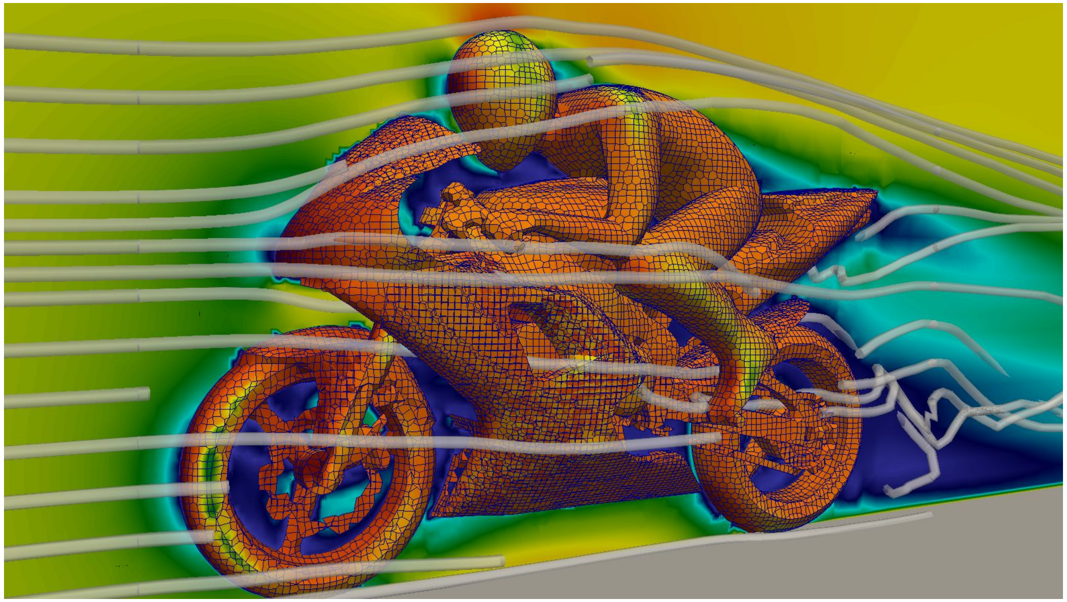
Open Source Computational Fluid Dynamics



An MSc course to gain extended knowledge in Computational Fluid Dynamics (CFD) using open source software.

Teachers: Miklós Balogh and Zoltán Hernádi

Department of Fluid Mechanics, Budapest University of Technology and Economics

M Ű E G Y E T E M 1 7 8 2

Course description

- Introduction to OpenFOAM simulations (and linux, gnuplot, paraview).
- Installation on several Linux distributions and virtual systems.
- Solution of simple fluid dynamics problems using OpenFOAM.
- Detailed introduction to OpenFOAM software components:
    - pre-processing (meshing tools and utilities)
    - solving (standard applications, user applications)
    - post-processing (ParaView).
- Single phase stationary and transient flows, turbulence, compressible flows.
- Multiphase and reactive flows.
- Extension of OpenFOAM capabilities by program code development in C++.
- Individual projects using OpenFOAM.
- Further open source CFD tools.

Course website

http://www.ara.bme.hu/~hernadi/OpenFOAM

Planned schedule

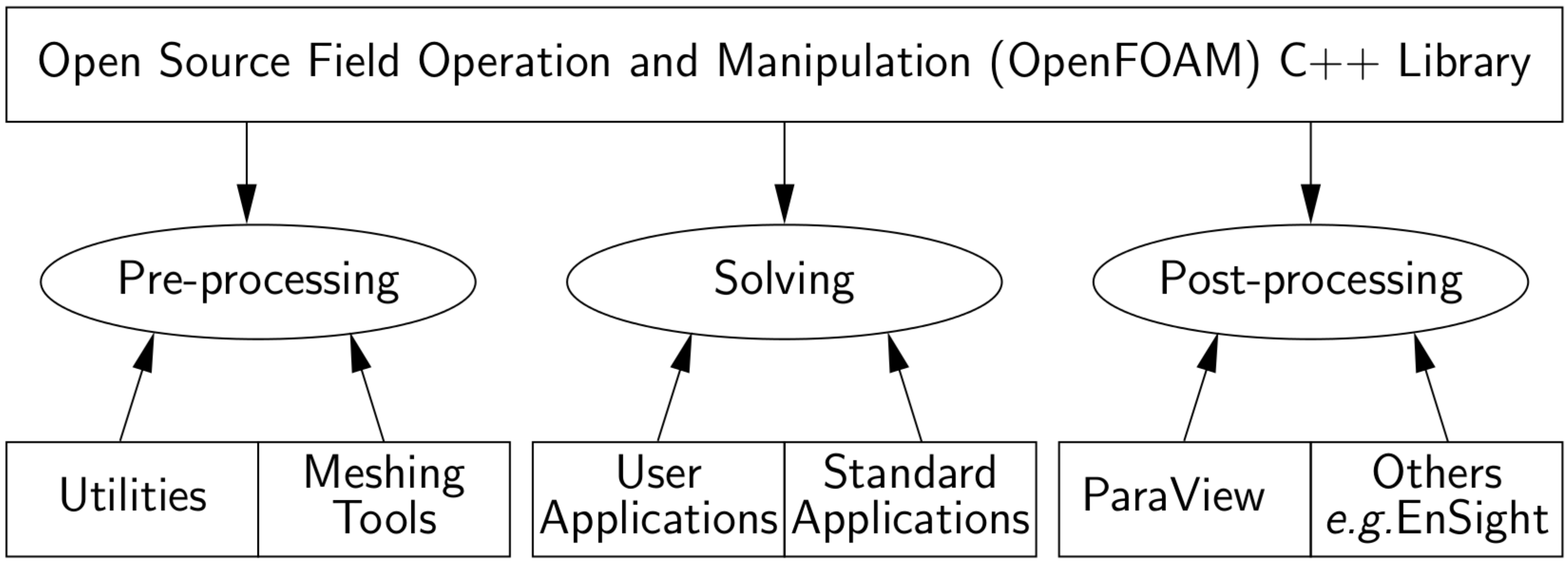
- **week 1:** introduction to OpenFOAM, installing
- **week 2:** solving simple fluid flow problems
- **week 3:** studying software components
- **week 4:** stationary and transient flows
- **week 5:** turbulent and compressible flows, **student project proposals deadline**
- **week 6:** multiphase and reactive flows, **final list of individual project proposals**
- **week 7:** OpenFOAM programming (C++), **individual project declarations**
- **week 8: mid-term exam**
- **week 9:** advanced programming tools
- **week 10:** other open-soruce tools
- **week 11:** (holiday)
- **week 12:** experiments and simulations
- **week 13-14: presentations of individual projects**

Final grade

The grading is based on 1 mid-term exam (50% in final grade) and an individual project (50% in final grade). In-class activity will be rewarded by bonus points.

Introduction to OpenFOAM

- **Open** source **F**ield **O**peration **A**nd **M**anipulation
- *C++ toolbox*, mainly designed for Computational Fluid Dynamics
- *Linux* is fully supported by OpenFOAM.



Linux operating system

- Linux comes in different flavours: **distributions**.
- Most popular Linux distributions are e.g. Arch Linux, CentOS, Debian, **Fedora**, Mageia, Linux Mint, **openSUSE**, PCLinuxOS, Slackware, **Ubuntu**.
- Different distributions have different desktop environments, but the linux **command-line** in very similar.
- The *command-line* can be accessed through a **terminal emulator**, e.g. gnome-terminal, LXTerminal, rxvt, xterm.

Commands for navigating in file system

ls	list directory contents
ls -l	(use a long listing format)
ls -ltr	(long listing, sort by time, reverse order)
ls --help	(to see other options)
man ls	manual of ls command (to exit press q)
cd ..	change directory to parent (relative to $PWD)
cd /bin	change directory to /bin (absolute position)
cd ~/OpenFOAM	change directory to $HOME/OpenFOAM
pwd	print name of current/working directory
echo HOME	display a line of text: HOME
echo $HOME	display a variable: $HOME
cd	change directory to $HOME
find -name OpenFOAM	find files or directories named OpenFOAM
man find	manual of find command (to exit press q)
which ls	locate a command

Commands for texts in files

cd $WM_PROJECT_DIR	change directory to OpenFOAM installation
cat Allwmake	print file content
less Allwmake	print file content and scroll (exit: q) (type `/source` to search "source") (type `n` for next, `N` for previous)
wc Allwmake	print newline, word, and byte counts
wc -w Allwmake	print word counts
man wc	manual for wc command
head Allwmake	output first part of file
tail Allwmake	output last part of file
grep echo Allwmake	print lines matching a pattern
sort Allwmake	sort lines of text
sort -r Allwmake	sort lines of text, reverse order
tac Allwmake	print file content, reverse order
grep -ri piso src	find files with content (-i: ignore case)
sort Allwmake | uniq	omit reported lines in sorted output
man sed; man awk	manuals of more advanced text processing

Commands for manipulating files

cp Allwmake ~/test1	copy file
mv ~/test1 ~/test2	rename file
mv ~/test2 ~/OpenFOAM/	move file
rm ~/OpenFOAM/test2	remove file
mkdir ~/test	make directory
rmdir ~/test	remove empty directory
man rm	manual for rm command (see -r option!)
sort Allwmake >~/test	create file by redirecting
sort Allwmake >~/test	overwrite file by redirecting
sort <~/test >>~/sorted	sort redirected input and save output
sort <~/test >>~/sorted	sort redirected input and add output
chmod 000 ~/test	change file mode (check with `ls -l`)
chmod u+rw ~/test	change file mode (check with `ls -l`)

Note

Read (4), write (2), execute (1). E.g. rwxr-xr-x = 755, rw-r--r-- = 644.

Job control and performance monitoring

ps	list current processes
ps aux | less	list all processes
top	interactive display of processes
kill	kill/terminate process
free -h	report free memory
df -h	report disk space usage
du -sh	estimate file space usage

Tip

Use TAB button for auto-completion of commands!

Visit http://linuxcommand.org for learning more commands.

Note

Recommended text editor GUI: `gedit`

Installing OpenFOAM

- Linux is fully supported by OpenFOAM.
- On non-Linux systems, you can use a virtualization platform, e.g. VirtualBox. You can download a VirtualBox image file from the course website:

  http://www.ara.bme.hu/~hernadi/OpenFOAM/virtualbox.html

- If you have Linux installed, you can install OpenFOAM from

  http://www.openfoam.org/download/

  - Easy installation is possible on Ubuntu. For other systems, you have to download and compile OpenFOAM source code.
  - If you use Git version control system, you can compile OpenFOAM source code with the latest fixes.
  - If you experience problems, you can check the (unofficial) wiki:

    http://openfoamwiki.net/index.php/Installation/Linux/OpenFOAM-2.3.1

> ***Warning***
>
> Compiling OpenFOAM from source usually take several hours! **Be patient!**

Testing OpenFOAM installation

If OpenFOAM is installed, several Linux environment variables are available, e.g

$WM_PROJECT_DIR	OpenFOAM installation directory
$FOAM_RUN	OpenFOAM user directory
$FOAM_TUTORIALS	OpenFOAM tutorials directory

Note

You can list other environment variables by

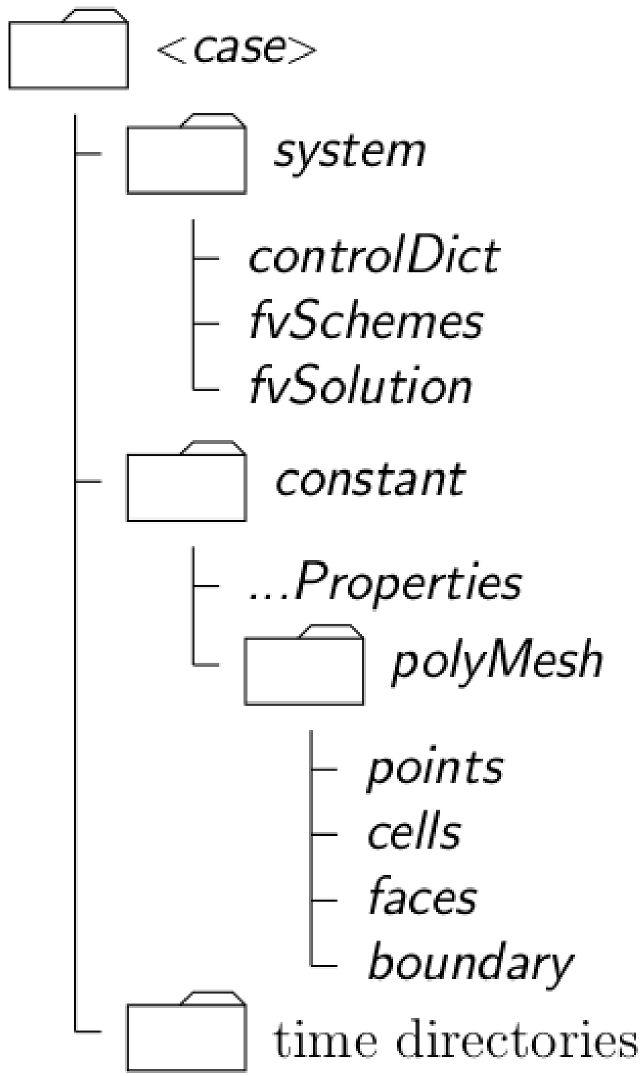
```
echo $FOAM_<TAB><TAB>
env | grep 'WM\|FOAM' | less
```

The usual way of testing OpenFOAM is to run a simple simulation:

```
mkdir -p $FOAM_RUN
cp -r $FOAM_TUTORIALS $FOAM_RUN
cd $FOAM_RUN/tutorials/incompressible/icoFoam/cavity
blockMesh
icoFoam
paraFoam
```

OpenFOAM cases

The basic directory structure:



How OpenFOAM solvers work

Partial differential equation

$$\frac{\partial \rho \mathbf{U}}{\partial t} + \nabla \cdot \phi \mathbf{U} - \nabla \cdot \mu \nabla \mathbf{U} = - \nabla p$$

is represented using high-level C++ syntax inside the solver:

```
solve
(
      fvm::ddt(rho,U)
    + fvm::div(phi,U)
    - fvm::laplacian(mu,U)
      ==
    - fvc::grad(p)
);
```

Numerical treatment settings (discretization schemes and linear-solvers) are inside OpenFOAM cases (*fvSchemes*, *fvSolution*).

```
divSchemes
{
    default         none;
    div(phi,U)      bounded Gauss linearUpwind grad(U);
}
```