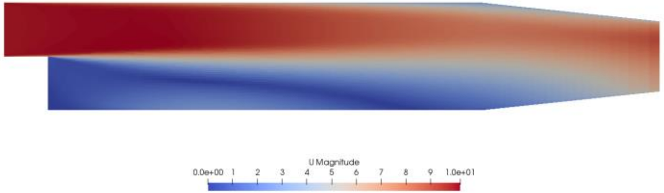
$k = 1$



$k = 70$

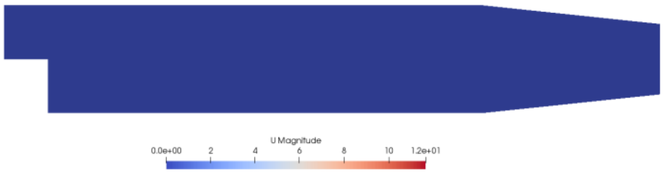


$k = 287$

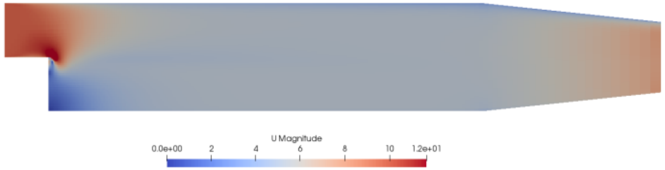


Development in time

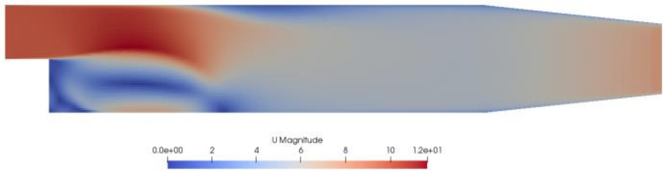
$t = 0s$



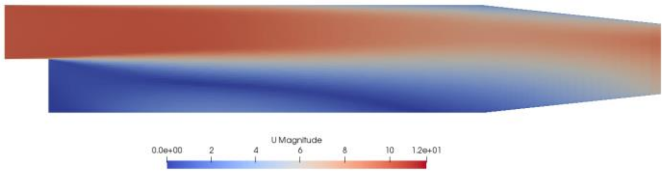
$t = 0.001s$



$t = 0.03s$

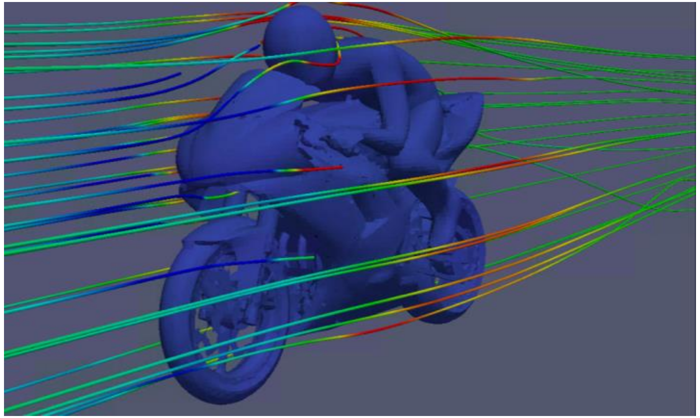


$t = 0.3s$

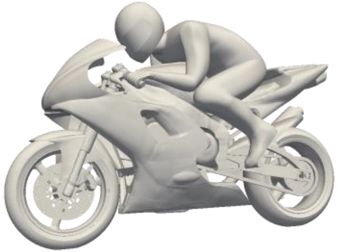


NEXT TOPIC

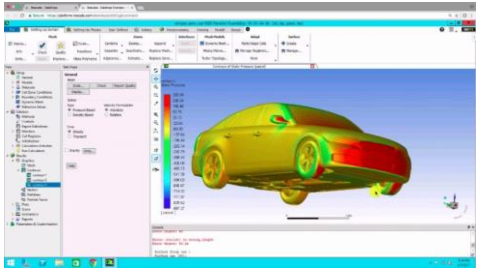
Steps of CFD solution

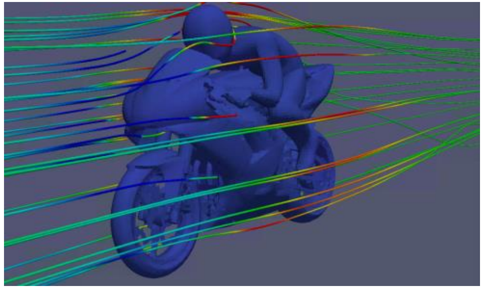


Steps of CFD solution

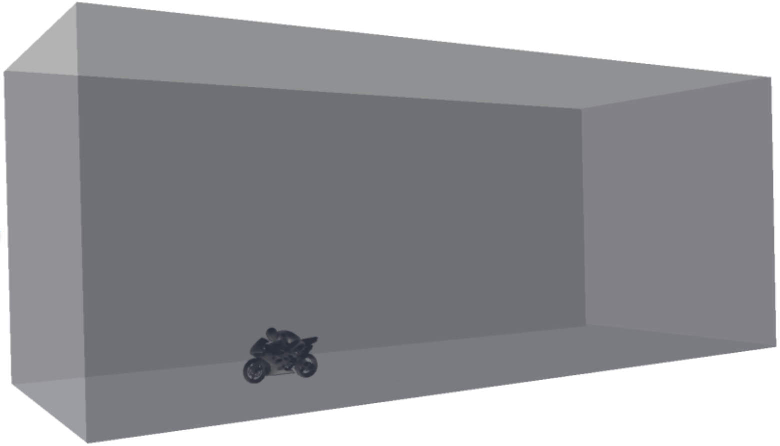


1. Preprocessing
2. Solver
3. Post processing





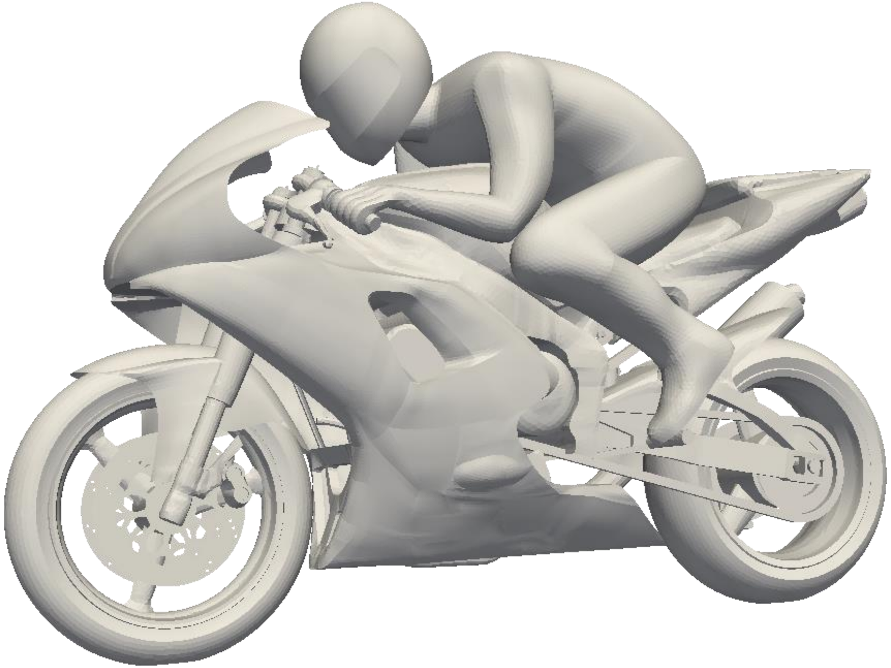
Preprocesing



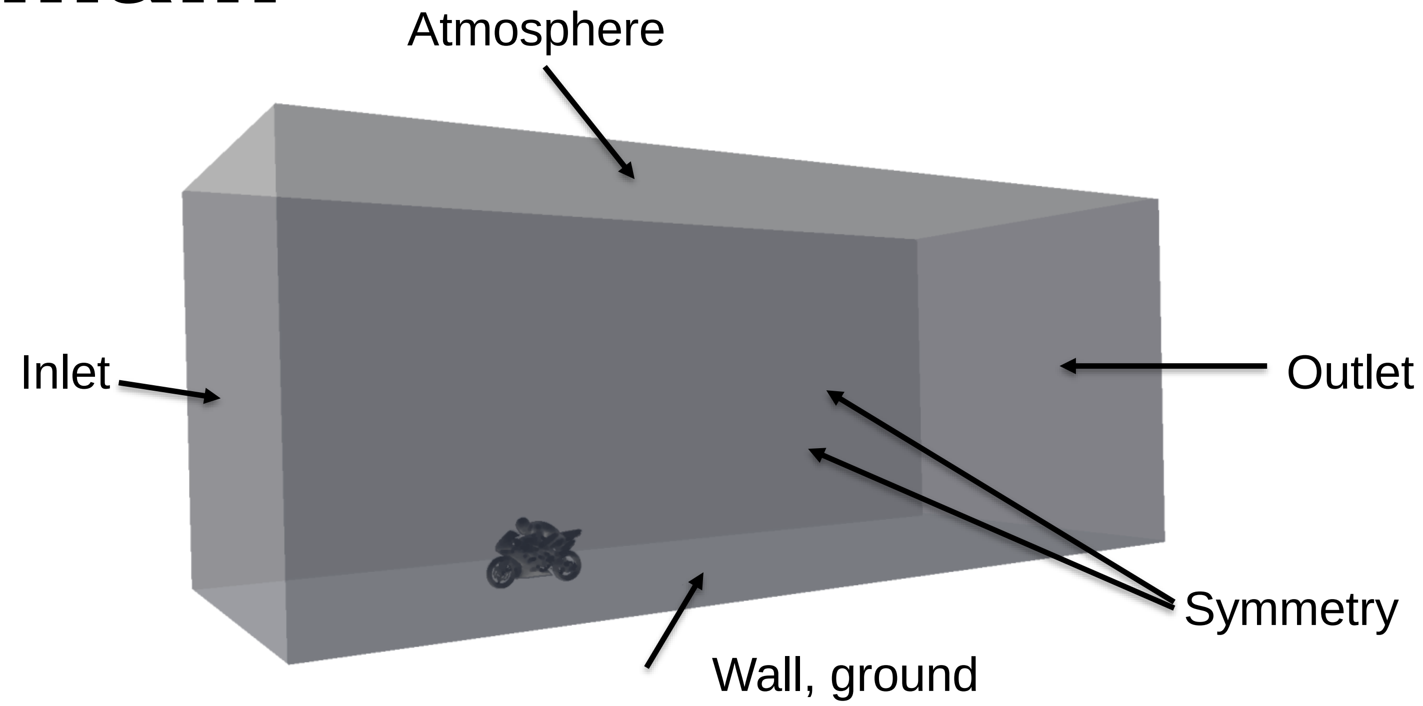
Preprocessing

- CAD
  - Geometry with suitable detail
  - Draw by yourself or prepare existing CAD for CFD
- Meshing
  - Creating the mesh for CFD solution
  - Often the most time consuming part

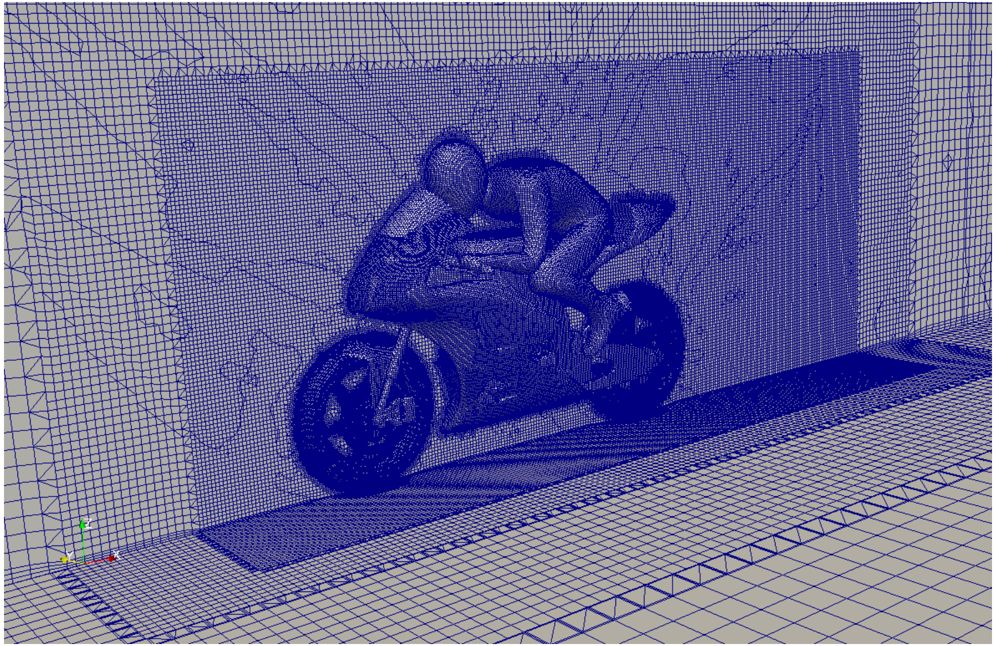
Suitable level of detail



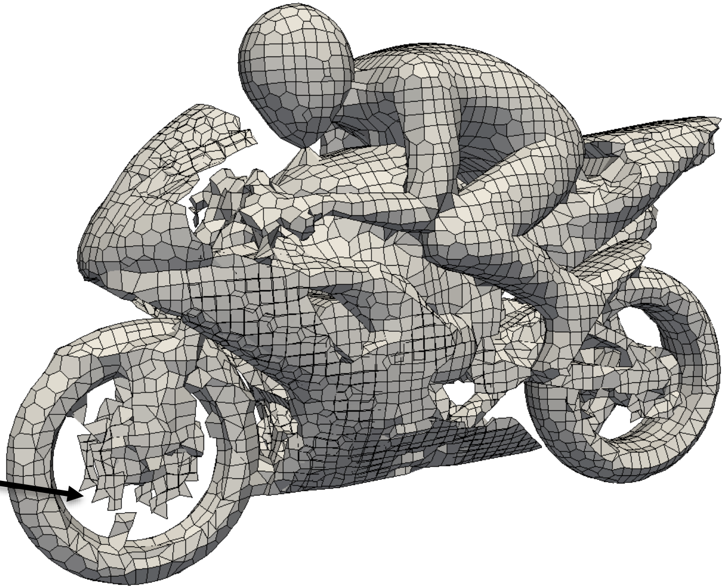
Domain



Meshing

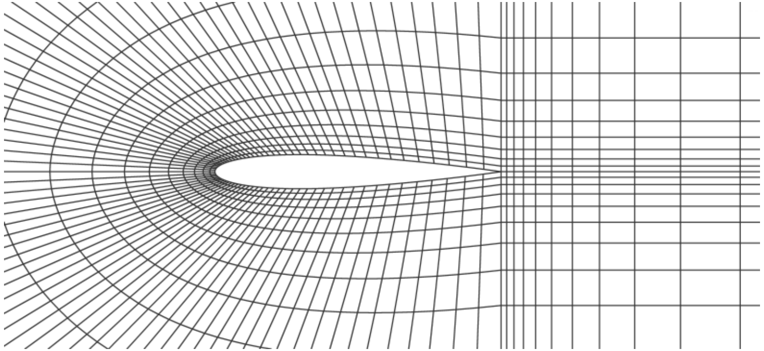


Surface details

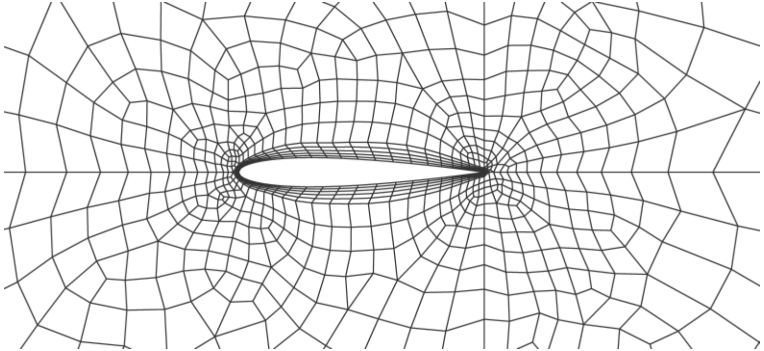


Insufficient level of detail

Mesh types

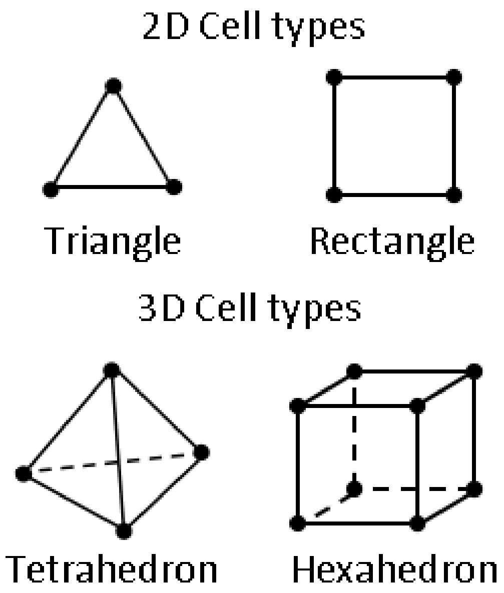


Structured mesh



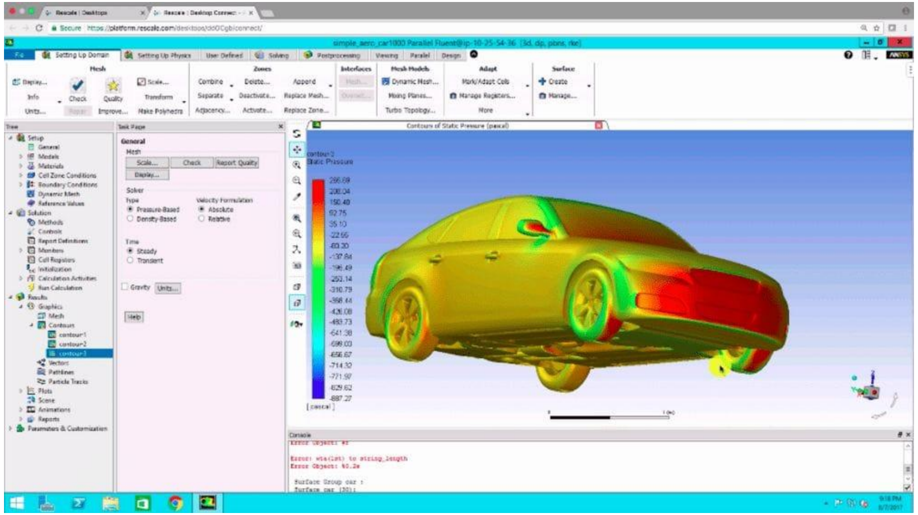
Unstructured mesh

Cell types


2D Cell types
Triangle
Rectangle
3D Cell types
Tetrahedron
Hexahedron

Solver

- Governing equations
- Boundary conditions
- Numerical methods
- Turbulence model
- Material properties
- Saving of results



Post processing

- Is the result reasonable?
- Is the solution converged?
- Is the solution grid independent?
- How to present the results
  - Mean heat transfer coefficient vs. video

Time derivative term

$$\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} = -\frac{1}{\rho}\frac{\partial p}{\partial x_i} + \nu\left(\frac{\partial^2 u_i}{\partial x_j^2}\right)$$

Explicit time stepping

- From Taylor expansion

$$\underbrace{f(t_0 + \Delta t) = f(t_0) + \Delta t f'(t_0)}_{\text{Explicit Euler}} + \cancelto{\text{small}}{\frac{1}{2}\Delta t^2 f''(t_0)} + \cdots$$

- Simple to implement but rarely used
- Easily becomes unstable

Implicit time stepping

- The function values are taken “from the future”

$$f(t_0 + \Delta t) = f(t_0) + \Delta t f'(t_0 + \Delta t)$$

- Might require iteration because of $f'(t_0 + \Delta t)$
- Commonly used in CFD
- Stable and simple to implement but inaccurate
- If possible, use Runge-Kutta. Rarely used in CFD.

Turbulence

More physics and chemistry

Algebraic equations

- These are the most common equations we usually deal with (analytical solutions for specific problems in our course context)
- The value of the variable we want to solve can be determined directly from a specific equation using for example a calculator
- Some examples:

$$\Delta p = \frac{1}{2}\rho V^2 \xi \frac{L}{D}$$

$$Nu_x = 0.332 Re_x^{1/2} Pr^{1/3}$$

Linear system of equations

- The solution for the system can be obtained with the inverse of matrix *A*:

$$\boldsymbol{x} = A^{-1}\boldsymbol{b}, \text{ where}$$

A^{-1} is the inverse of the coefficient matrix *A*

Advection and diffusion terms in PDEs

Momentum conservation of fluid in x-direction

$$\frac{\partial u}{\partial t} + \underbrace{u\frac{\partial u}{\partial x} + \mathrm{v}\frac{\partial u}{\partial y} + w\frac{\partial u}{\partial z}}_{\text{Advection terms}} = -\frac{1}{\rho}\frac{\partial p}{\partial x} + \underbrace{\nu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}\right)}_{\text{Diffusion terms}}$$

Heat conduction in solids

$$\frac{\partial T}{\partial t} = \underbrace{\alpha\left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2}\right)}_{\text{Diffusion terms}}$$

Fluid flow equations

- Navier-Stokes equations:

$$\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial x}+\mathrm{v}\frac{\partial u}{\partial y}+w\frac{\partial u}{\partial z}=-\frac{1}{\rho}\frac{\partial p}{\partial x}+\nu\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}+\frac{\partial^2 u}{\partial z^2}\right)$$
$$\frac{\partial v}{\partial t}+u\frac{\partial v}{\partial x}+\mathrm{v}\frac{\partial v}{\partial y}+w\frac{\partial v}{\partial z}=-\frac{1}{\rho}\frac{\partial p}{\partial y}+\nu\left(\frac{\partial^2 v}{\partial x^2}+\frac{\partial^2 v}{\partial y^2}+\frac{\partial^2 v}{\partial z^2}\right)$$
$$\frac{\partial w}{\partial t}+u\frac{\partial w}{\partial x}+\mathrm{v}\frac{\partial w}{\partial y}+w\frac{\partial w}{\partial z}=-\frac{1}{\rho}\frac{\partial p}{\partial z}+\nu\left(\frac{\partial^2 w}{\partial x^2}+\frac{\partial^2 w}{\partial y^2}+\frac{\partial^2 w}{\partial z^2}\right)$$

Fluid flow equations

- NS with tensor notation:

$$\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} = -\frac{1}{\rho}\frac{\partial p}{\partial x_i} + \nu\left(\frac{\partial^2 u_i}{\partial x_j^2}\right)$$

- NS in vector form:

$$\frac{\partial \boldsymbol{V}}{\partial t} + \boldsymbol{V}\cdot\nabla\boldsymbol{V} = -\frac{1}{\rho}\nabla p + \nu\nabla^2\boldsymbol{V}$$