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

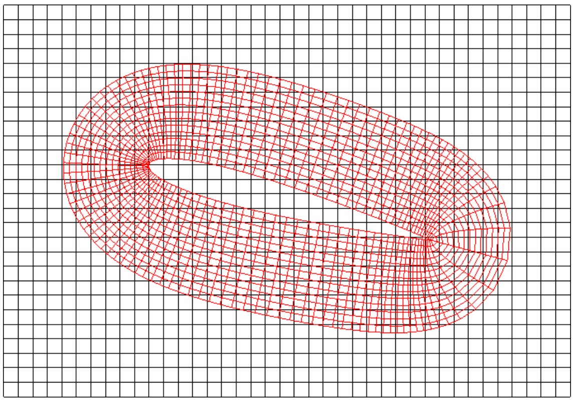

Fig 1: Airfoil grid overlapping background Cartesian grid

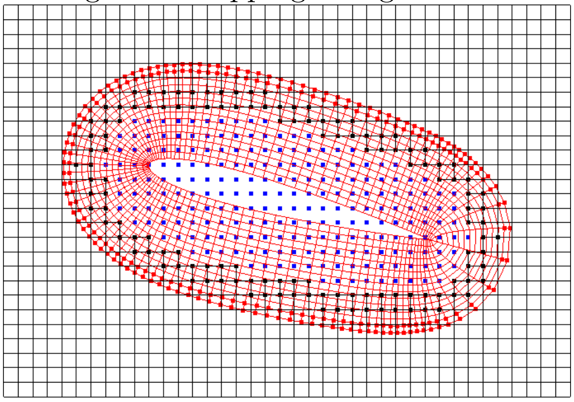

Fig 2: Grid system with hole and fringe points identified after application of process to reduce the amount of overlap between component grids.

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.

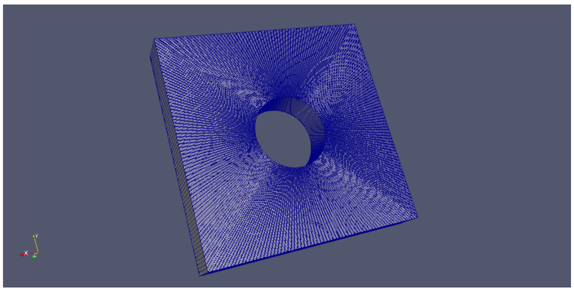

Fig 3: Spherical Submesh

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.

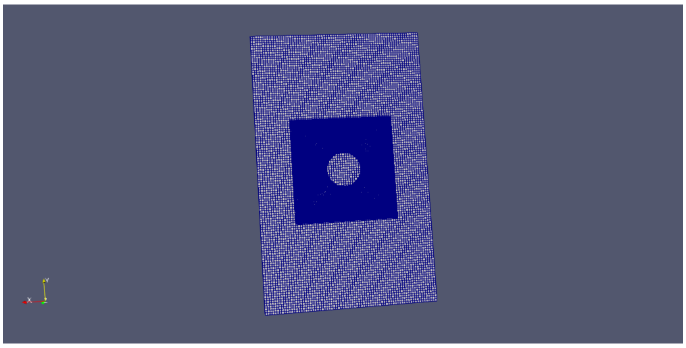

Fig 3: Merged mesh

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