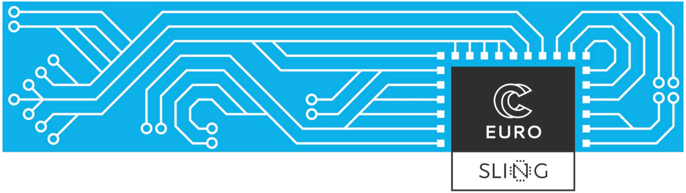
CFD on HPC – OpenFOAM

Literature and OF Tools



Aleksander GRM

Literature

Literature

There are two main web pages with links to literature:

- openfoam.org (on PC workstations)
- cfd.direct (on HPC systems)

User guides:

- User Guide #1
- User Guide #2
- Programming Guide – Learn c++ code!

My GitHub repository:

- OpenFOAM_School@github

Literature

Link to GoogleDrive location for myOpenFOAM

Fluid Mechanics and Its Applications

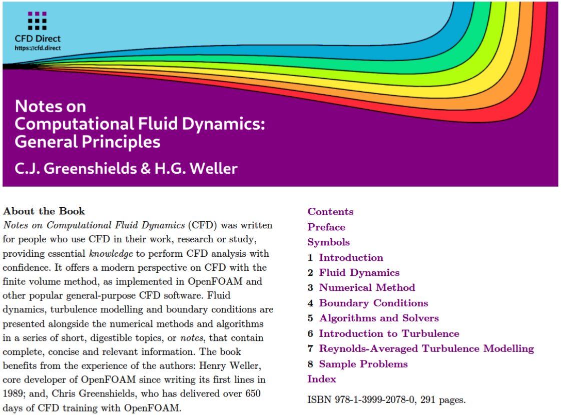
F. Moukalled
L. Mangani
M. Darwish

The Finite Volume Method in Computational Fluid Dynamics

An Advanced Introduction with OpenFOAM® and Matlab®

Springer

link to the book



CFD Direct
https://cfd.direct

Notes on
Computational Fluid Dynamics:
General Principles

C.J. Greenshields & H.G. Weller

About the Book

Notes on Computational Fluid Dynamics (CFD) was written for people who use CFD in their work, research or study, providing essential *knowledge* to perform CFD analysis with confidence. It offers a modern perspective on CFD with the finite volume method, as implemented in OpenFOAM and other popular general-purpose CFD software. Fluid dynamics, turbulence modelling and boundary conditions are presented alongside the numerical methods and algorithms in a series of short, digestible topics, or *notes*, that contain complete, concise and relevant information. The book benefits from the experience of the authors: Henry Weller, core developer of OpenFOAM since writing its first lines in 1989; and, Chris Greenshields, who has delivered over 650 days of CFD training with OpenFOAM.

ISBN 978-1-3999-2078-0, 291 pages.

link to the book

GMSH Tools

Set up GMSH

Set GMSH environment

```
1. load module gmsh:
  $> ml av gmsh (check target version)
  $> ml gmsh/4.11.1-foss-2022a

2. Run gmsh:
  $> gmsh
```

Load **only** Python environment

To use Python environment we need only to load it

```
1. load module python:
  $> ml av python (check target version)
  $> ml python-version

2. Create new env:
  $> python3 -m venv local

3. Activate new env:
  $> source local/bin/activate
```

Set up Python environment with GMSH

To be able to run advanced GMSH examples we need to set up Python environment

```
1. load module python:
  $> ml av python (check target version)
  $> ml python-version

2. Create new env:
  $> python3 -m venv local

3. Activate new env:
  $> source local/bin/activate

4. Install new packages (active env local):
  $(local)> pip install numpy scipy sympy matplotlib gmsh
```

OpenFOAM Tools

Load OpenFOAM environment

To use OpenFOAM environment we need to load

```
List available modules:
    $> module avail openfoam
    $> ml av openfoam (equvalent with upper command)

For OpenFOAM to be running on HPC@ULFS we need to load this modules:
    $> ml OpenFOAM
    $> source $FOAM_BASH (set new OpenFOAM environment variables)
```

Load OpenFOAM gnuplot environment

Additional modules for OpenFOAM environment

```
Load additional modules to support gnuplot in OpenFOAM:
    $> ml OpenFOAM
    $> source $FOAM_BASH

    $> ml av gnuplot (for foamMonitor application)
    $> ml spider gnuplot-version (see needed additional modules to load)

    and load your Python evn with
    $> source work/Python/local/bin/activate (or path where your local
    Python is)
```

OpenFOAM monitor

Add the following part at the end in `system/controlDict`

```
functions
{
    #includeFunc  residuals
}
```

Create residual dictionary file `system/residuals` and include

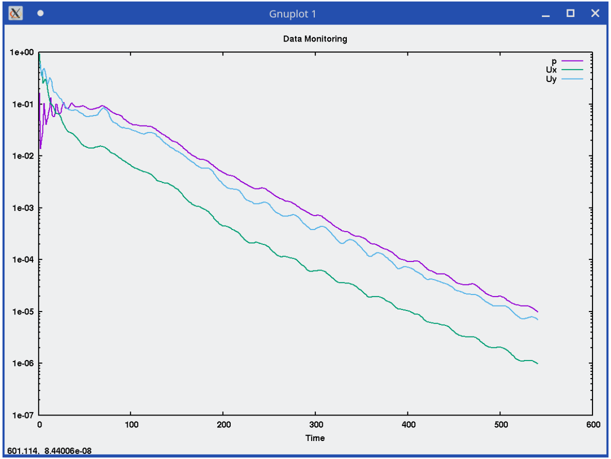
```
#includeEtc "caseDicts/postProcessing/numerical/residuals.cfg"

fields  (p U);
```

Run monitor command

```
foamMonitor -l -r 1 postProcessing/residuals/0/residuals.dat
```

OpenFOAM monitor

```
foamMonitor -l -r 1 postProcessing/residuals/0/residuals.dat
```



Parallel run OpenFOAM @ HPC

Use of Slurm environment

Parallel run is executed via `srun` command (for help: `srun -help`)

```
$> srun --mpi=pmix -c 5 -p haswell foamRun -parallel
```

Important: Use only the partition ***haswell***!

To check your queue use command

`$> squeue` or only for user jobs `$> squeue --me`

To cancel use command

```
$> scancle JOBID
```

Check HPC constellation

```
$> sinfo
```

Use of Slurm environment

Parallel run is executed via `srun` command as seen.

To automatise the process, all commands are packed in shell script and run with command

```
$> sbatch parallel_run.sh
```

Links to Slurm help:

- Slurm help
- Slurm help @ FS-HPC

Bash script parallel_run.sh

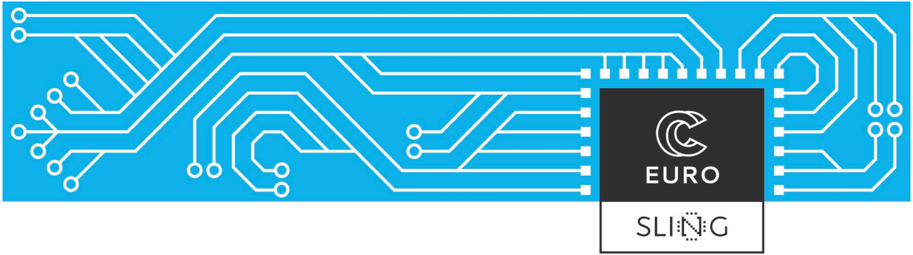
Bash script containing all commands to run OpenFOAM at HPC

```bash
#!/bin/bash
#SBATCH --export=ALL,LD_PRELOAD=
#SBATCH --partition=haswell
#SBATCH --mem=0
#SBATCH --ntasks 32
#SBATCH --ntasks-per-node=16

module purge
module load OpenFOAM
source $FOAM_BASH

# Source tutorial run functions
source $FOAM_ETC/../bin/tools/RunFunctions
```

Bash script parallel_run.sh

```bash
# Path to running case
caseName="cavityFine"
cd $caseName

# decompose the case (number of decompositions is equal to --ntasks)
runApplication decomposePar

# run parallel
echo "Start $(getApplication) in parallel. Log is written in case/log.$(getApplication)!"
srun --mpi=pmix  $(getApplication) -parallel

# Check the running process with: tail -f case/log.$(getApplication)
```

Sponsorship



Thank you for attention!

EuroHPC Joint Undertaking

REPUBLIC OF SLOVENIA
MINISTRY OF HIGHER EDUCATION,
SCIENCE AND INNOVATION

This project has received funding from the European High-Performance Computing Joint Undertaking (JU) under grant agreement No 101101903. The JU receives support from the Digital Europe Programme and Germany, Bulgaria, Austria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, Greece, Hungary, Ireland, Italy, Lithuania, Latvia, Poland, Portugal, Romania, Slovenia, Spain, Sweden, France, Netherlands, Belgium, Luxembourg, Slovakia, Norway, Türkiye, Republic of North Macedonia, Iceland, Montenegro, Serbia.