Summer Fellowship Report

On

Analysis of body(sphere) falling in liquid(water)

Submitted by

Raj Niraj Patil

Under the guidance of

Prof. Shivasubramanian Gopalakrishnan
Mechanical Engineering Department
IIT Bombay

July 8, 2019

Acknowledgment

I thank Prof. Shivasubramanian Gopalakrishnan and Prof.Kannan M. Moudgalya for their invaluable guidance. At the same time, I would like to thank Mr Sathish Kanniappan and Ms Deepa Vedartham for their undying support.

These six weeks were one of the most prestigious weeks. I've learnt so much more about CFD, OpenFOAM and other FLOSS. I acknowledge the FOSSEE team for conducting this summer internship. The last but not least, I would like to thank the authority of IIT Bombay for letting us use the resources during this period and helping us from starting day of the internship till the end.

Contents

Chapter 1

Introduction

We usually observe bodies falling in the water. It is an important thing to analyse forces on these kind of bodies. This generally includes life boats, body physics of divers, etc. This case study is prepared to explore the capabilities of OpenFOAM to solve this particular case. Case intends to spread knowledge about 6 DOF dynamics and overset.

After going through this report, reader is expected to be able to use overset grid method in his/her further simulations. Before overset grid method, people used to use dynamic mesh approach in which mesh used to get deformed. When mesh deforms it increases skewness and as it increases, result gets worse. To avoid these kind of problems due to mesh deformation, overset grid method was developed.

The power inherent in the simple concept of disconnecting domain connectivity from grid construction cannot be overstated. In addition to simplifying the grid generation process, component grids can now be tailored to the local geometry, physics, and even solution model. Time and time again, compromises in grid quality to facilitate domain connectivity have been shown to reduce simulation accuracy and robustness. By using overset grid technology, such problems can be mitigated.

Overset mesh generation is then conceptually split into off-body or background grids and near-body grids which resolve geometry and viscous effects. Often structured hexahedral component grids are used for their efficiency and accuracy. However, the overset technique is routinely applied using hybrid unstructured grids for highly automated meshing of complex configurations.

Chapter 2

Analysis of body(sphere) falling in liquid(water)

2.1 What is Overset grid?

The **Overset** or **Chimera** grid approach utilizes a set of grids that encompass the computation domain and possibly overlap each other without requiring point-matched connectivity between individual grids.

A CFD solution on the system of grids requires coupling the solution between grids in the overlapped regions. This is typically performed by identifying appropriate intergrid boundary locations in one grid and obtaining the value to be applied by interpolating the solution from grids that overlap the region.

The domain connectivity information (DCI) consists of the locations that are to be excluded from the computation, the location of the intergrid boundary locations, and the corresponding interpolation sources. This domain connectivity information is computed by a code typically called an overset grid assembly code.

The overset approach also enables changing the geometry and grid system locally without requiring regeneration of other grids. This flexibility greatly simplifies design studies as geometry perturbations can easily be added to an existing design and grid system by gridding the new feature and possibly including grids to connect the new feature with the existing grids. Since the baseline grid system is not altered the changes in the flow are more reflective of the change in the geometry and not changes resulting from regridding the entire geometry.

Local enrichment is another similar capability that is enabled by the use of overset grids. In this use case, additional grids with enhanced resolution are added in appropriate regions. The baseline grid system again does not need to be regenerated, which simplifies the grid generation task and isolates the flow changes to the improved flow resolution.

The use of an overset grid system is also an enabling technology for the simulation of bodies in relative motion where geometry components or whole bodies move relative to one another. This capability has been widely used for aircraft such as weapon separation, where a bomb or missile is dropped from a parent aircraft, and rotorcraft for high fidelity simulations of helicopters with blades that may rotate, flap, and flex relative to the fuselage. Candidate hydrodynamic applications

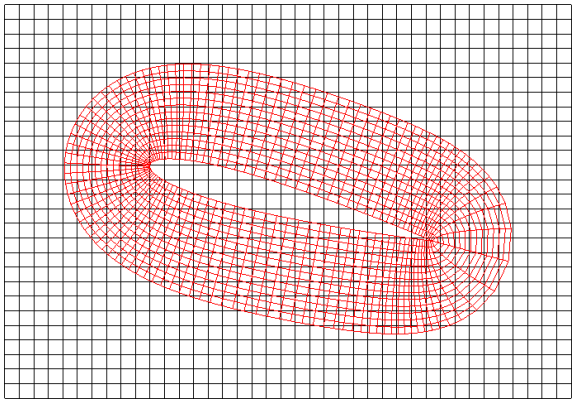

Fig 1: Airfoil grid overlapping background Cartesian grid

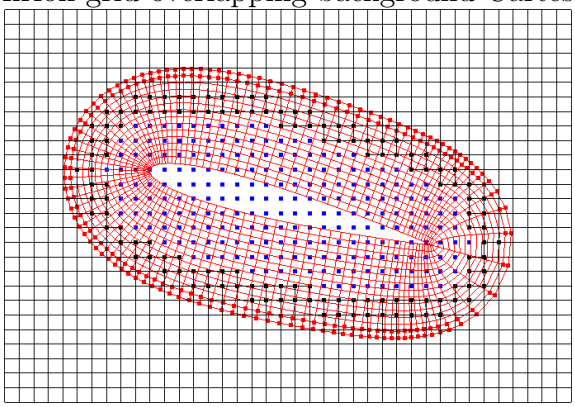

Fig 2: Grid system with hole and fringe points identified after application of process to reduce the amount of overlap between component grids.

include ships or submarines with rotating propulsors, the launch of torpedo or mini-submarines from a parent ship, moving control surfaces, ship motion relative to the sea surface, and sea keeping simulations with multiple ships in close proximity.

The Overset approach is also very useful for unstructured grid systems. The relative motion capability enabled by the use of overset grids is widely used with unstructured grids to enable simulations such as weapon separation, helicopter blade motions, etc. The addition of design changes via overset grids is also being utilized with unstructured grids. [1]

2.2 Six DOF model

In many applications, the influence of flow on position and orientation of a rigid body is of interest. This requires allowing complete freedom to translate and rotate for the body. This model available in OpenFOAM allows user to use thin kind of methodology in his problem. We can control all sorts of constraints and restrains in this model. The nature of this model for the particular case is defined in **dynamicMeshDict**.

2.3 Geometry & Meshing

Geometry of this case contains a sphere of diameter 2cm freely falling on the surface of water which is 3cm below the center of mass of the sphere. In this case we are going to use overset grid with six degrees of freedom(DOF) model. To use overset, we have to separately mesh the components (in this case, sphere and domain). Meshing for sphere was done in **Salome** and domain was set up in **blockMesh**. This problem was solved in 2D but, it can be solved in 3D by following the exact steps. The size of 2D domain is 10cm x 16cm. As OpenFOAM can not work with 2D cases directly, we have to create 3D domain with only one cell in 3rd direction. After creation of separate geometries, geometries are combined together by the command "**mergeMeshes address_of_first_mesh_directory address_of_second_mesh_directory**". While creating submeshes i.e meshes for all the sub components, we have to take care of few things. The sides surrounding overset mesh are defined as overset while defining the patch type. Also, there is need to define a patch with type overset in background or domain mesh. This will trigger overset interpolations in given problem. Every submesh should be finer than the background mesh.

The geomery of spherical submesh looks like shown below.

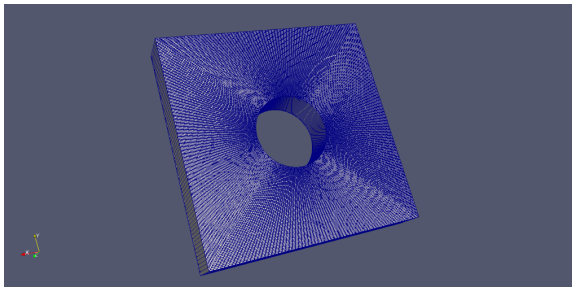

Fig 3: Spherical Submesh

Here the periphery around the sphere is named as **sphere** under patch type wall. The sides surrounding the overset mesh are named as sides with patch type overset. Front and back are named as **frontAndBack** are kept empty.

Background mesh contains three types of patches namely **atmosphere**, **stationaryWalls**, **defaultFaces** with top side being **atmosphere**, front and back being **defaultFaces** and remaining as **stationaryWalls**. **Atmosphere** is patch, **defaultFaces** are empty and **stationaryWalls** are set as wall.

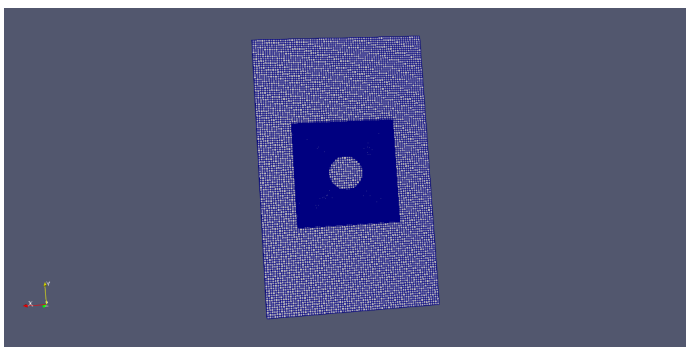

Fig 3: Merged mesh

2.4 Solver

The solver used for this casse study is **overInterDyMFoam**. **overInterDyMFoam** solves for two incompressible immiscible fluids under isothermal conditions using a volume of fluid approach. It also allows us to use mesh motion, mesh topology changes and adaptive re-meshing. The solver solves the Navier Stokes equations for two incompressible, isothermal immiscible fluids. That means that the material properties are constant in the region filled by one of the two fluid except at the interphase.

2.5 Case Setup

2.5.1 Boundary Conditions

As for velocity boundary conditions, **stationaryWalls** were set to noSlip, atmosphere to **pressureInletOutletVelocity** (velocity is calculated from available pressure value) and sphere was set to be **movingWallVelocity** with velocity magnitude zero in all the directions. Note that, since this is a 2D simulation, **defaultFaces** and **frontAndBack** are kept empty in all of the case files. Also, the overset patches (here, sides and oversetPatch) are needed to be set as overset in the entry patchType of all the case files. Pressure boundary conditions were set to be **fixedFluxPressure** for patches **stationaryWalls** and sphere and atmosphere was set at **totalPressure**. In pointDisplacement file, everything was made stationary by making **fixedValue** of everything zero except sphere, which comes under six DOF model analysis. Six DOF model calculates all the forces acting on given body and helps to govern motion to that body accordingly. **zeroGradient** was set for all the patches except atmosphere which was set as **inletOutlet** in alpha.water.

2.5.2 Setting up dynamicMeshDict

```
motionSolverLibs     ("libsixDoFRigidBodyMotion.so");

dynamicFvMesh        dynamicOversetFvMesh;

solver           sixDoFRigidBodyMotion;

sixDoFRigidBodyMotionCoeffs
{

    patches          (sphere); // because we want to solve this model for
    innerDistance    100.0;  //these are the distances between mesh deform
    outerDistance    101.0;  //allowed. In overset there is no deformation

    centreOfMass     (0 0 0);
```

```
    // Density of the solid
    rhoSolid        500;

    // Cuboid mass
    mass             0.002;

    // Cuboid moment of inertia about the centre of mass
    momentOfInertia ( 0.00000008 0.00000008 0.00000008);

    report          on;
    accelerationRelaxation 0.6;
    accelerationDamping    0.9;

    solver
    {
        type Newmark;
    }

    constraints
    {


        arrestRotation
            {
                    sixDoFRigidBodyMotionConstraint    orientation;

            }
    }
}
```

2.6 Results

Reference

- http://celeritassimtech.com/?page_id=15

Summer Fellowship Report

On

Hydraulic Jump

Submitted by

Raj Niraj Patil

Under the guidance of

Prof. Shivasubramanian Gopalakrishnan
Mechanical Engineering Department
IIT Bombay

July 8, 2019

Acknowledgment

I thank Prof. Shivasubramanian Gopalakrishnan and Prof.Kannan M. Moudgalya for their invaluable guidance. At the same time, I would like to thank Mr Sathish Kanniappan and Ms Deepa Vedartham for their undying support.

These six weeks were one of the most prestigious weeks. I've learnt so much more about CFD, OpenFOAM and other FLOSS. I acknowledge the FOSSEE team for conducting this summer internship. The last but not least, I would like to thank the authority of IIT Bombay for letting us use the resources during this period and helping us from starting day of the internship till the end.

Contents

Chapter 1

Introduction

A hydraulic jump is a phenomenon commonly observed in day to day life. When liquid at high velocity discharges into a zone of lower velocity, a rather abrupt rise occurs in the liquid surface. The rapidly flowing liquid is abruptly slowed and increases in height, converting some of the flow's initial kinetic energy into an increase in potential energy. In an open channel flow, this manifests as the fast flow rapidly slowing and piling up on top of itself similar to how a shockwave forms.

The phenomenon is dependent upon the initial fluid speed. If the initial speed of the fluid is below the critical speed, then no jump is possible. For initial flow speeds which are not significantly above the critical speed, the transition appears as an undulating wave. As the initial flow speed increases further, the transition becomes more abrupt, until at high enough speeds, the transition front will break and curl back upon itself. When this happens, the jump can be accompanied by violent turbulence, eddying, air entrainment, and surface undulations, or waves.

The hydraulic jump is the most commonly used choice of design engineers for energy dissipation below spillways and outlets. A properly designed hydraulic jump can provide for 60-70 % energy dissipation of the energy in the basin itself, limiting the damage to structures and the streambed. Even with such efficient energy dissipation, stilling basins must be carefully designed to avoid serious damage due to uplift, vibration, cavitation, and abrasion. An extensive literature has been developed for this type of engineering. [1]

Chapter 2

Hydraulic Jump

2.1 Abstract

This report aims to simulate the Hydraulic Jump using OpenFOAM. This is a simple yet essential simulation for engineering purposes, especially for civil applications. The same simulations can be used to simulate the crown formation by a water droplet. This multi-phase simulation is done using interFoam with geometry and meshing have done using blockMesh.

2.2 Geometry & Meshing

Geometry is simple 10cm x 15cm x 10 cm cube, which was defined using blockMesh after which meshing was carried out in the same. The bottom side is named as **table**, the top most side is named as **atmosphere** and all other remaining sides are named as **walls**. Walls and table are defined as wall while atmosphere is as patch.

2.3 Solver

This simulation is done using interFoam. InterFoam solves for two incompressible immiscible fluids under isothermal conditions using a volume of fluid approach. It also allows us to use mesh motion, mesh topology changes and adaptive re-meshing. The solver solves the Navier Stokes equations for two incompressible, isothermal

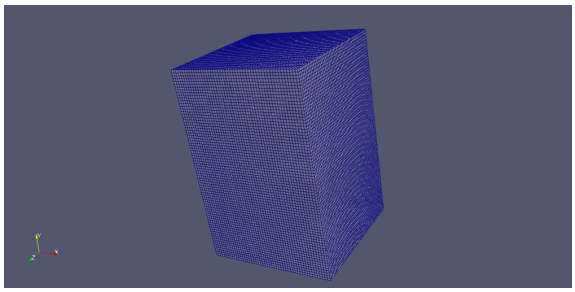

Fig 1: Mesh

immiscible fluids. That means that the material properties are constant in the region filled by one of the two fluid except at the interphase.

2.3.1 Equations

Continuity Equation

$$\frac{\partial u_j}{\partial x_j} = 0 \tag{2.1}$$

Momentum Equation

$$\frac{\partial(\rho u_i)}{\partial t} + \frac{\partial(\rho u_j u_i)}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \frac{\partial(\tau_{tij} + \tau_{ij})}{\partial x_j} + \rho g_i + f_{\sigma i} \tag{2.2}$$

u represent the velocity, g_i the gravitational acceleration, p the pressure and τ_{ij} and $\tau_{t_{ij}}$ are the viscose and turbulent stresses. $f_{\sigma i}$, is the surface tension.

The density ρ is defined as follows:

$$\rho = \alpha\rho_1 + (1 - \alpha)\rho_2 \tag{2.3}$$

is 1 inside fluid 1 with the density ρ_1 and 0 inside fluid 2 with the density ρ_2 . At the interphase between the two fluids α varies between 0 and 1. The surface tension $f_{\sigma i}$, is modelled as continuum surface force (CSF). It is calculated as follows:

$$f_{\sigma i} = \sigma\kappa\frac{\partial\alpha}{\partial x_i} \tag{2.4}$$

σ is the surface tension constant and κ the curvature. The curvature can be approximated as follows:

$$\kappa = -\frac{\partial n_i}{\partial x_i} = -\frac{\partial}{\partial x_i}\left(\frac{\partial\alpha/\partial x_i}{|\partial\alpha/\partial x_i|}\right) \tag{2.5}$$

Interphase Equation

In order to know where the interphase between the two fluids is, an additional equation for α has to be solved.

$$\frac{\partial\alpha}{\partial t} + \frac{\partial(\alpha u_j)}{\partial x_j} = 0 \tag{2.6}$$

The equation can be seen as the conservation of the mixture components along the path of a fluid parcel. [2]

2.4 Case Setup

2.4.1 Boundary Conditions

The velocity boundary condition was set to be **noSlip** for table and walls, and **pressureInletOutletVelocity** for atmosphere. Pressure boundary condition was set **fixedFluxPressure** everywhere except at atmosphere where it is set as **totalPressure**. Alpha.water boundary conditions where set to be **zeroGradient**

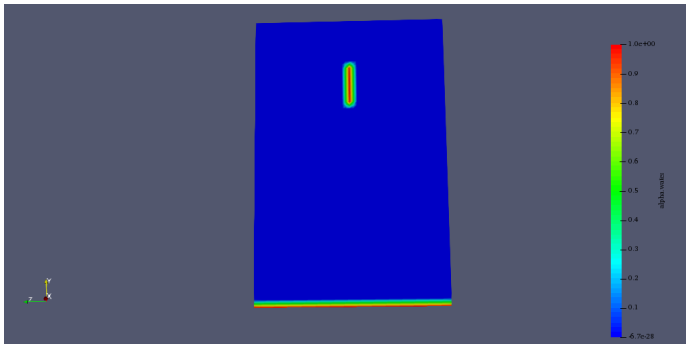

Fig 2: Case Setup

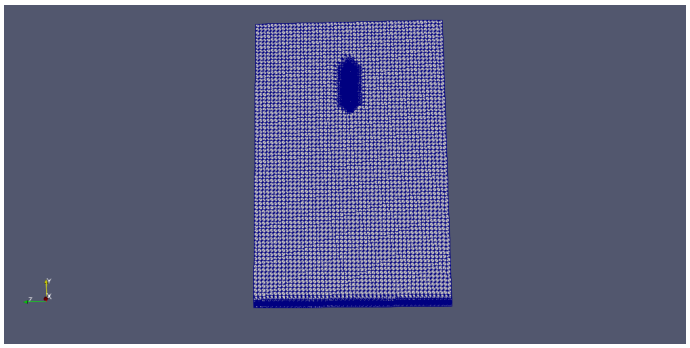

Fig 3: Adaptive Mesh Refinement

at table and walls and **inletOutlet** at atmosphere. Thin (2mm) water bed was set on the table and cylindrical block of water was set 10cm above the bed. This was done using **setFieldsDict** where default field was set to be air. Gravity was set in the negative Y direction. Adaptive mesh refinement has to be specified in **dynamicMeshDict**.

2.4.2 Adaptive mesh refinement

In numerical analysis, adaptive mesh refinement (AMR) is a method of adapting the accuracy of a solution within certain sensitive or turbulent regions of simulation, dynamically and during the time the solution is being calculated. When solutions are calculated numerically, they are often limited to pre-determined quantified grids as in the Cartesian plane which constitute the computational grid, or 'mesh'. Many problems in numerical analysis, however, do not require a uniform precision in the numerical grids used for graph plotting or computational simulation, and would be better suited if specific areas of graphs which needed precision could be refined in quantification only in the regions requiring the added precision. Adaptive mesh refinement provides such a dynamic programming environment for adapting the precision of the numerical computation based on the requirements of a computation problem in specific areas of multi-dimensional graphs which need precision while leaving the other regions of the multi-dimensional graphs at lower levels of precision and resolution. [3]

2.5 Results

As shown in the figure, very good result agreeing with experiments was obtain. The hydraulic jump can be seen in the figure.

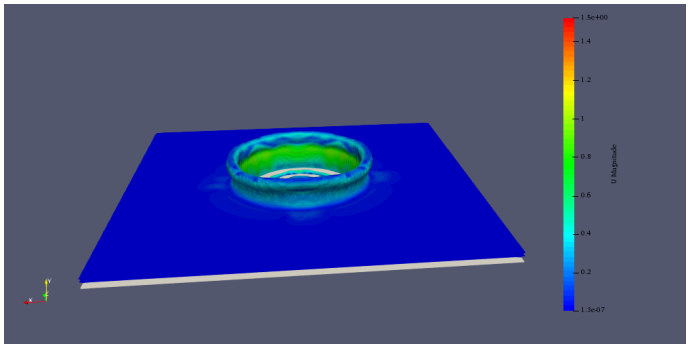

Fig 4: Result

Reference

- https://en.wikipedia.org/wiki/Hydraulic_jump
- https://openfoamwiki.net/index.php/InterFoam
- https://en.wikipedia.org/wiki/Adaptive_mesh_refinement

Summer Fellowship Report

On

Rayleigh-Taylor instability

Submitted by

Raj Niraj Patil

Under the guidance of

Prof. Shivasubramanian Gopalakrishnan
Mechanical Engineering Department
IIT Bombay

July 8, 2019

Acknowledgment

I thank Prof. Shivasubramanian Gopalakrishnan and Prof.Kannan M. Moudgalya for their invaluable guidance. At the same time, I would like to thank Mr Sathish Kanniappan and Ms Deepa Vedartham for their undying support.

These six weeks were one of the most prestigious weeks. I've learnt so much more about CFD, OpenFOAM and other FLOSS. I acknowledge the FOSSEE team for conducting this summer internship. The last but not least, I would like to thank the authority of IIT Bombay for letting us use the resources during this period and helping us from starting day of the internship till the end.

Contents

Chapter 1

Introduction

The RayleighTaylor instability, or RT instability (after Lord Rayleigh and G. I. Taylor), is an instability of an interface between two fluids of different densities which occurs when the lighter fluid is pushing the heavier fluid. Examples include the behavior of water suspended above oil in the gravity of Earth, mushroom clouds like those from volcanic eruptions and atmospheric nuclear explosions, supernova explosions in which expanding core gas is accelerated into denser shell gas, instabilities in plasma fusion reactors and inertial confinement fusion.

Water suspended atop oil is an everyday example of RayleighTaylor instability, and it may be modeled by two completely plane-parallel layers of immiscible fluid, the more dense on top of the less dense one and both subject to the Earth's gravity. The equilibrium here is unstable to any perturbations or disturbances of the interface: if a parcel of heavier fluid is displaced downward with an equal volume of lighter fluid displaced upwards, the potential energy of the configuration is lower than the initial state. Thus the disturbance will grow and lead to a further release of potential energy, as the more dense material moves down under the (effective) gravitational field, and the less dense material is further displaced upwards. This was the set-up as studied by Lord Rayleigh. The important insight by G. I. Taylor was his realisation that this situation is equivalent to the situation when the fluids are accelerated, with the less dense fluid accelerating into the more dense fluid. This occurs deep underwater on the surface of an expanding bubble and in a nuclear explosion.

As the RT instability develops, the initial perturbations progress from a linear growth phase into a non-linear growth phase, eventually developing "plumes" flowing upwards (in the gravitational buoyancy sense) and "spikes" falling downwards. In the linear phase, equations can be linearized and the amplitude of perturbations is growing exponentially with time. In the non-linear phase, perturbation amplitude is too large for the non-linear terms to be neglected. In general, the density disparity between the fluids determines the structure of the subsequent non-linear RT instability flows (assuming other variables such as surface tension and viscosity are negligible here). [1]

Chapter 2

Rayleigh-Taylor instability

2.1 Abstract

The report about the simulation of Reyleigh-Taylor instability of two fluids with density 1020 kg/m^3 and 800 kg/m^3 with heavier liquid being top on the lighter. The result obtained is good but not excellent. It could have been better but it needs high computational power. This case is solved with interFoam with k-Epsilon model. The result fairly agrees with experimental videos taken by several people.

2.2 Geometry & Meshing

It is 20cm x 20cm x 1cm cuboid with only single cell in z direction to make it 2D simulation. Top wall is named as topWall while remaining sides are named as walls. Meshing was done using blockMesh with all sides except front and back being wall and front and back side being empty.

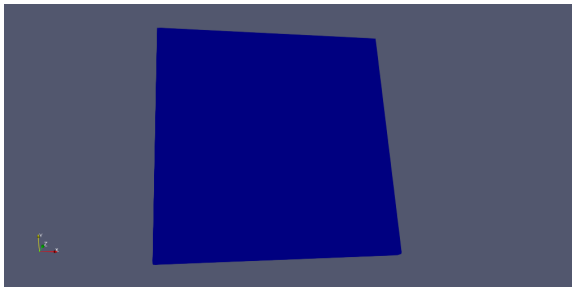

Fig 1: Mesh

2.3 Solver

This simulation is done using interFoam with K-Epsilon model. InterFoam solves for two incompressible immiscible fluids under isothermal conditions using a volume of fluid approach. It also allows us to use mesh motion, mesh topology changes and adaptive re-meshing. The solver solves the Navier Stokes equations for two incompressible, isothermal immiscible fluids. That means that the material properties

are constant in the region filled by one of the two fluid except at the interphase. It should be highlighted that interFoam applies the OpenFOAM specific algebraic VOF scheme called MULES (Multidimensional Universal Limiter with Explicit Solution)for the task of advecting the sharp interface. That it should be noted because of the MULES scheme has a number of desirable properties:

- It preserves the volume of fluid, i.e. it does not artificially create or destroy fluid.
- It keeps the so-called volume fraction field in the physically meaningful range-between 0 and 1.
- The interface stays sharp to within a few cell widths
- It works on unstructured meshes both in 2D and 3D
- It is efficient so only a minor fraction of the calculation time is spent on interfaceadvection. [2]

Concerning the solver algorithm used to solve the types of problems under consideration here, the PISO (Pressure Implicit with Splitting of Operators) loop was chosen.The main reason is that this algorithm works proper for transient calculations and for low Courant numbers (less than one). As it will be check, all the simulations run in the present thesis match with both approaches. The PISO approach was proposed by Issa [3] (1986). It consists on three steps for each time step and the iterations are only needed for the two last steps. One summary of this algorithm can be presented as [4]:

- Momentum predictor: The momentum equations are optionally solved using a best-guess pressure field to produce a best-guess velocity field. However, Issa [3] (1986) notes that while this momentum predictor step is formally required for the method, many applications do not require it, and can proceed straight to the second step using the previous time-steps velocity field.
- Using the previous velocities the pressure field is obtained. Therefore, thefirst estimate of the new pressure field is obtained.
- The velocity field is corrected using the new pressures.

Within each time step, additional equations for multi-phase flow are solved before the PISO algorithm, while other equations, such as turbulence models, are solved afterwards.

2.3.1 Equations

Continuity Equation

$$\frac{\partial u_j}{\partial x_j} = 0 \tag{2.1}$$

Momentum Equation

$$\frac{\partial(\rho u_i)}{\partial t} + \frac{\partial(\rho u_j u_i)}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \frac{\partial(\tau_{tij} + \tau_{ij})}{\partial x_j} + \rho g_i + f_{\sigma i} \tag{2.2}$$

u represent the velocity, g_i the gravitational acceleration, p the pressure and τ_{ij} and $\tau_{t_{ij}}$ are the viscose and turbulent stresses. $f_{\sigma i}$, is the surface tension.

The density ρ is defined as follows:

$$\rho = \alpha\rho_1 + (1-\alpha)\rho_2 \tag{2.3}$$

is 1 inside fluid 1 with the density ρ_1 and 0 inside fluid 2 with the density ρ_2 . At the interphase between the two fluids α varies between 0 and 1. The surface tension $f_{\sigma i}$, is modelled as continuum surface force (CSF). It is calculated as follows:

$$f_{\sigma i} = \sigma\kappa\frac{\partial\alpha}{\partial x_i} \tag{2.4}$$

σ is the surface tension constant and κ the curvature. The curvature can be approximated as follows:

$$\kappa = -\frac{\partial n_i}{\partial x_i} = -\frac{\partial}{\partial x_i}\left(\frac{\partial\alpha/\partial x_i}{|\partial\alpha/\partial x_i|}\right) \tag{2.5}$$

Interphase Equation

In order to know where the interphase between the two fluids is, an additional equation for α has to be solved.

$$\frac{\partial\alpha}{\partial t} + \frac{\partial(\alpha u_j)}{\partial x_j} = 0 \tag{2.6}$$

The equation can be seen as the conservation of the mixture components along the path of a fluid parcel. [5]

2.3.2 K-Epsilon

K-epsilon (k-) turbulence model is the most common model used in Computational Fluid Dynamics (CFD) to simulate mean flow characteristics for turbulent flow conditions. It is a two equation model that gives a general description of turbulence by means of two transport equations (PDEs). The original impetus for the K-epsilon model was to improve the mixing-length model, as well as to find an alternative to algebraically prescribing turbulent length scales in moderate to high complexity flows. [6]

- The first transported variable is the turbulence kinetic energy (k).
- The second transported variable is the rate of dissipation of turbulence energy (ϵ).

For turbulent kinetic energy k

$$\frac{\partial(\rho k)}{\partial t} + \frac{\partial(\rho k u_i)}{\partial x_i} = \frac{\partial}{\partial x_j}\left[\frac{\mu_t}{\sigma_k}\frac{\partial k}{\partial x_j}\right] + 2\mu_t E_{ij}E_{ij} - \rho\varepsilon \tag{2.7}$$

For dissipation ϵ

$$\frac{\partial(\rho\varepsilon)}{\partial t} + \frac{\partial(\rho\varepsilon u_i)}{\partial x_i} = \frac{\partial}{\partial x_j}\left[\frac{\mu_t}{\sigma_\varepsilon}\frac{\partial\varepsilon}{\partial x_j}\right] + C_{1\varepsilon}\frac{\varepsilon}{k}2\mu_t E_{ij}E_{ij} - C_{2\varepsilon}\rho\frac{\varepsilon^2}{k} \tag{2.8}$$

[Rate of change of k or ϵ + Transport of k or ϵ by convection = Transport of k or ϵ by diffusion + Rate of production of k or ϵ - Rate of destruction of k or ϵ]

where,

u_i represents velocity component in corresponding direction

E_{ij} represents component of rate of deformation

μ_t represents eddy viscosity

$$\mu_t = \rho C_\mu \frac{k^2}{\varepsilon} \tag{2.9}$$

The equations also consist of some adjustable constants σ_k, σ_ϵ, $C_{1\epsilon}$ and $C_{2\epsilon}$. The values of these constants have been arrived at by numerous iterations of data fitting for a wide range of turbulent flows. These are as follows:

C_μ= 0.09 σ_k= 1.00 σ_ϵ= 1.30 $C_{1\epsilon}$= 1.44 $C_{2\epsilon}$= 1.92

2.4 Case Setup

2.4.1 Boundary Conditions

All patches with type were set with **slip** velocity boundary condition with respective wall functions in files k, nut and epsilon (**kqRWallFunction** in k file, **nutkWallFunction** for nut and **epsilonWallFunction** for epsilon). The values were made dependent on internal field. **topWall** was kept at fixed pressure where walls were made **zeroGradient**. Alpha.water conditions were **zeroGradient** everywhere. **frontAndBack** was made empty in all of the above files.

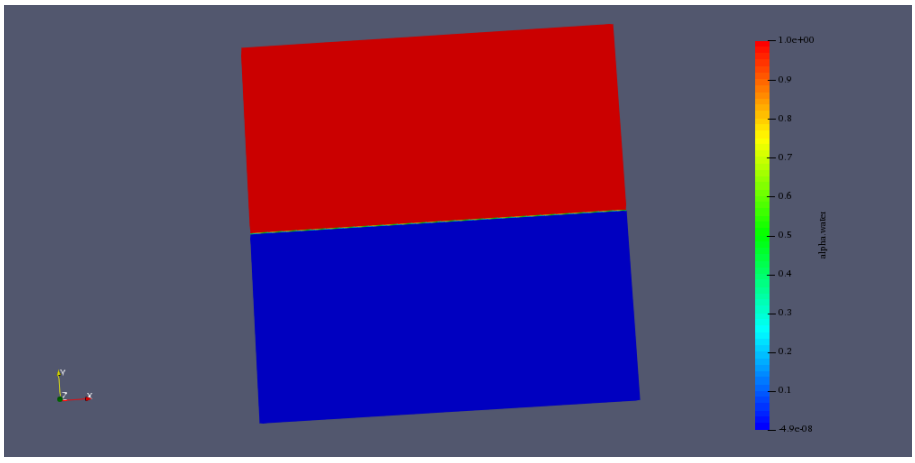

Fig 2: Case setup

2.5 Results

As shown in the following figure, very good results agreeing with experiments were obtain. From the figures shown below, the mushroom formation with heavier liquid penetrating into lighter one can be noticed.

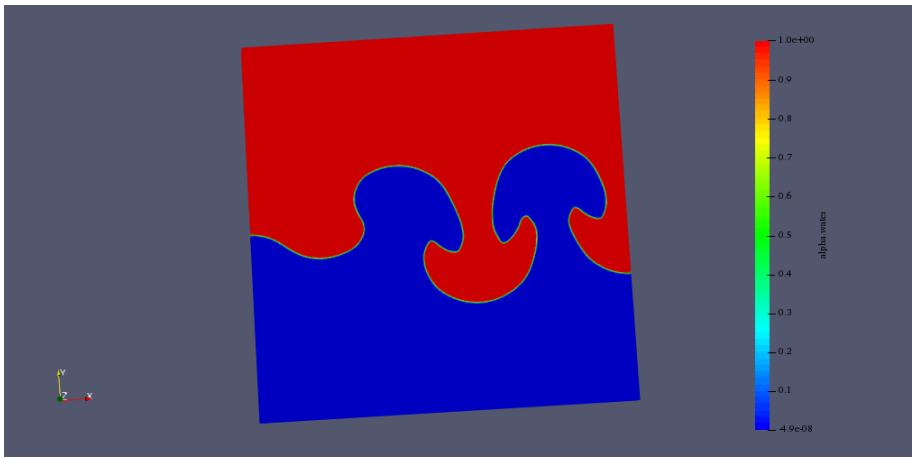

Fig 3a

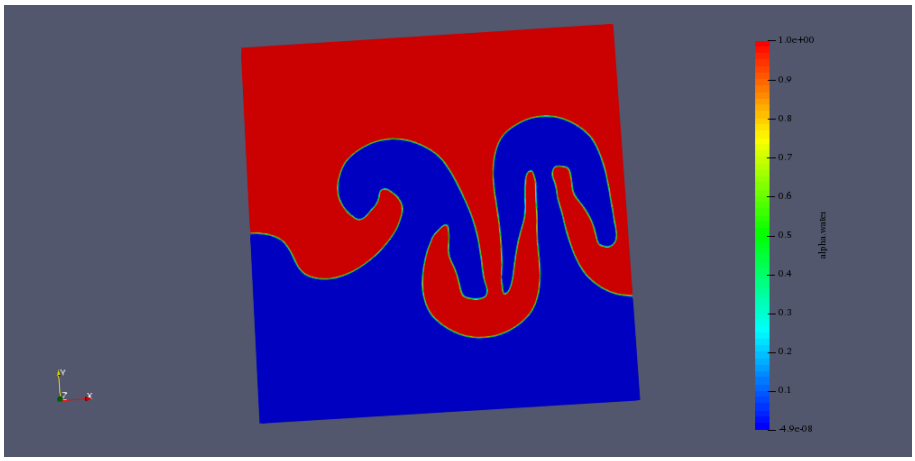

Fig 3b

2.6 Overview

The above case reasonably satisfies the experimental results, but it is not precisely correct. In the actual case, it must locally form the mushrooms and should continue to form until there is no surface present between the liquids (before complete mixing and the final separation).

Reference

- https://en.wikipedia.org/wiki/Rayleigh-Taylor_instability
- Andrs Calvo Luz. Study of bubbly flows in an open-source software. *Aalto University*.
- R.I. Issa. Solution of the implicitly discretised fluid flow equations by operator-splitting.Jpurnal of Computational Physics, 62:4065, January 1986.
- Gerald C.J. Morgan. Application of the interfoam vof code to coastal wave/ structure interaction. Masters thesis, University of Bath Department of Architecture and Civil Engineering, February 2013.
- https://openfoamwiki.net/index.php/InterFoam
- https://en.wikipedia.org/wiki/K-epsilon_turbulence_model