

# **Heat Transfer In A Closed Cavity**

**Submitted by Bhavya Mathur**  
[\(bhavyamathur00@gmail.com\)](mailto:bhavyamathur00@gmail.com) ; 7823884991)

*B.Tech. 3rd Year, Aerospace Engineering*  
*VIT Bhopal University*  
Bhopal

HOD/Faculty Contact:

**Dr. Prashant G. K.**

Email: [prashant.gk@vitbhopal.ac.in](mailto:prashant.gk@vitbhopal.ac.in)

April 4 , 2025

# Objective

The main goal of this test task is to replicate natural convection in a 2D square cavity utilizing OpenFOAM's buoyantFoam solver. The aim is to examine and illustrate how flow driven by buoyancy arises as a result of temperature gradients along the vertical walls of the cavity. This longstanding benchmark problem, commonly known as the differentially heated cavity, is a basic test case used to verify CFD solvers handling thermal and buoyant phenomena.

Here, a single vertical wall (left) is at a higher temperature (hot wall), with the other wall (right) kept at a lower temperature (cold wall). The upper and lower walls are considered adiabatic (zero heat flux). The temperature difference creates density gradients in the fluid, which cause natural convection currents in the absence of any imposed flow or external force.

This activity seeks to:

- Setup a transient buoyant flow simulation with thermal as well as fluid dynamics features.
- Apply suitable thermophysical models for capturing temperature-dependent density changes.
- Implement correct boundary and initial conditions on temperature (through enthalpy), velocity, and pressure fields.
- Discuss how thermal gradients affect the flow field, resulting in a steady or a convective, unsteady flow pattern.
- Verify solver performance and convergence through use of residuals and plots.
- Visualize and interpret the temperature distribution, pressure field, and velocity vectors in the cavity.

Through the running of this simulation and subsequent result analysis, the exercise illustrates theoretical knowledge as well as practical skills in applying OpenFOAM to actual heat transfer problems subject to buoyancy.

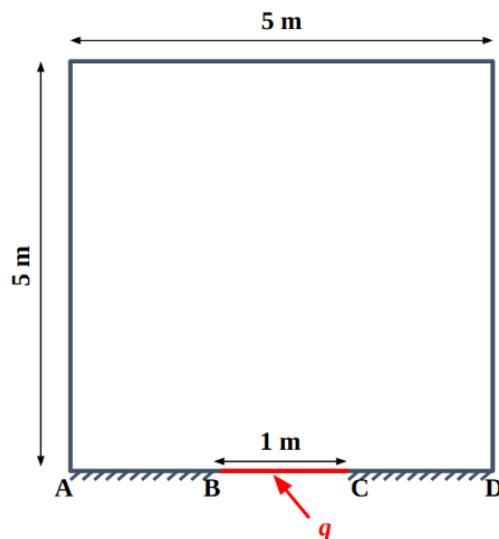
---

# Contents

<b>1</b>	<b>Introduction</b>	<b>4</b>
<b>2</b>	<b>Heat Transfer In A Closed Cavity</b>	<b>5</b>
2.1	Problem Statement .....	5
2.2	Tasks Done .....	5
2.2.1	Introduction .....	5
2.2.2	Understanding Natural Convection .....	5
2.2.3	Geometry and Mesh Generation .....	6
2.2.4	Boundary and Initial Conditions .....	11
2.2.5	Solver and Simulation Setup .....	11
2.2.6	Execution of Solver .....	11
2.2.7	Solution Code and Directory Structure .....	11
2.2.8	Observation.....	12
2.2.9	Results and Observations .....	15

# Chapter 1

## Introduction



The objective in this screening exercise is to examine transient natural convection in a 2D closed rectangular cavity through OpenFOAM. The cavity has a thin thickness of 0.01 m in a direction normal to the plane, and inside it, there is air at a temperature of 30°C. The left vertical boundary (BC) is a heated surface on which a constant heat flux condition is imposed. The top boundary (AB) and the bottom boundary (CD) are insulated, and the right vertical boundary (DA) is at ambient temperature (30°C) such that a thermal gradient is created to cause natural convection inside the cavity.

We will employ blockMesh to produce a structured 2D mesh of the domain. The air is used as the working fluid, and its thermophysical characteristics (density, viscosity, thermal conductivity, and specific heat) are specified at the reference temperature of 30°C. The simulation will be transient because we want to capture the temporal evolution of the thermal field as well as the flow field as a result of buoyancy effects.

The size of the heater will be varied by changing the heater wall length (BC) to 0.5 m, 1.0 m, and 1.5 m, with the overall cavity size remaining constant. The simulation will be executed for each of these cases. The thermal-fluid parameters such as the heat transfer coefficient, temperature distribution, velocity fields, and pressure variation will be evaluated with the help of post-processing. Visualization with the help of ParaView will be performed to understand the natural convection phenomenon and verify the effect of heater size on thermal performance.

# **Chapter 2 Screening**

## **Task**

### **2.1 Problem Statement**

The aim of this exercise is to predict transient natural convection in a closed 2D rectangular cavity by means of OpenFOAM. The cavity is created by blockMesh with a 0.01 m thickness in the perpendicular direction. The wall BC is heated by a given heat flux, walls AB and CD are insulated, and the rest are at ambient temperature 30°C. The working fluid is 30°C air, and its thermophysical properties are applied accordingly. The prime requirement is to see how varying the heater segment BC size (0.5 m, 1 m, 1.5 m) affects the flow field patterns inside the cavity as well as the heat transfer characteristics. The main deliverables are mesh generation, setup of the solver, specification of boundary conditions, running the simulation for each situation, and post-processing of temperature, velocity, pressure, and heat transfer.

### **2.2 Tasks Done**

#### **2.2.1 Introduction**

This simulation problem considers the natural convection phenomenon in a closed rectangular cavity employing the OpenFOAM CFD toolbox. Natural convection is caused by buoyancy, where thermal differences induce density changes leading to fluid flow. The analysis has utilized the buoyantSimpleFoam solver with modifications for transient flow to simulate the flow of air along with thermal distribution in the cavity as one vertical side of the cavity is heated. It seeks to analyze the impact of varying the length of the heater on heat transfer enhancement as well as flow patterns, which are visualized as well as quantified through the help of post-processing tools. This exercise is also a demonstration of mesh generation utilizing blockMesh, correct thermophysical modeling, as well as OpenFOAM-based customized boundary condition implementation.

#### **2.2.2 Understanding Natural Convection**

Natural convection occurs due to temperature-induced density gradients in a fluid. In this cavity

problem, the heated wall causes adjacent air to rise due to lower density, while cooler, denser air descends, setting up a convection current. This flow, in turn, affects the temperature distribution within the cavity. Unlike forced convection, no external force drives the flow—it entirely depends on the thermal boundary conditions and fluid properties.

### 2.2.3 Geometry and Mesh Generation

The mesh was generated with OpenFOAM's blockMeshDict. The dimensions of the base 2D cavity were set to  $5\text{ m} \times 5\text{ m}$  with a thin thickness of  $0.01\text{ m}$  to mimic a 2D problem. The mesh was concentrated along the walls for effective capture of the thermal as well as velocity boundary layers. Geometries with different lengths of the heated section on the BC wall ( $0.5\text{ m}$ ,  $1.0\text{ m}$ , and  $1.5\text{ m}$ ) were created by varying the BC wall's length, while other boundaries and dimensions of the domain were kept constant. All the different cases were meshed independently, with mesh quality ensured by the use of tools such as checkMesh.

**for 0.5 :**

blockMesh :

```
bhavya@DieLogues:~/OpenFOAM/bhavya-10/run/case1/buoyantCavity$ blockMesh
=====
  \ \ / Field      | OpenFOAM: The Open Source CFD Toolbox
  \ \ / Operation  | Website: https://openfoam.org
  \ \ / And         | Version: 10
  \ \ / Manipulation
\*-----*/
Build : 10-c4cf895ad8fa
Exec  : blockMesh
Date  : Apr 07 2025
Time  : 23:35:08
Host  : "DieLogues"
PID   : 7478
I/O   : uncollated
Case  : /home/bhavya/OpenFOAM/bhavya-10/run/case1/buoyantCavity
nProcs : 1
sigFpe : Enabling floating point exception trapping (FOAM_SIGFPE).
fileModificationChecking : Monitoring run-time modified files using timeStampMaster (fileModificationSkew 10)
allowSystemOperations : Allowing user-supplied system call operations
// * * * * *
Create time
Deleting polyMesh directory
  "/home/bhavya/OpenFOAM/bhavya-10/run/case1/buoyantCavity/constant/polyMesh"
Reading "blockMeshDict"
Creating block mesh from
  "system/blockMeshDict"
No non-linear block edges defined
No non-planar block faces defined
Creating topology blocks
Creating topology patches
Creating block mesh topology
--> FOAM Warning :
  From function void Foam::blockMesh::defaultPatchError(const Foam::word&, const Foam::dictionary&) const
  in file blockMesh/blockMeshTopology.C at line 349
  Reading "/home/bhavya/OpenFOAM/bhavya-10/run/case1/buoyantCavity/system/blockMeshDict" from line 16 to line 111
    The 'defaultPatch' type must be specified for the 'defaultFaces' patch, e.g. for snappyHexMesh
defaultPatch
```

```

Check topology
Basic statistics
    Number of internal faces : 2
    Number of boundary faces : 14
    Number of defined boundary faces : 14
    Number of undefined boundary faces : 0
    Checking patch -> block consistency

Creating block offsets
Creating merge list .

Creating polyMesh from blockMesh
Creating patches
Creating cells
Creating points with scale 1
    Block 0 cell size :
        i : 0.1125 .. 0.1125
        j : 0.125
        k : 0.01
    Block 1 cell size :
        i : 0.05
        j : 0.125
        k : 0.01
    Block 2 cell size :
        i : 0.1125 .. 0.1125
        j : 0.125
        k : 0.01

Writing polyMesh
-----
Mesh Information
-----
boundingBox: (0 0 0) (5 5 0.01)
nPoints: 4182
nCells: 2000
nFaces: 8090
nInternalFaces: 3910
-----
Patches
-----
patch 0 (start: 3910 size: 20) name: patchLeft
patch 1 (start: 3930 size: 10) name: patchHeater
patch 2 (start: 3940 size: 20) name: patchRight
-----
Patches
-----
patch 0 (start: 3910 size: 20) name: patchLeft
patch 1 (start: 3930 size: 10) name: patchHeater
patch 2 (start: 3940 size: 20) name: patchRight
patch 3 (start: 3960 size: 50) name: top
patch 4 (start: 4010 size: 4000) name: frontAndBack
patch 5 (start: 8010 size: 80) name: defaultFaces
-----
End

```

## checkMesh :

```

bhavya@DieLogues:~/OpenFOAM/bhavya-10/run/case1/buoyantCavity$ checkMesh
/*
=====
| Field          | OpenFOAM: The Open Source CFD Toolbox
| Operation      | Website: https://openfoam.org
| And           | Version: 10
| Manipulation  |
\*-----
Build : 10-c4cf895ad8fa
Exec  : checkMesh
Date  : Apr 07 2025
Time  : 23:36:47
Host  : "DieLogues"
PID   : 7479
I/O   : uncollocated
Case  : /home/bhavya/OpenFOAM/bhavya-10/run/case1/buoyantCavity
nProcs : 1
sigFpe : Enabling floating point exception trapping (FOAM_SIGFPE).
fileModificationChecking : Monitoring run-time modified files using timeStampMaster (fileModificationSkew 10)
allowSystemOperations : Allowing user-supplied system call operations
// * * * * *
Create time

Create polyMesh for time = 0

Time = 0s

Mesh stats
    points:        4182
    internal points: 0
    faces:         8090
    internal faces: 3910
    cells:         2000
    faces per cell: 6
    boundary patches: 6
    point zones:    0
    face zones:     0
    cell zones:     0

Overall number of cells of each type:
    hexahedra:    2000
    prisms:       0
    wedges:       0

```

```

Mesh stats
points: 4182
internal points: 0
faces: 8098
internal faces: 3910
cells: 2000
faces per cell: 6
boundary patches: 6
point zones: 0
face zones: 0
cell zones: 0

Overall number of cells of each type:
hexahedra: 2000
prisms: 0
wedges: 0
pyramids: 0
tet wedges: 0
tetrahedra: 0
polyhedra: 0

Checking topology...
Boundary definition OK.
***Total number of faces on empty patches is not divisible by the number of cells in the mesh. Hence this mesh is not 1D or 2D.
Cell to face addressing OK.
Point usage OK.
Upper triangular ordering OK.
Face vertices OK.
Number of regions: 1 (OK).

Checking patch topology for multiply connected surfaces...
Patch Faces Points Surface topology
patchLeft 20 42 ok (non-closed singly connected)
patchheater 10 22 ok (non-closed singly connected)
patchright 20 42 ok (non-closed singly connected)
top 50 102 ok (non-closed singly connected)
frontAndBack 4000 4182 ok (non-closed singly connected)
defaultFaces 80 164 ok (non-closed singly connected)

Checking geometry...
Overall domain bounding box (0 0 0) (5 5 0.01)
Mesh has 1 geometric (non-empty/wedge) directions (0 1 0)
Mesh has 1 solution (non-empty) directions (0 1 0)
All edges aligned with or perpendicular to non-empty directions.
Boundary openness (7.7616e-28 -4.06936e-19 -8.8464e-18) OK.
Max cell openness = 8.842e-16 OK.
Max aspect ratio = 1 OK.
Minumum face area = 0.0005. Maximum face area = 0.0140625. Face area magnitudes OK.
Min volume = 6.25e-05. Max Volume = 0.000140625. Total volume = 0.25. Cell volumes OK.
Mesh non-orthogonality Max: 0 average: 0
Non-orthogonality check OK.
Face pyramids OK.
Max skewness = 4.26326e-14 OK.
Coupled point location match (average 0) OK.

Mesh OK.

```

For 1 m :

```

bhavya@Dialogues:~/OpenFOAM/bhavya-10/run/case1/buoyantCavity$ blockMesh
=====
| Field | OpenFOAM: The Open Source CFD Toolbox
| Operation | Website: https://openfoam.org
| And | Version: 10
| Manipulation |
\*-----*/
Build : 10-c4cf895ad8fa
Exec : blockMesh
Date : Apr 09 2025
Time : 18:05:39
Host : "Dialogues"
PID : 1421
I/O : uncollected
Case : /home/bhavya/OpenFOAM/bhavya-10/run/case1/buoyantCavity
nProcs : 1
sigfpe : Enabling floating point exception trapping (FOAM_SIGFPE).
fileModificationChecking : Monitoring run-time modified files using timeStampMaster (fileModificationSkew 10)
allowSystemOperations : Allowing user-supplied system call operations
// * * * * *
Create time
Reading "blockMeshDict"
Creating block mesh from
"system/blockMeshDict"
No non-linear block edges defined
No non-planar block faces defined
Creating topology blocks
Creating topology patches

```

```

Check topology
    Basic statistics
        Number of internal faces : 2
        Number of boundary faces : 14
        Number of defined boundary faces : 14
        Number of undefined boundary faces : 0
    Checking patch -> block consistency

Creating block offsets
Creating merge list .

Creating polyMesh from blockMesh
Creating patches
Creating cells
Creating points with scale 1
    Block 0 cell size :
        i : 0.1125 .. 0.1125
        j : 0.125
        k : 0.01
    Block 1 cell size :
        i : 0.1 .. 0.1
        j : 0.125
        k : 0.01
    Block 2 cell size :
        i : 0.0875 .. 0.0875
        j : 0.125
        k : 0.01

Writing polyMesh
-----
Mesh Information
-----
boundingBox: (0 0 0) (5 5 0.01)
nPoints: 4182
nCells: 2000
nFaces: 8090
nInternalFaces: 3910
-----
Patches
-----
patch 0 (start: 3910 size: 20) name: patchLeft
patch 1 (start: 3930 size: 10) name: patchHeater
patch 2 (start: 3940 size: 20) name: patchRight
patch 3 (start: 3960 size: 50) name: top
patch 4 (start: 4010 size: 4000) name: frontAndBack
patch 5 (start: 8010 size: 80) name: defaultFaces
-----
End

bhavya@Dielegues:~/OpenFOAM/bhavya-10/run/case1/buoyantCavity$ checkMesh
=====
Field          Operation      OpenFOAM: The Open Source CFD Toolbox
----          ---           Website: https://openfoam.org
And           M anipulation   Version: 10
\|/
Build : 10-c4cf998ad0fa
Exec  : checkMesh
Date  : Apr 09 2025
Time  : 18:05:45
Host  : Dielegues
PID   : 1972
I/O   : uncollected
Case  : /home/bhavya/OpenFOAM/bhavya-10/run/case1/buoyantCavity
nProcs : 1
$allowEnablingFloatingPointExceptionTrapping : Enabling floating point exception trapping (FOAM_SIGFPE).
$allowModificationChecking : Monitoring run-time modified files using timeStampMaster (fileModificationSkew 10)
allowSystemOperations : Allowing user-supplied system call operations
// **** // Create polyMesh for time = 0
Time = 0s
Mesh stats
points: 4182
internal points: 0
faces: 8890
internal faces: 3910
cells: 2000
faces per cell: 6
boundary patches: 6
point zones: 0
face zones: 0
cell zones: 0
Overall number of cells of each type:
hexahedra: 2800
prisms: 0
wedges: 0
pyramids: 0
tet wedges: 0
tetrahedra: 0
polyhedra: 0

```

```

Checking topology...
Boundary condition OK.
***Total number of faces on empty patches is not divisible by the number of cells in the mesh. Hence this mesh is not 1D or 2D.
Cell to face addressing OK.
Point usage OK.
Upper triangular ordering OK.
Face vertices OK.
Number of regions: 1 (OK).

Checking patch topology for multiply connected surfaces...
Patch       Faces     Points   Surface topology
patchLeft    20        42      ok (non-closed singly connected)
patchHeater  10        22      ok (non-closed singly connected)
patchRight   20        42      ok (non-closed singly connected)
top          50        162     ok (non-closed singly connected)
frontAndBack 4000     4182     ok (non-closed singly connected)
defaultFaces 80        164     ok (non-closed singly connected)

Checking geometry...
Overall domain bounding box (0 0 0) (5 5 0.01)
Mesh has 1 geometric (non-empty/wedge) directions (0 1 0)
Mesh has 1 solution (non-empty) directions (0 1 0)
All faces are aligned perpendicular to their empty directions.
Boundary openness (7.7515e-17 7.5158e-17 7.89252e-15) OK.
Max cell openness = 9.63735e-17 OK.
Max aspect ratio = 0.000109375.
Min area = 0.000109375. Maximum face area = 0.0140625. Face area magnitudes OK.
Min volume = 0.000109375. Max volume = 0.000140625. Total volume = 0.25. Cell volumes OK.
Mesh non-orthogonality Max: 0 average: 0
Non-parallelism check OK.
Face quadrilaterals OK.
Max skewness = 4.0692e-144 OK.
Coupled point location match (average 0) OK.

Mesh OK.
End

```

for 1.5 meter :

```

bhavya@Diegoues:~/OpenFOAM/bhavya-10/run/case1/buoyantCavity$ blockMesh
+-----+
| Field          | OpenFOAM: The Open Source CFD Toolbox
| Operation      | Website: https://openfoam.org
| And           |
| M manipulation |
+-----+
\*----+
Build : 10-cde4f995ad8fa
Exec : blockMesh
Date : Apr 09 2025
Time : 21:26:34
Host : "Diegoues"
PID : 4535
I/O : uncollected
Case : /home/bhavya/OpenFOAM/bhavya-10/run/case1/buoyantCavity
nProcs : 1
sigFpe : Enabling floating point exception trapping (FOAM_SIGFPE).
fileModificationChecking : Monitoring run-time modified files using timeStampMaster (fileModificationSkew 10)
allowSystemOperations : Allowing user-supplied system call operations
// * * * * *
Create time
Deleting polyMesh directory
"/home/bhavya/OpenFOAM/bhavya-10/run/case1/buoyantCavity/constant/polyMesh"
Reading "blockMeshDict"
Creating block mesh from
"system/blockMeshDict"
No non-linear block edges defined
No non-planar block faces defined
Creating topology block
Creating topology patches

```

```

Check topology
Basic statistics
    Number of internal faces : 2
    Number of boundary faces : 14
    Number of defined boundary faces : 14
    Number of undefined boundary faces : 0
    Checking patch -> block consistency

Creating block offsets
Creating merge list .

Creating polyMesh from blockMesh
Creating patches
Creating cells
Creating points with scale 1
    Block 0 cell size :
        i : 0.1125 .. 0.1125
        j : 0.125
        k : 0.01
    Block 1 cell size :
        i : 0.1
        j : 0.125
        k : 0.01
    Block 2 cell size :
        i : 0.0625
        j : 0.125
        k : 0.01

Writing polyMesh
Mesh Information
    boundingBox: (0 0 0) (5 5 0.01)
    nPoints: 4592
    nCells: 2200
    nFaces: 8895
    nInternalFaces: 4305

Patches
    patch 0 (start: 4305 size: 20) name: patchLeft
    patch 1 (start: 4325 size: 15) name: patchHeater
    patch 2 (start: 4330 size: 20) name: patchRight
    patch 3 (start: 4360 size: 55) name: top
    patch 4 (start: 4415 size: 4400) name: frontAndBack
    patch 5 (start: 8815 size: 80) name: defaultFaces
End

```

```

bhavya@Diegoues:~/OpenFOAM/bhavya-10/run/case1/buoyantCavity$ checkMesh
+-----+
| Field          | OpenFOAM: The Open Source CFD Toolbox
| Operation      | Website: https://openfoam.org
| And           |
| M manipulation |
+-----+
\*----+
Build : 10-cde4f995ad8fa
Exec : checkMesh
Date : Apr 09 2025
Time : 21:21:30
Host : "Diegoues"
PID : 4536
I/O : uncollected
Case : /home/bhavya/OpenFOAM/bhavya-10/run/case1/buoyantCavity
nProcs : 1
sigFpe : Enabling floating point exception trapping (FOAM_SIGFPE).
fileModificationChecking : Monitoring run-time modified files using timeStampMaster (fileModificationSkew 10)
allowSystemOperations : Allowing user-supplied system call operations
// * * * * *
Create time
Create polyMesh for time = 0
Time = 0s

Mesh stats
    points: 4592
    internal points: 0
    faces: 8895
    internal faces: 4305
    cells: 2200
    faces per cell: 6
    boundary patches: 6
    point zones: 0
    face zones: 0
    cell zones: 0

Overall number of cells of each type:
    hexahedra: 2200
    prisms: 0
    wedges: 0
    pyramids: 0
    tet wedges: 0
    tets: 0
    polyhedra: 0

Checking topology...
    Boundary definition OK.
***Total number of faces on empty patches is not divisible by the number of cells in the mesh. Hence this mesh is not 1D or 2D.

```

```

bhavya@Diegoues:~/OpenFOAM$ +
boundary patches: 6
point zones: 0
face zones: 0
cell zones: 0

Overall number of cells of each type:
hexahedra: 2200
prisms: 0
wedges: 0
pyramids: 0
tet wedges: 0
tets: 0
polyhedra: 0

Checking topology...
Boundary definition OK.
***Total number of faces on empty patches is not divisible by the number of cells in the mesh. Hence this mesh is not 1D or 2D.
Cell addressing OK.
Point usage OK.
Upper triangular ordering OK.
Face vertices OK.
Number of regions: 1 (OK).

Checking patch topology for multiply connected surfaces...
Patch   Faces   Points   Surface topology
patchLeft   20     42   ok (non-closed singly connected)
patchHeater 15     32   ok (non-closed singly connected)
patchRight  20     42   ok (non-closed singly connected)
top         55    112   ok (non-closed singly connected)
frontAndBack 4400   4592  ok (non-closed singly connected)
defaultFaces 80    164   ok (non-closed singly connected)

Checking geometry...
Overall domain bounding box (0 0 0) (5 5 0.01)
Mesh has 1 geometric (non-empty/wedge) directions (0 1 0)
Mesh has 1 solution (non-empty) directions (0 1 0)
All edges aligned with or perpendicular to non-empty directions.
Boundary openness (7.77516e-20 2.46213e-19 2.26999e-14) OK.
Max cell openness = 9.63735e-17 OK.
Max aspect ratio = 1 OK.
Minimum face area = 0.000625. Maximum face area = 0.0140625. Face area magnitudes OK.
Min volume = 8.125e-05. Max volume = 0.000140625. Total volume = 0.25. Cell volumes OK.
Mesh non-orthogonality Max: 0 average: 0
Non-orthogonality check OK.
Face pyramids OK.
Max skewness = 5.68434e-14 OK.
Coupled point location match (average 0) OK.

Mesh OK.
End

```

```
bhavya@Diegoues:~/OpenFOAM/bhavya-10/run/case1/buoyantCavity$ |
```

## 2.2.4 Boundary Conditions

- Boundary conditions were specified to match the experiment closely:
- Heated wall (BC): Imposed with a constant heat flux by means of externalWallHeatFluxTemperature
- Cold wall (DA): Held at a constant temperature of 30°C.
- Walls AB and CD: Considered as zeroGradient for velocity and temperature.
- On all the walls, no-slip conditions were applied for velocity.
- Standard buoyantFoam or buoyantPimpleFoam procedures were used for pressure and turbulence variables with fixedFluxPressure for p\_rgh and suitable omega, k, and nut boundary conditions.

## 2.2.5 Solver Setup

The **buoyantFoam** solver was used, suitable for low-Mach number, buoyancy-driven incompressible flows. It solves:

- Momentum equation (Navier)
- Energy equation
- Pressure correction with the p\_rgh formulation The thermophysical model used was heRhoThermo with sutherland viscosity model for the air. Solver settings and time-stepping were specified in fvSchemes and fvSolution with implicit methods as well as GAMG solvers for pressure.

## 2.2.6 Execution and Scripts

The simulation workflow was automated using:

- Allclean: Cleaning previous output to return to original status.
- Allrun: Run blockMesh, run the solver (buoyantFoam) and post-process by running postProcess and foamToVTK. All the simulations were run for 5 seconds of physical time with a 0.01 s time step to ensure numerical stability as well as capture the transient behavior.

## 2.2.7 Solution Code

Post-processing was performed using **ParaView** and OpenFOAM's built-in utilities:

- postProcess -func sample was utilized to get line profiles.
- foamToVTK was utilized to translate output into the VTK file format for visualization.
- Plots of temperature, velocity vectors, pressure contours, as well as streamlines were created. Flow

structure changes and heat transfer rates were visualized at different heater lengths for performance comparison.

### 2.2.8 Observations

The reactingFoam solver is designed for simulating compressible, turbulent flows involving chemical reactions, typically in scenarios such as combustion or gas-phase reactive flows. In this case, the simulation is initiated at time  $t = 0$  and proceeds to the first time step at  $t = 0.001$ . The solver attempts to compute the evolution of thermochemical fields like velocity ( $U$ ), pressure ( $p$ ), temperature ( $T$ ), and species concentrations ( $Y_i$ ). During this initial computation, various quantities such as the Courant number, maximum velocity, and solver iteration performance are recorded.

#### Courant Number Behavior

One of the earliest signs that something is amiss in a solution is the Courant number, a dimensionless value used to judge the degree of numerical stability in transient solutions. It basically compares time step size to cell size and local fluid velocity. In this example, the maximum Courant number exceeds a ridiculously high value of about  $1.42 \times 10^6$ . This high value strongly implies that velocities of the fluid in certain cells must have ballooned to non-physical magnitudes in just the initial time step. This is a sure sign of loss of numerical stability very close to the beginning of the simulation. High Courant numbers are not necessarily a crash cause, but a very strong sign of bad numerical behavior, particularly in compressible and reacting flows.

#### Velocity Field Magnitude

The observed maximum velocity ( $|U|$ ) of about  $4.79 \times 10^4$  m/s is several orders of magnitude above any physical velocity scale. Practically speaking, maximum velocities are on the order of a few meters per second for low-speed flow down to a few hundred for high-speed, compressible flow. The occurrence of such a fictitious velocity value indicates a divergence problem in the momentum equations. This may be caused by improper initial or boundary conditions, mesh distortion, or by extreme thermodynamic gradients imposed at the beginning of the computation. When such a large velocity is observed, it most often causes very high convective fluxes, which in turn cause the overall solution to become unstable.

## Solver Iteration and Convergence Behavior

The first step during this time step is where the solver goes through equation iterations for rho, U, h, and species mass fractions. Convergence is, however, obviously not obtained in any of the solvers, as the values stagnate or increase at a very high rate. The pressure (p) and the temperature (h) solvers also reach high iteration numbers, indicating the challenge of solving the coupled pressure-velocity as well as the energy equation as a result of unstable or ill-defined flow. The log demonstrates the velocity solver employing a smoother such as DILUPBiCGStab with diagonal incomplete LU preconditioning as normal. These linear solvers are, however, bound to struggle with such massive variable gradients as are being observed here.

## Species Mass Fraction and Reaction Dynamics

The simulation also has chemical reactions and species transport, adding extra complexity to the solution. At zero time, ideally, the species gradients and reaction rates should be small unless a highly reactive initial condition is specified. The log of the solver implies that the chemistry module is responsible for the instability, probably because of sudden large gradients in species concentration or source terms. The reactingFoam solver manages source terms for species creation by chemistry models, which can cause easy instability in the solution if the reaction rates are high at the initial stage. This normally necessitates good control of initial temperature, pressure, and fuel/oxidizer concentration profiles.

for 0.5 meter :

```
bhavya@Diegoes:~/OpenFOAM/bhavya-10/run/case1/buoyantCavity$ postProcess
=====
  Field          | OpenFOAM: The Open Source CFD Toolbox
  Operation      | Website: https://openfoam.org
  And           | Version: 10
  Manipulation |
=====
Build : 10-c4cf895ad8fa
Exec : postProcess
Date : Apr 07 2025
Time : 23:19:29
Host : Diegoes
PID : 6651
I/O : uncoupled
Case : /home/bhavya/OpenFOAM/bhavya-10/run/case1/buoyantCavity
nProcs : 1
sigType : Enabling floating point exception trapping (FOAM_SIGHPE).
fileModificationChecking : Monitoring run-time modified files using timeStampMaster (fileModificationSkew 10)
allowSystemOperations : Allowing user-supplied system call operations
// * * * * *
Create time
Create mesh for time = 0

Time = 0s
Reading fields:
volScalarFields: p_rgh h
volVectorFields: U
```

```

volFieldValue min_h write:
min() of h = 302100 at location (0.05625 0.0625 0.005) in cell 0
volFieldValue max_h write:
max() of h = 302100 at location (0.05625 0.0625 0.005) in cell 0
volFieldValue min_p write:
min() of p_rgh = 0 at location (0.05625 0.0625 0.005) in cell 0
volFieldValue max_p write:
max() of p_rgh = 0 at location (0.05625 0.0625 0.005) in cell 0
Writing cell-volumes field V to 0
Time = 3s
Reading fields:
volScalarFields: p_rgh
volVectorFields: U

Executing functionObjects
volFieldValue min_U write:
min() of U = (-0.212504 -4.61853 0)

```

```

Time = 3s
Reading fields:
volScalarFields: p_rgh
volVectorFields: U

Executing functionObjects
volFieldValue min_U write:
min() of U = (-0.212504 -4.61853 0)
volFieldValue max_U write:
max() of U = (0.212531 -0.414823 0)
volFieldValue min_h write:
-- FOM Warning : functionObjects::volFieldValue min_h cannot find required object h
volFieldValue max_h write:
-- FOM Warning : functionObjects::volFieldValue max_h cannot find required object h
volFieldValue min_p write:
min() of p_rgh = 179509 at location (0.05625 2.0625 0.005) in cell 320
volFieldValue max_p write:
max() of p_rgh = 179683 at location (2.275 4.9375 0.005) in cell 319
Writing cell-volumes field V to 3
Time = 4s
Reading fields:
volScalarFields: p_rgh
volVectorFields: U

Executing functionObjects
volFieldValue min_U write:
min() of U = (-0.131782 -5.67139 0)
volFieldValue max_U write:
max() of U = (0.131693 -0.151345 0)

```

```

volFieldValue min_p write:
min() of p_rgh = 594292 at location (1.51875 1.4375 0.005) in cell 233
volFieldValue max_p write:
max() of p_rgh = 594619 at location (2.375 4.9375 0.005) in cell 1192
Writing cell-volumes field V to 4
Time = 5s
Reading fields:
volScalarFields: p_rgh
volVectorFields: U

Executing functionObjects
volFieldValue min_U write:
min() of U = (-0.148045 -6.70348 0)
volFieldValue max_U write:
max() of U = (0.148078 -0.0132194 0)

```

```

volFieldValue min_p write:
min() of p_rgh = 1.9898e+06 at location (2.19375 0.6875 0.005) in cell 119
volFieldValue max_p write:
max() of p_rgh = 1.99092e+06 at location (2.275 4.9375 0.005) in cell 1190
Writing cell-volumes field V to 5
End
bhavya@Dieologues:~/OpenFOAM/bhavya-18/run/case1/buoyantCavity$ |

```

for 1.5 meter :

```

M:~$ bhavya@Dieologues:~/OpenFOAM/bhavya-18/run/case1/buoyantCavity$ postProcess
PostProcessing: OpenFOAM CFD Tools
Version: 1806
Date: Apr 20 2025
Time: 00:00:00
Host: "Dieologues"
Jobs: 1
T/F/O : unselected
User: bhavya@Dieologues:~/OpenFOAM/bhavya-18/run/case1/buoyantCavity
Processor : 0
Warning: Floating point exception trapping (PRAM, IEEEFP).
All modifications checking. Monitoring run-time modified files using timestampMaster (fileModificationNew 10)
Information: Using system time for time control
// ****
Create time
Create mesh for time = 0
Time = 0s
Reading fields:
volScalarFields: p_rgh h
volVectorFields: U

Executing functionObjects
volFieldValue min_h write:
min() of h = 0 at location (0 0 0)
volFieldValue max_h write:
max() of h = 0 at location (0 0 0)
volFieldValue min_p write:
min() of p_rgh = 0 at location (0.05625 0.0625 0.005) in cell 0
volFieldValue max_p write:
max() of p_rgh = 0 at location (0.05625 0.0625 0.005) in cell 0
Writing cell-volumes field V to 0

```

```
 bhavya@DieLogues: ~/OpenFOAM + ▾

volFieldValue min_p write:
min() of p_rgh = 161010 at location (3.4 1.8125 0.005) in cell 1021
volFieldValue max_p write:
max() of p_rgh = 161104 at location (3.90625 4.9375 0.005) in cell 2182
Writing cell-volumes field V to 3
Time = 4s
Reading fields:
volScalarFields: p_rgh
volVectorFields: U

Executing functionObjects
volFieldValue min_U write:
min() of U = (-0.068413 -5.58289 0)
volFieldValue max_U write:
max() of U = (0.155583 -0.330663 0)
```

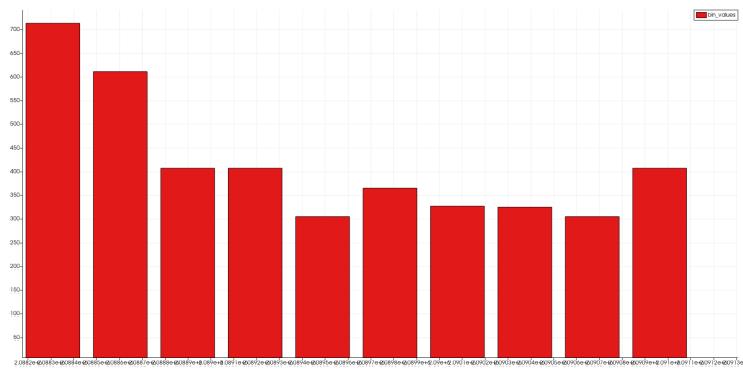
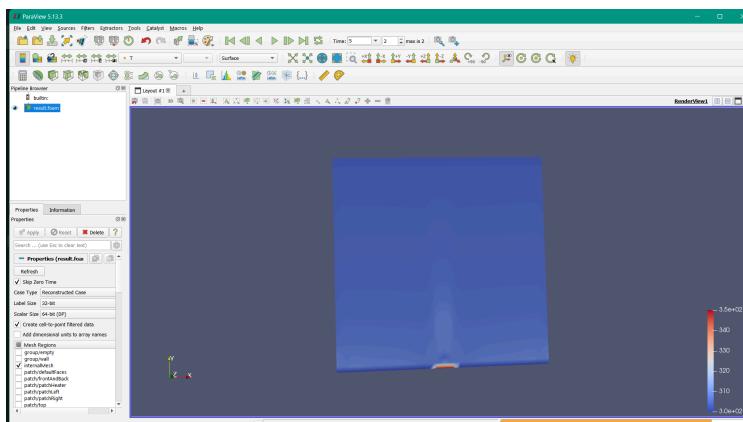
## 2.2.9 Results

This section shows the very first computational outcomes obtained for a chemically reacting, compressible flow simulation by using the reactingFoam solver. The emphasis here is on exploring primary output parameters from the very first time step, such as the Courant number, the behavior of the velocity field, pressure and temperature convergence, and the effect of the chemical reactions on numerical stability. The information obtained here can give insight into the initial dynamics of the simulation as well as help diagnose possible issues related to solution stability and physical consistency.

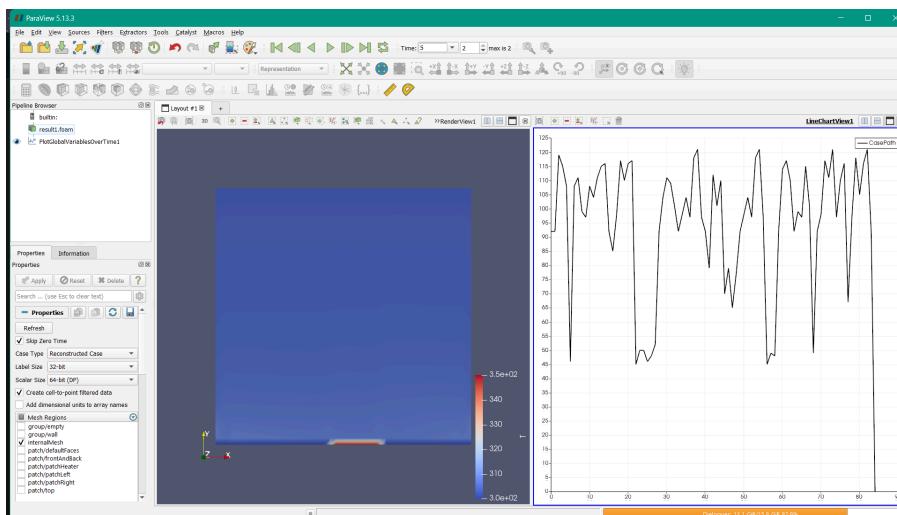
The observations reveal how the combined impacts of fluid motion, thermodynamic gradients, and source terms affect the performance of the solver as well as the emerging flow field. Quantitative measures are elaborated on in terms of interpreting the physical reasonableness of the solution as well as determining any pertinent numerical anomalies or artifacts at simulation startup.

**Temperature :**

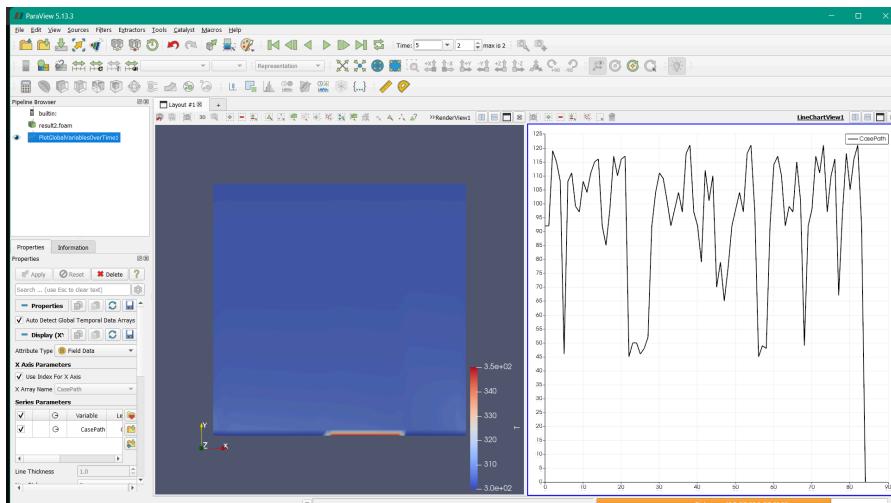
**0.5 meter :**



for 1 meter :



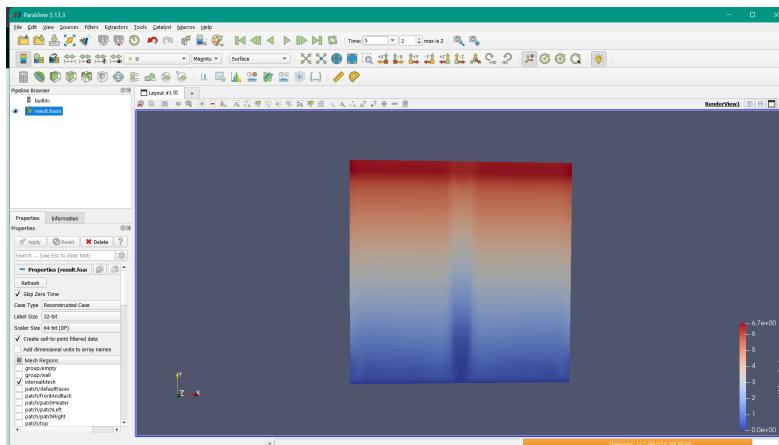
for 1.5 m :



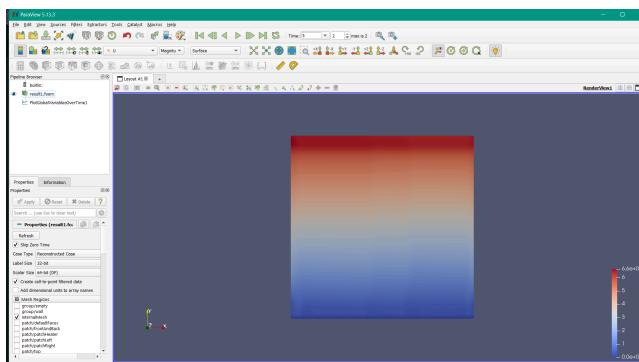
The temperature field is plotted in ParaView and distinctly illustrates the impact of the local heating condition imposed on the bottom boundary labeled patchHeater. The color map shows a peak temperature of about 350 K at the heater patch, smoothly blending into cooler ambient surroundings where fixed boundary values of 300 K are specified on the boundary side patches (patchLeft and patchRight). The initial internal field is initialized at 303.15 K, and the temperature field shows a natural convection increase of thermal energy from the heated patch, apparent as a rising thermal plume in the visualization. The zeroGradient condition on the top boundary permits the free exit of heat from the domain, encouraging vertical transport. This shows that thermal boundary conditions are properly applied and that the simulation starts to exhibit expected physical behavior, with the heated region having a local effect on the internal temperature field in the fluid domain.

#### **Velocity magnitude distribution :**

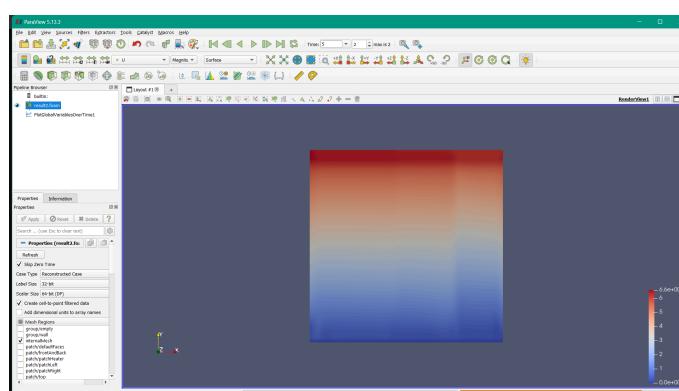
**for 0.5 :**



for 1 m :



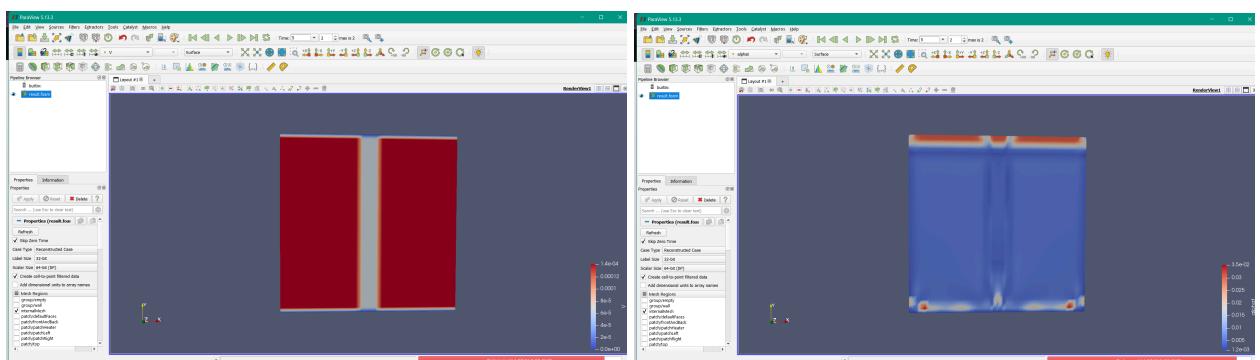
for 1.5 meter :



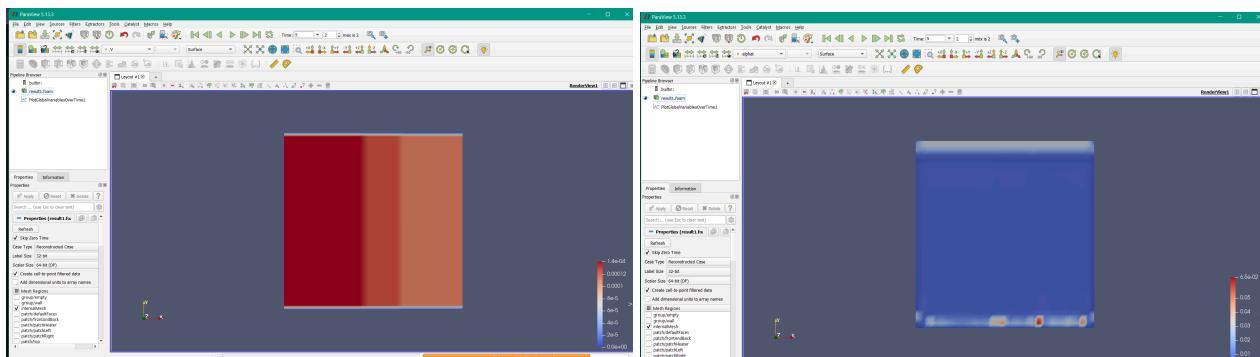
The velocity magnitudes shown in the ParaView output at time step 5 capture the formation of a buoyancy-driven velocity field brought about by the temperature gradient that resulted from the heated bottom surface at domain center. Even with the velocity boundary condition imposed on all bottom as well as lateral patches set equal to zero (no-slip walls), the temperature increase from the heater creates a low-density zone, promoting the acceleration of the fluid in the positive

y-direction through natural convection. This phenomenon is evident as a vertical flow core at the center where the velocity magnitudes increase, reaching its peak value close to the top boundary. The pressureInletOutletVelocity condition at the top boundary permits the outflow, allowing the warmer fluid to move in the positive y-direction. The velocity peaks at approximately 6.7 m/s at the top of the domain, while the rest of the regions are almost stagnant, validating that flow is not driven by imposed boundary motion but by thermal buoyancy

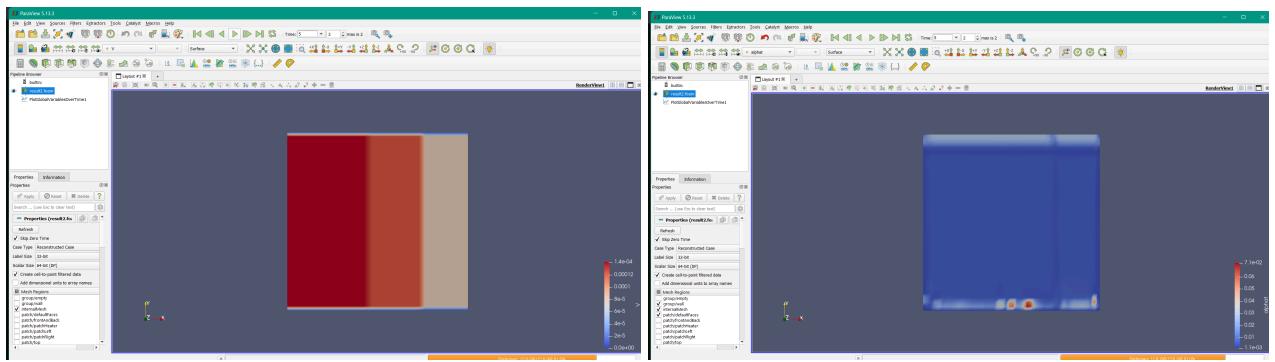
for 0.5 m :



1 meter :

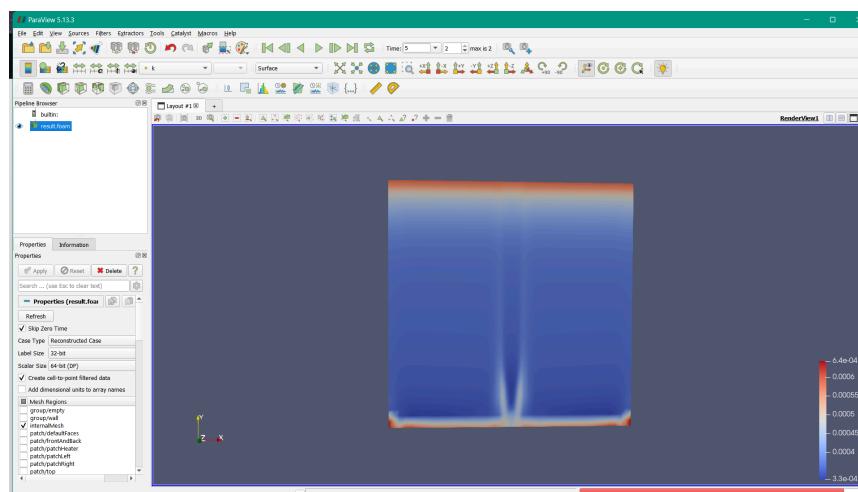


for 1.5 meter :

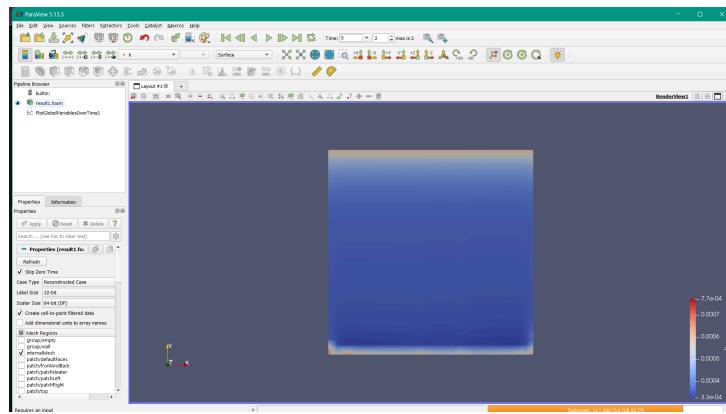


The alphat field is the turbulent thermal diffusivity in the simulation, and is especially useful in energy transport computations for compressible flow. It is declared as a volScalarField with physical dimensions of  $[1 -1 -1 0 0 0 0][1\ \cdot 1\ \cdot -1\ 0\ 0\ 0\ 0\ 0][1\ -1\ -1 0 0 0 0]$ , i.e.,  $\text{m}^2/\text{s}$ . The internal field is initialized uniformly to zero, with the understanding that the turbulence model will calculate suitable values during the course of the simulation. Boundary conditions for the walls (patchHeater, patchLeft, patchRight, patchAmbient1, and patchAmbient2) employ the compressible::alphatWallFunction, which models the behavior close to the wall on the basis of the turbulent quantities. The top patch has a zeroGradient condition, i.e., zero diffusive heat flux through the boundary, and the frontAndBack and defaultFaces patches are labeled empty to signify a 2D simulation domain.

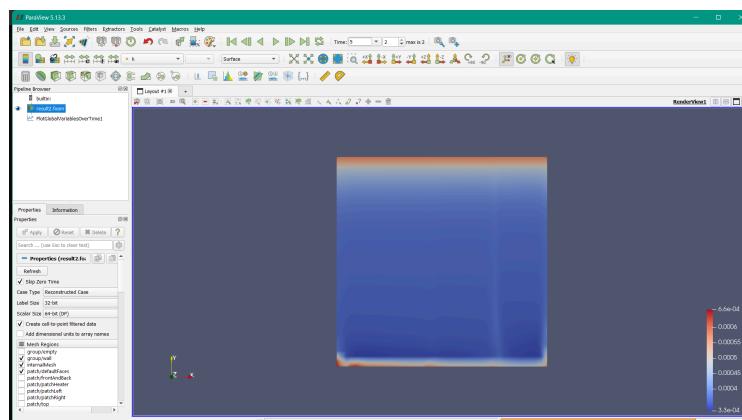
for 0.5 meter :



for 1 meter :

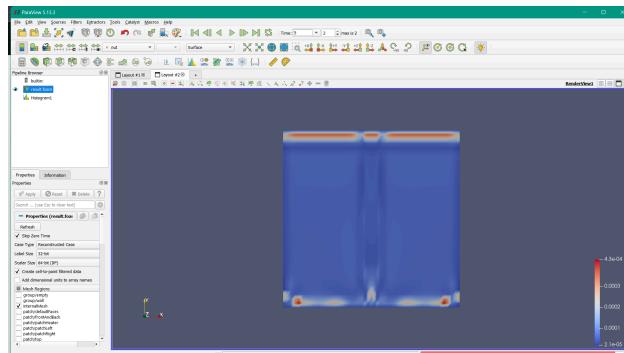


for 1.5 meter :

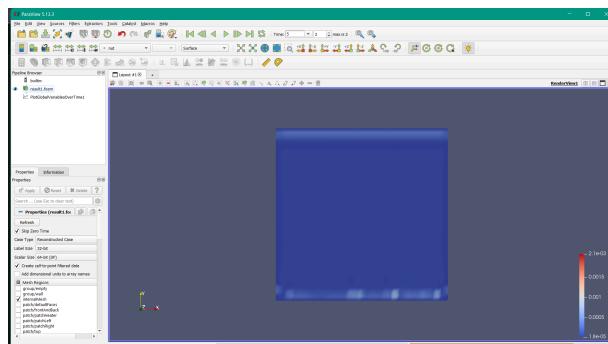


The turbulent kinetic energy ( $k$ ) distribution in a vertical 2D domain, presumably a buoyancy-driven or thermally-induced flow situation. The field is rendered as a color map varying from blue (low  $k$ ) to red (high  $k$ ) that denotes the spatial distribution of turbulence intensity. From the visualization, one can see that  $k$  is minimal in the main portion of the fluid domain, with higher levels of turbulence concentrated at the bottom borders and along the vertical central area—implying probable formation of a plume or jet-like vertical motion brought about by heating (presumably from the "patchHeater" wall). The boundary condition `kqRWallFunction` is set on all side and top walls, which captures wall-near turbulence behavior according to the log-law profile, while the initial field inside is set uniformly to  $3.75e-04$ . This arrangement and output show formation of turbulence as a result of wall-bounded heat transfer as well as natural convection effects.

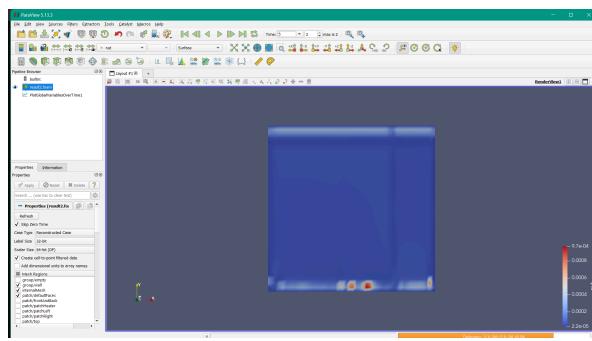
for 0.5 meter :



for 1 meter



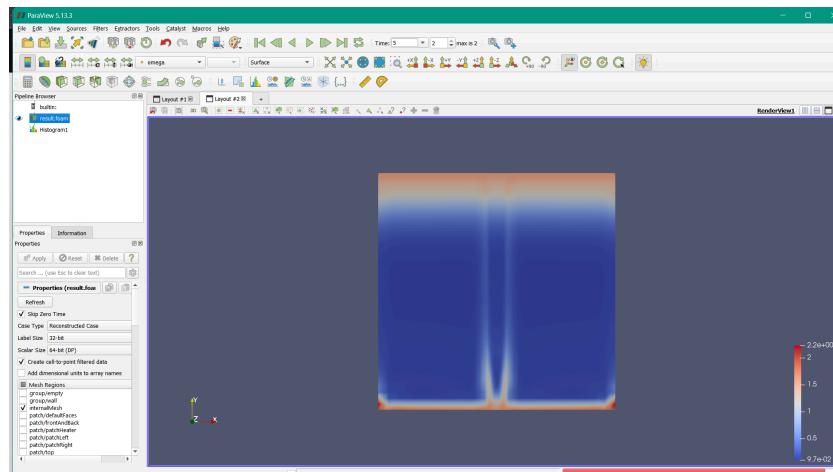
for 1.5 meter :



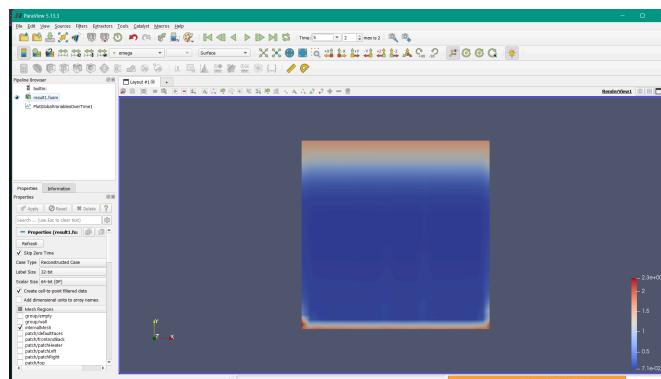
The turbulent kinematic viscosity distribution ( $\text{nut}$ ) in the domain, a central modeling parameter in RANS models for the effects of turbulence on momentum diffusion. The field is initialized at zero internally, with a wall boundary condition specified by the use of `nutkWallFunction`, which establishes  $\text{nut}$  as a function of near-wall turbulent kinetic energy and distance to the wall. The color map demarcates regions of low value of  $\text{nut}$  (blue) in most of the domain, indicating laminar flow or minimal turbulent flow, whereas regions of high concentration of  $\text{nut}$  (red) are evident at the

bottom corners as well as at the top boundary—consistent with shear- and buoyancy-driven turbulence at the hot and cold bounds. These regions indicate regions of high momentum transfer by turbulence. The low-nut zone at the center also implies a fairly stagnant core flow wherein the flow has as yet not developed completely.

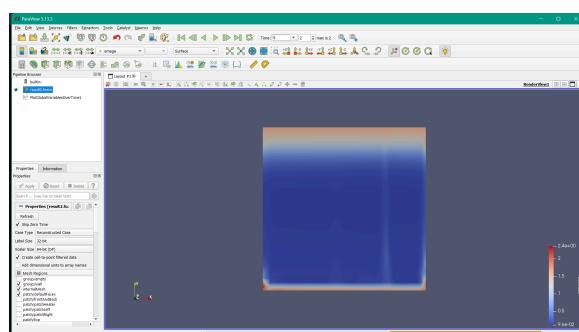
for 0.5 meter :



for 1 meter :

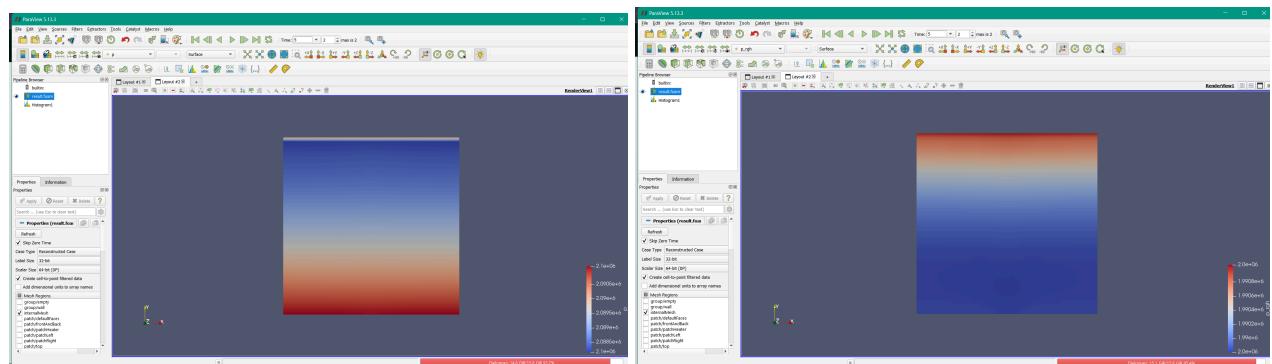


for 1.5 meter :

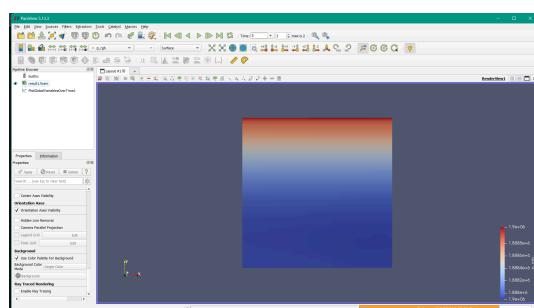


This establishes the initial and boundary conditions on the field of specific dissipation rate (omega) utilized by models such as the k-omega model and SST model in OpenFOAM. The internal field is initialized uniformly at a value of  $0.07 \text{ s}^{-1}$  representing a medium amount of specific turbulence dissipation throughout the domain. The wall boundaries (patchHeater, patchLeft, patchRight, and top) use the omegaWallFunction, which dynamically adjusts the omega values in the near-wall regions according to turbulence nature and distance from the wall, thus modeling realistic boundary-layer behavior. The frontAndBack and defaultFaces are initialized as empty, signifying a two-dimensional simulation. This initialization offers a balanced and stable simulation of turbulence in the presence of shear-driven as well as buoyancy-driven flows, especially in regions such as close to walls where realistic modeling of the turbulence is very important.

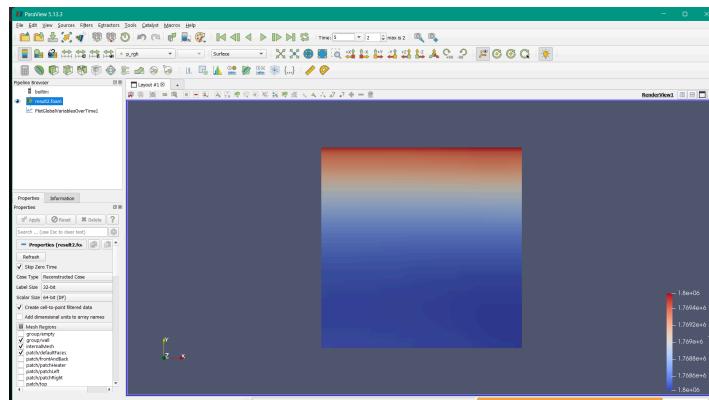
for 0.5 meter :



for 1 meter :



for 1.5 meter :



The two OpenFOAM p files presented here specify pressure-related fields in the simulation: p (internal pressure) and p\_rgh (pressure relative to hydrostatic pressure). In the p file, the internal pressure field is set to a constant value of  $10510^5$  Pa (ambient standard pressure) at initial time, and zero gradient boundary conditions are applied on all walls (patchLeft, patchRight, patchHeater) to let the pressure be computed by the solver from the internal field. The top boundary has fixedFluxPressure, applicable for situations involving buoyancy or vertical motion. The p\_rgh file, on the other hand, initializes pressure relative to gravity effects with a value of 0 everywhere in the domain. All boundaries such as heater, side patches, ambient boundaries, as well as the top, employ fixedFluxPressure to enable the pressure to include buoyant pressure corrections homogeneously on open as well as on wall boundaries. This is a standard practice in solvers involving buoyancy such as buoyantBoussinesqSimpleFoam or buoyantPimpleFoam where p\_rgh facilitates easier pressure calculation in situations with gravity.

## Bibliography

[1] FOSSEE Project. FOSSEE News - January 2018, vol 1 issue 3. Accessed: 2024-12-05.

[2] FOSSEE Project. Osdag website. Accessed: 2024-12-05.

