Chapter 2

Creating a Simple Geomtery in OpenFoam

Introduction

In this chapter we will learn how to create a simple geometry and mesh it. Here we will use the lid-driven cavity problem example, mentioned in the previous chapter for pre-processing. As previously mentioned you can type the following path in the command terminal to open the id-driven cavity problem:

```
cd OpenFOAM/OpenFOAM-2.3.0/run/tutorials/incompressible/icoFoam/cavity
```

After this if you type “ls” in the command terminal would see three folder inside it given as:

- 0
- constant
- system

where the 0 folder gives the initial boundary conditions for that particular geometry, constant gives the geomtery file and other fluid properties while system folder gives the number of the iterations the solver would run along other important files. You can find the geometry data of a problem in the polymesh folder within constant. In order to open that, type the following in the command terminal and then press <enter>:

```
cd constant/polymesh
```

Then type ls to in the command terminal and press ¡enter¿. This shows the geomtery file inside polyMesh folder given as blockMeshDict file. In order to view this file type the following in the command terminal:

```
gedit blockMeshDict
```

where gedit it the name of the editor we have used. Note that you may use any other text file editor to view and edit this file.Now you can see the gedit window containing the geometry file and study it in details.

How to Write a Geometry File

In order to draw a geomtery in OpenFoam you need to know some basics regarding OpenFOAM.In OpenFOAM a geomtery is broken down into small hexahedral blocks and are then numbered starting from 0, 1,2,... andso on. For our current problem the block given as shown in the Fig 2.1.

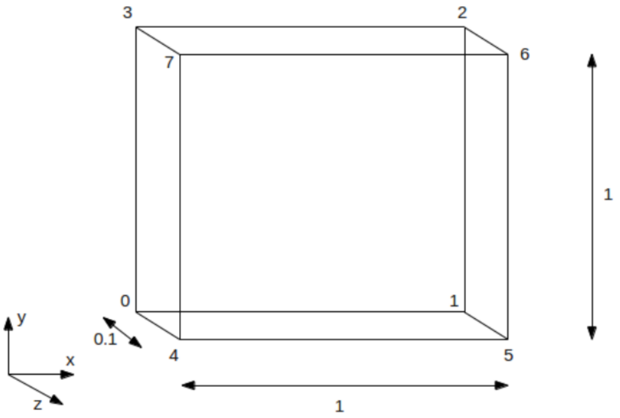


Figure 2.1: Geomtery points of the lid driven cavity

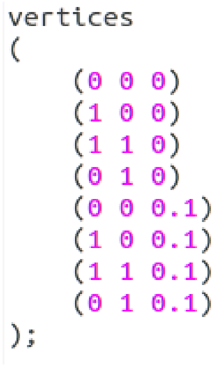
Note that in OpenFOAM to create a 2-D geometry you need to give a unit cell thickness in the Z axis and keep the boundary patches empty. Now in order to create a new geomtry file open a new folder in destop and rename it a “blockMeshDict”.

A blockMeshDict file basically has the following parts:

- Foam File details
- vertices
- blocks
- edges
- boundary
- mergepatchpairs

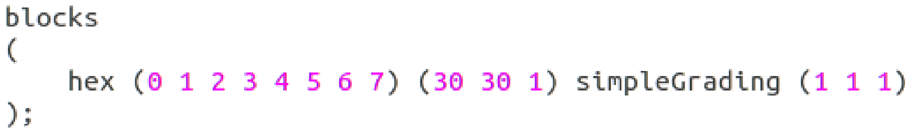
Note that the line convertToMeter gives unit in which the geomtery is to be drawn. For example,for our current problem we are drawing the geomtery in meters so we will keep convertToMeters as 1. Now after opening the new blockMeshDict file created in the desktop, copy the lines from initial Foam File till convertToMeters from the old file and paste it. After this type vertices and then you can give the X, Y and Z co-ordinates of the boundary as shown below:

Then type block, inside which you can give the details of the boundary co-ordinates along with the number of mesh divisions in X, Y and Z direction in the following way:



```
vertices
(
    (0 0 0)
    (1 0 0)
    (1 1 0)
    (0 1 0)
    (0 0 0.1)
    (1 0 0.1)
    (1 1 0.1)
    (0 1 0.1)
);
```

Figure 2.2: Coordinates of boundary geomtery points of the lid driven cavity



```
blocks
(
    hex (0 1 2 3 4 5 6 7) (30 30 1) simpleGrading (1 1 1)
);
```

Figure 2.3: Block details of the geomtery

Here hex represents hexahedral block and the numbers inside first braces, next to that gives the names of the points at the boundary in clock-wise direction to form a block. Note that for more than one blocks the number of points would be more. The number of grid points can be modified as per requirement. For this problem we have used a 2-D mesh having 30x30 divisions and unit dept. Now since we have all straight edges in this geometry, we will keep the egdes empty.

```
edges
(
);
```

Figure 2.4: edge details of the geomtery

Next we give the details of the boundary conditions of the geometry. In this current problem we can see the following boundary conditions as shown below.

- moving wall
- fixed wall
- front and back

where it has a moving wall at theb top and three fixed wall. The front and back faces are kept empty as this is a 2-D problem. Now in the blockMeshDict file you can type the boundary as following:

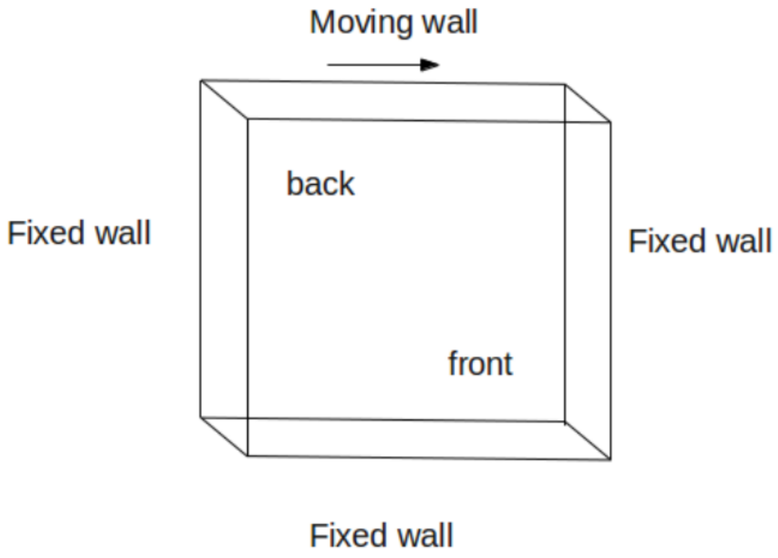


Figure 2.5: Boundary names of the lid driven cavity problem

Here within the boundary names enter the type of boundary you wish to use and then faces, giving the points of the block forming a particular boundary. Note that you should be very careful while giving the order of the points. It should be given in such a way that if you place a folded palm on the surface of a boundary the thumb should be pointing normal to the surface and the fingers should be folded such that they make a curl in clockwise or anti-clockwise direction. Note that you should use either clockwise or anti-clockwise convention throughout the file and but not both at a time. Also you should be very careful regarding opening and closing of brackets in these files.

After this, in a new line type mergePatchPairs. Since in this problem we do not have to merge any patches we will keep this empty as given below.

After completing writing save it and close this file. Thus you have learned how to create a geomtery file.

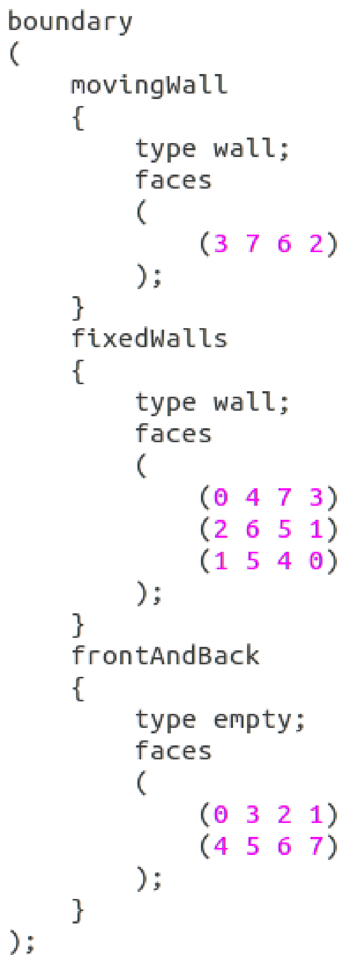
Now go back to the command terminal and type the following twice to go back to cavity folder:

```
cd ..
```

Next you can mesh this geomtry by typing blockMesh in the command terminal.This command would calculate the grid lines, number of cells, mesh details, etc for this particular geometry.

Post-Processing

After this you can view the geometry by opening paraview. For this, type paraFoam in the command terminal and press <enter>.

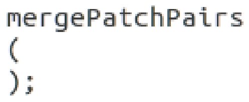


```
boundary
(
    movingWall
    {
        type wall;
        faces
        (
            (3 7 6 2)
        );
    }
    fixedWalls
    {
        type wall;
        faces
        (
            (0 4 7 3)
            (2 6 5 1)
            (1 5 4 0)
        );
    }
    frontAndBack
    {
        type empty;
        faces
        (
            (0 3 2 1)
            (4 5 6 7)
        );
    }
);
```

Figure 2.6: boundary details of the geomtery

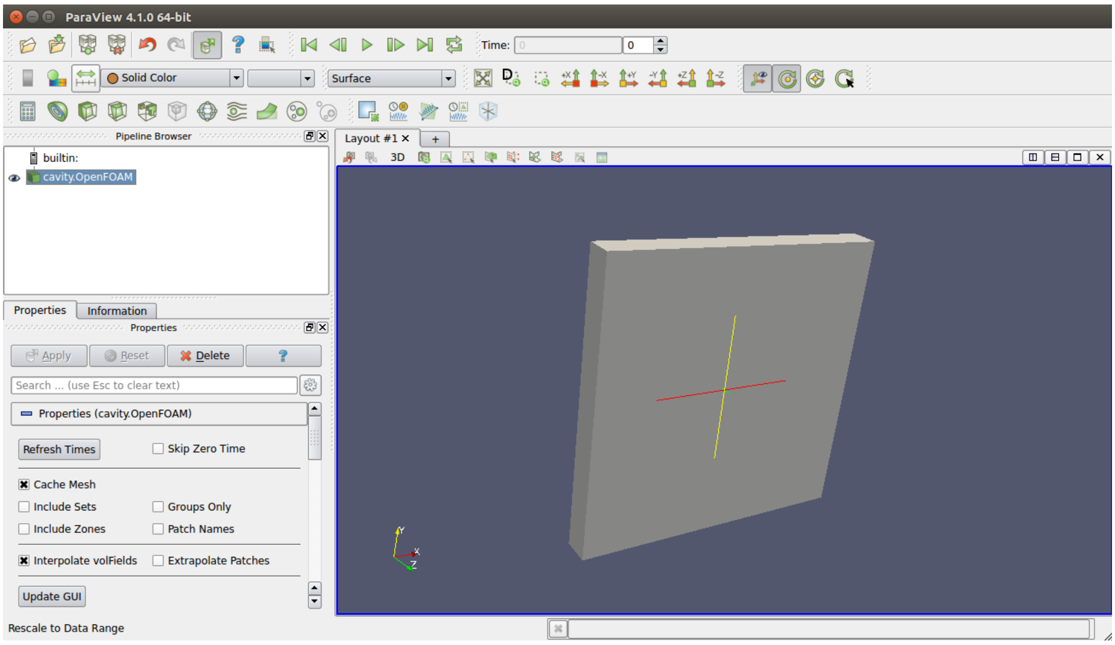
In the paraview window press Apply button on the left hand side(to include the geometry file in ParaFoam) of the Object Inspector Menu to view the Geometry, as shown in the Fig 2.8 :

As you have learned in the previous chapter, you can use different feature in the paraview window to check the details of the geometry.



```
mergePatchPairs
(
);
```

Figure 2.7: merge patch details of the geomtery



☐ Skip Zero Time

☒ Cache Mesh

☐ Include Sets ☐ Groups Only

☐ Include Zones ☐ Patch Names

☒ Interpolate volFields ☐ Extrapolate Patches

Figure 2.8: Paraview window showing the 2-D geometry