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

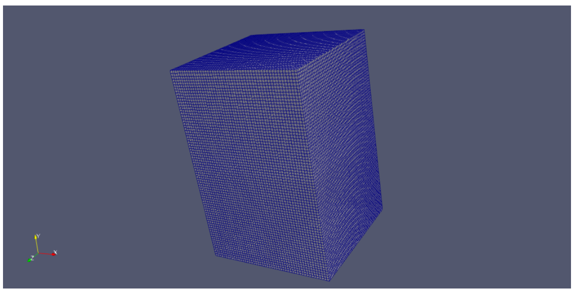

Fig 1: Mesh

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**

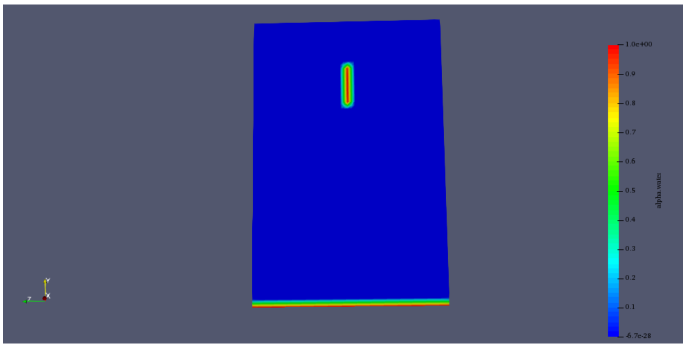

Fig 2: Case Setup

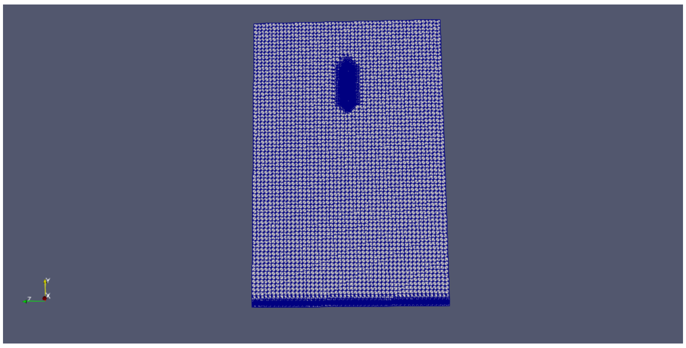

Fig 3: Adaptive Mesh Refinement

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.

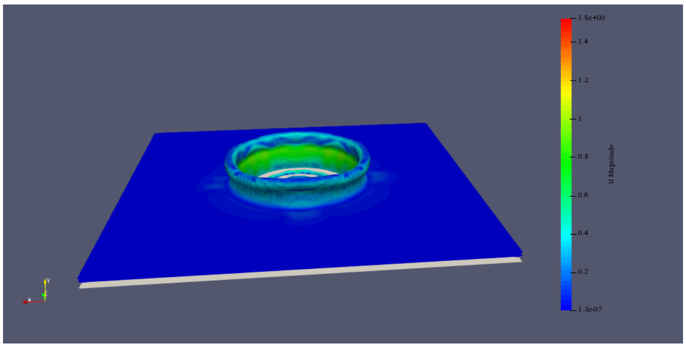

Fig 4: Result

Reference

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