

Week 2 Progress Report

Anugya Chaubey

Semester-Long Internship 2026
OpenFOAM GUI

March 20, 2026

Work Overview

During Week 1, the primary focus was on understanding the internal working of OpenFOAM and preparing the environment for further development.

Work Completed

- ◀ Installed and configured OpenFOAM using WSL (Ubuntu environment)
- ◀ Executed the first simulation using the cavity flow case
- ◀ Studied the OpenFOAM case directory structure and key files
- ◀ Modified important configuration files:
 - fvSchemes (numerical methods)
 - turbulenceProperties (turbulence models)
- ◀ Debugged multiple real-world configuration errors

This phase helped in building a strong foundation before moving towards automation.

Relevance to Project and Future Work

Week 1 played a critical role in ensuring the success of the overall project.

Why This Step Was Important

- ◀ OpenFOAM is a file-driven system with strict syntax requirements
- ◀ Incorrect configurations can cause simulation failures
- ◀ UI-based automation requires precise understanding of:
 - File structure
 - Syntax rules
 - Required parameters

How This Helps in Coming Weeks

- ◀ Enables accurate automatic file generation
- ◀ Reduces chances of runtime errors
- ◀ Forms the base for:
 - UI development
 - Backend logic implementation

System Setup

The development environment was prepared to run OpenFOAM simulations.

Setup Details

- ▀ Ubuntu installed using Windows Subsystem for Linux (WSL)
- ▀ OpenFOAM v2306 successfully installed
- ▀ Verified installation using tutorial cases

```
Mesh Information
-----
boundingBox: (0 0 0) (0.1 0.1 0.01)
nPoints: 882
nCells: 480
nFaces: 1640
nInternalFaces: 760
-----
Patches
-----
patch 0 (start: 760 size: 20) name: movingWall
patch 1 (start: 780 size: 60) name: fixedWalls
patch 2 (start: 840 size: 800) name: frontAndBack
-----
End

      /\
     //  F ield      OpenFOAM: The Open Source CFD Toolbox
    //   O peration   Version: 2306
   //    A nd         Website: www.openfoam.com
  //   M anipulation
 //
//

Build : c8f9eeb2-20260127 OPENFOAM=2306 patch=260127 version=2306
Arch   : "LSB;label=32;scalar=64"
Exec   : icoFoam
Date   : Mar 29 2026
Time   : 04:28:10
Host   : DESKTOP-3U39T50
PID    : 5326
I/O    : uncollated
Case   : /home/dell/OpenFOAM/run/cavity/cavity
nProcs : 1
trapFpe: Floating point exception trapping enabled (FOAM_SIGFPE).
fileModificationChecking: Monitoring run-time modified files using timeStampMaster (fileModificationSkew 5, maxFileModificationPolls 20)
allowSystemOperations: Allowing user-supplied system call operations
```

Running First Simulation

The first simulation was executed to validate the setup.

Simulation Details

- ◀ Used the cavity flow tutorial case
- ◀ Observed time-step based simulation execution
- ◀ Verified correct solver functioning

```
// ***** //  
Create time  
  
Create mesh for time = 0  
  
PISO: Operating solver in PISO mode  
  
Reading transportProperties  
  
Reading field p  
  
Reading field U  
  
Reading/calculating face flux field phi  
  
Starting time loop  
  
Time = 0.005  
  
Courant Number mean: 0 max: 0  
smoothSolver: Solving for Ux, Initial residual = 1, Final residual = 8.90511e-06, No Iterations 19  
smoothSolver: Solving for Uy, Initial residual = 0, Final residual = 0, No Iterations 0  
DICPCG: Solving for p, Initial residual = 1, Final residual = 0.0492854, No Iterations 12  
time step continuity errors : sum local = 0.000466513, global = -1.79995e-19, cumulative = -1.79995e-19  
DICPCG: Solving for p, Initial residual = 0.590864, Final residual = 2.65225e-07, No Iterations 35  
time step continuity errors : sum local = 2.74685e-09, global = -2.6445e-19, cumulative = -4.44444e-19  
ExecutionTime = 0.02 s ClockTime = 0 s  
  
Time = 0.01  
  
Courant Number mean: 0.0976825 max: 0.585607  
smoothSolver: Solving for Ux, Initial residual = 0.160686, Final residual = 6.83031e-06, No Iterations 19  
smoothSolver: Solving for Uy, Initial residual = 0.260828, Final residual = 9.65939e-06, No Iterations 18  
DICPCG: Solving for p, Initial residual = 0.428925, Final residual = 0.0103739, No Iterations 22
```

Understanding OpenFOAM File Structure

OpenFOAM simulations are controlled through multiple configuration files.

Key Files and Their Purpose

- ◀ fvSchemes – Defines numerical discretization schemes
- ◀ fvSolution – Defines solver settings
- ◀ transportProperties – Defines fluid properties
- ◀ turbulenceProperties – Defines turbulence models

Understanding these files is essential for automating simulation setup.

Numerical Schemes Configuration

Numerical schemes define how equations are solved in the simulation.

Work Done

- Modified divergence scheme in fvSchemes:
 - Changed from Gauss linear to Gauss upwind

Key Learning

- ▶ Linear scheme provides higher accuracy
- ▶ Upwind scheme provides better stability

```

cmake_minimum_required(VERSION 3.10)
project(OpenSSH)

set(CMAKE_CXX_STANDARD 11)

include_directories(
    ${CMAKE_SOURCE_DIR}/include
)

add_executable(ssh
    src/main.cpp
)

target_link_libraries(ssh
    ${CMAKE_SOURCE_DIR}/lib/libssh.a
)

```

```

{
  "flow": {
    "priority": "uint16",
    "vid": "uint16",
    "destination": "string"
  },
  "path": {
    "source": "string",
    "target": "string",
    "type": "enum"
  },
  "default": {
    "path": {
      "source": "linear",
      "target": "linear",
      "type": "linear"
    }
  },
  "required": [
    "path"
  ]
}

```

Debugging and Issues Solved

Several practical issues were encountered and resolved.

Errors Fixed

- ◀ Missing headers in configuration files
- ◀ Incorrect syntax in numerical schemes
- ◀ Missing turbulence-related files
- ◀ Incorrect transport property definitions

This improved understanding of OpenFOAM's strict requirements.

Final Working Case

After modifications and debugging, the simulation ran successfully.

Outcome

- ◀ Simulation executed without errors
- ◀ Modified numerical schemes applied correctly
- ◀ Case validated successfully

```
Create time

Create mesh for time = 0

PISO: Operating solver in PISO mode

Reading transportProperties

Reading field p

Reading field U

Reading/calculating face flux field phi

Starting time loop

Time = 0.005

Courant Number mean: 0 max: 0
smoothSolver: Solving for Ux, Initial residual = 1, Final residual = 8.98511e-06, No Iterations 19
smoothSolver: Solving for Uy, Initial residual = 0, Final residual = 0, No Iterations 0
DICPCG: Solving for p, Initial residual = 1, Final residual = 0.0492854, No Iterations 12
time step continuity errors : sum local = 0.000466513, global = -1.79995e-19, cumulative = -1.79995e-19
DICPCG: Solving for p, Initial residual = 0.590864, Final residual = 2.65225e-07, No Iterations 35
time step continuity errors : sum local = 2.74685e-09, global = -2.6445e-19, cumulative = -4.44444e-19
ExecutionTime = 0.01 s ClockTime = 0 s

Time = 0.01

Courant Number mean: 0.0976825 max: 0.585607
smoothSolver: Solving for Ux, Initial residual = 0.158673, Final residual = 7.34669e-06, No Iterations 19
smoothSolver: Solving for Uy, Initial residual = 0.260484, Final residual = 6.2252e-06, No Iterations 19
```

Why Focus Only on Simulation First?

Understanding Before Automation

- ◀ OpenFOAM is a file-driven system where simulations depend on configuration files
- ◀ Any incorrect syntax or missing parameter leads to simulation failure
- ◀ Direct UI implementation without understanding would produce incorrect outputs

Approach Taken

- ◀ Focused on how simulations run internally
- ◀ Studied required configuration files and their structure
- ◀ Identified common errors and how to fix them

Repo understanding and implementation

Core Changes

◀ Create Central Data File

- `CfdTurbulenceModels.py`
- Stores models, fields, and default coefficients

◀ Extend Physics Model Object

- Modify: `CfdPhysicsSelection.py`
- Add:
 - `TurbulenceModel`
 - `TurbulenceModelCoeffs`
 - `PrintCoeffs`

Key Idea: Store turbulence configuration as structured data

Continued

UI Changes

- ◀ Modify:
`TaskPanelPhysics.ui`
- ◀ Add:
 - Turbulence model dropdown
 - Dynamic coefficient input fields

Key Idea: User-controlled turbulence model selection

Directory / File	Purpose	Relevance to Project
CfdOF/Solve/CfdPhysicsSelection.py	Defines the FreeCAD document object that stores all physics settings (solver type, turbulence category, etc.)	Primary file to extend for both features
CfdOF/Solve/CfdCaseWriterFoam.py	Reads model object properties and writes the full <code>OpenFOAM</code> case directory	Must be extended to write new turbulence and scheme files
Gui/TaskPanelPhysics.ui	Qt Designer XML layout for the Physics Selection task panel	Add turbulence model combo box and scheme controls here
CfdOF/Solve/ (task panel .py)	Python controller class that wires UI signals to the model object	Add signal/slot connections for new UI elements
Data/defaults/	Template <code>OpenFOAM</code> case files used as base for generated cases	Reference for default <code>fvSchemes</code> and <code>turbulenceProperties</code> content

Figure 3: CfdOF Repository Structure

Backend Integration & File Writing

Changes to be Implemented

◀ Update Controller

- TaskPanelPhysics.py
- Handle turbulence model selection
- Dynamically update UI fields

◀ Modify Case Writer

- CfdCaseWriterFoam.py
- Replace template-based writing
- Use selected model data dynamically

◀ Generate OpenFOAM File

- constant/turbulenceProperties

Example Output:

```
simulationType RAS;  
RASModel kOmegaSST;  
turbulence on;
```

Numerical Scheme Configuration

Changes to be Implemented

◀ Create Scheme Definition Files

- `CfdNumericalSchemes.py`
- Define available schemes and presets

◀ Create Numerical Settings Object

- `CfdNumericalSettings.py`
- Store selected schemes and presets

◀ Add New UI Panel

- `TaskPanelNumerics.ui`
- Preset selector + scheme controls

◀ Update Case Writer

- Modify: `CfdCaseWriterFoam.py`
- Generate `system/fvSchemes`

Key Idea: Users control simulation accuracy through discretization schemes

Estimated changes and additions

File	Action	Description of Change
CfdOF/Solve/CfdTurbulenceModels.py	CREATE NEW	Central data dictionary for all supported turbulence models with metadata and default coefficients
CfdOF/Solve/CfdNumericalSchemes.py	CREATE NEW	Central data dictionary for <u>fvSchemes</u> options and named <u>preset profiles</u>
CfdOF/Solve/CfdNumericalSettings.py	CREATE NEW	<u>FreeCAD</u> document object class for storing numerical scheme settings
CfdOF/Solve/CfdPhysicsSelection.py	MODIFY	Add <u>TurbulenceModel</u> , <u>TurbulenceModelCoeffs</u> , <u>PrintCoeffs</u> properties

Importance of Week 2

This phase was essential for ensuring correct automation in future stages.

Key Points

- ◀ OpenFOAM relies on correct file configurations
- ◀ Automation requires exact replication of these files
- ◀ Week 1 ensured:
 - Correct syntax understanding
 - Complete parameter knowledge
 - Reliable and validated configurations

Outcome

- ◀ Built a strong foundation for automation
- ◀ Reduced chances of errors in future development

Connection to Week 3 and Week 4

Week 3: Backend Development

- ◀ Convert validated configurations into Python functions
- ◀ Automate generation of OpenFOAM files (e.g., fvSchemes)
- ◀ Ensure generated files match correct syntax and structure

Week 4: FreeCAD UI Integration

- ◀ Integrate backend logic into FreeCAD interface
- ◀ Add user input options (dropdowns for schemes and models)
- ◀ Connect UI selections to automatic file generation

Final Workflow

- ◀ User Input (UI) → File Generation → Simulation Execution