

Study of Flow Pattern and Lift and Drag on a cylinder with two inlets at diametrically opposite points with parabolic velocity profile at inlet

Shouvik Ghorui

B.E, Jadavpur University, Kolkata

Abstract

The present paper deals with the numerical simulation of unsteady, two-dimensional laminar fluid flow over a circular cylinder. Effects of injection of a secondary similar fluid from two diametrically opposite peripheral slots on the cylinder are of prime interest in the study. The injection arrangements have been investigated in a cross-flow arrangement, in which the second fluid is injected perpendicular to the free stream flow direction. Free stream velocity enters with a parabolic velocity profile through the primary inlet. A parametric variation of the velocity of the injected stream in the downstream wake region of the cylinder has been studied qualitatively by observing the velocity, vorticity & streamline and along with it a basic understanding and intuition about how mixing occurs within a combustor. The injection arrangements affected the vortex shedding in distinctly different manners. The geometry and meshing of the domain are created using blockMesh utility. The simulations are performed using OpenFOAMv7.

keyword: CFD, OpenFOAM, parabolic velocity, secondary inlet

1. Introduction

Mixing is very important for proper combustion and flame stabilization. In this paper, the aim is to find out the suitable injection velocity so that mixing may be proper. The purpose of this case study is to learn OpenFOAM software and understand the flow characteristic of 2D flow over a cylinder with two diametrically opposite injection ports. Numerical simulations of hydraulic problems such as flow over a cylinder with two inlets are performed using computational fluid dynamics. The simulations are performed using OpenFOAM-v7. In this case, simulation approaches are considered using pimpleFoam solver in **OpenFOAM-v7**.

Flow over cylinder is a very common problem. Adding secondary inlets to cylinder has been explored using FLUENT by Achintya et al. [4] This is attempted here using OpenFOAM.

This case study demonstrates how to do the following:

- Set up a problem case
- Create a cylindrical geometry using blockMesh
- Mesh the region outside a truncated cylinder using blockMesh
- Set up the properties of the fluids

- Initialize the flow
- Consider the laminar model for all flow regime
- Post-processing the case for results

2. Problem statement:

This case considers the flow over a Cylinder with 2 inlets at diametrically opposite points confined between two parallel plates. Fluid enters in between the parallel plates with fixed parabolic velocity (U , m/s) inlet.

Figure 1 shows the geometry of two parallel plates with a Cylinder with 2 inlets at diametrically opposite points considered in the present study. The fluid will flow through the inlet of parallel plates and same type (in the property) of fluid will also flow through the cylinder's two inlets and the two fluid streams will mix with each other. The geometrical parameters and flow conditions are shown as in Table 1 and Table 2 respectively.

Parameters	Value
distance between two parallel plates(H), cm	60
Length of the two plates(L), cm	200
Radius of the cylinder (R), cm	3
Length of the inlet2 & inlet3 (on the cylinder)	2

Table 1: Details of geometrical parameters

Parameters	Value
Density of the fluid	1000 kg/m ³
Dynamic viscosity of the fluid	1.5×10^{-3} Pa.s

Table 2: Details of fluids property

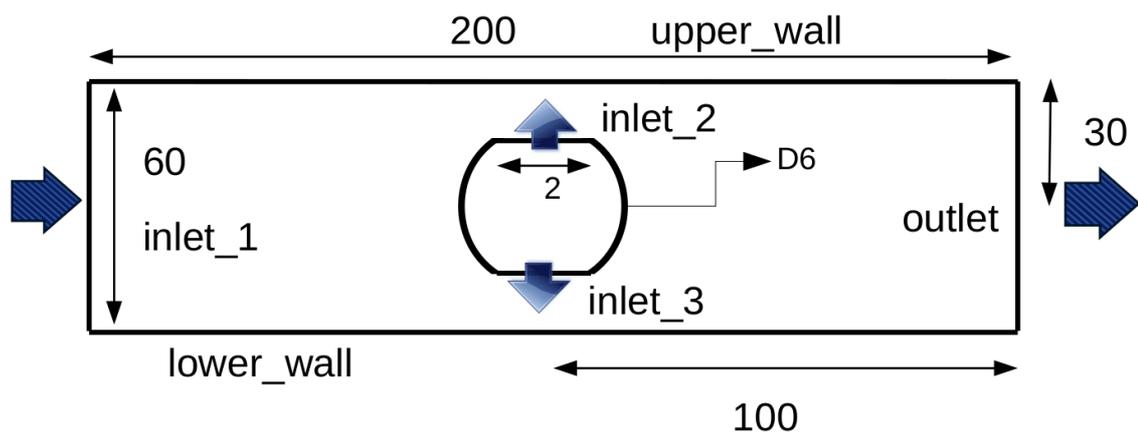


Figure 1: Schematic diagram of the geometry

We will see how the velocity, lift & drag will change by changing the ratio of velocity inlet through the plate (U_∞) (free stream velocity) to the Velocity through the cylinder inlet (injected stream velocity) (v_2 & v_3).

The velocity ratio ε , is defined as the ratio of the injected stream velocity to the free stream velocity, and is expressed as

$$\varepsilon = \frac{v_f}{U_\infty}$$

3. Mathematical modelling:

PimpleFoam (solver) solves the Navier-Stokes equations for isothermal and incompressible flow.

3.1 Continuity equation

The continuity equation (constant-density) is defined as:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0$$

3.2 Momentum equation

x-Momentum:

$$\frac{\partial u}{\partial t} + \frac{\partial(uu)}{\partial x} + \frac{\partial(uv)}{\partial y} = -\frac{1}{\rho} \frac{\partial p}{\partial x} + \nu \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right)$$

y-Momentum:

$$\frac{\partial v}{\partial t} + \frac{\partial(uv)}{\partial x} + \frac{\partial(vv)}{\partial y} = -\frac{1}{\rho} \frac{\partial p}{\partial y} + \nu \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right)$$

3.3 Lift Force

The lift force for incompressible, laminar, two-dimensional flow past a circular cylinder can be expressed as:

$$F_L = \frac{1}{2} \int_0^{2\pi} (-p \sin(\sin \theta) + \nu \omega \cos(\cos \theta)) D d\theta$$

$$\omega = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}$$

The corresponding lift coefficient is obtained as:

$$C_L = \frac{F_L}{\frac{1}{2} \rho U_\infty^2 D}$$

4. Simulation procedure:

The case deals with 2D laminar flow simulation of liquid over a cylinder having 2 inlets. First, copy a **pimpleFoam** tutorial case from the tutorial folder and paste the folder where we want to keep our simulation file. Then we need to set all required input parameters before starting the

simulation. Also, we need to write the blockMeshdict file as our requirement. This case study is considered with the transient flow. And we use a pimpleFoam solver in this case.

4.1 Creating geometry and mesh

- Mesh can be generated using 'blockMesh' utilities in OpenFOAM.
- The region around the cylinder is constructed with 8 blocks as shown (in the below picture) and another two blocks constituting the entire domain is described
- Grading is applied to the blocks around the cylinder and mesh near the cylinder is made finer in order to capture the intricate physics there.
- Figure 2 shows the isometric view of the generated mesh using 'blockMesh' utility.

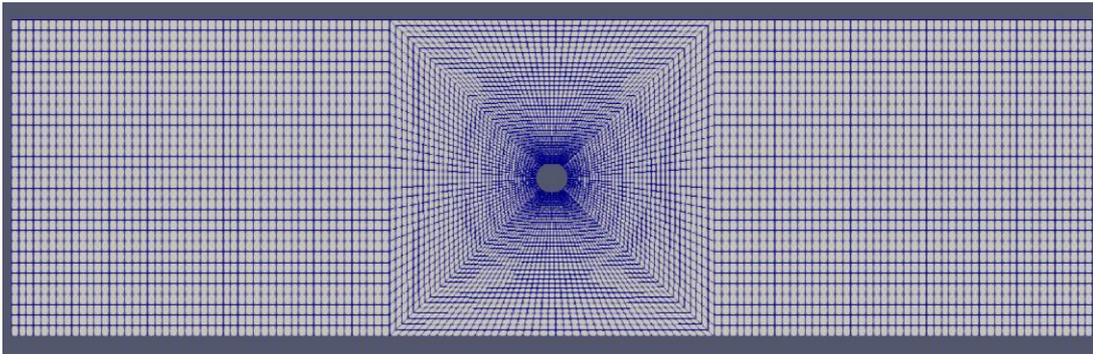


Fig 2: 2D Mesh using blockMesh

4.2 Setting boundary conditions (BC) / Initial conditions (IC):

There are two files in the 0-folder name one is 'U' and the other one is 'p'. Boundaries are assigned and added nine boundaries of the present case in two the files, i.e. 'U' & 'p'. Details of the boundary conditions are listed in Table 4.

Boundary	U	p
inlet	parabolicVelocity	zeroGradient
inlet_2	fixedValue	zeroGradient
inlet_3	fixedValue	zeroGradient
outlet	zeroGradient	fixedValue
Lower_wall	noSlip	zeroGradient
Upper_wall	noSlip	zeroGradient
cylinder	fixedValue	zeroGradient

Table 4: Details of boundary conditions

4.3 Creating parabolic Velocity:

This type of technique for creating the parabolic velocity profile was originally developed by professor **Hrvoje Jasak**.[\[4\]](#)

The implementations of the boundary conditions are located in:

```
$FOAM_SRC/finiteVolume/fields/fvpatchFields/fvPatchFields
```

The standard procedure when implementing a new boundary condition is to find one that is already implemented and does almost what you want, copy that to your user directory and do the required modifications (change names of folder, files, class, member functions, and member data). We will not do that here since we will build a boundary condition more or less from scratch.

→ Initialize and prepare

Start by creating directory and template files. We will first do this as a separate library and later merge it into our own finiteVolume library.

```
mkdir $WM_PROJECT_USER_DIR/src
cd $WM_PROJECT_USER_DIR/src
foamNewBC -f -v parabolicVelocity
```

This creates the following templated structure:

```
parabolicVelocity
├── Make
│   ├── files
│   └── options
├── parabolicVelocityFvPatchVectorField.C
└── parabolicVelocityFvPatchVectorField.H
```

These are just templates, so we need to do some modifications.

Modify the parabolicVelocityFvPatchVectorField.H file (declarations) &

Modify the parabolicVelocityFvPatchVectorField.C file (definitions)

Type below command in the terminal in order to give the author's name in the file.

```
sed -i s/'OpenFOAM Foundation'/'Original author'/g
parabolicVelocity/parabolicVelocityFvPatchVectorField.H
```

→ Compile the library

```
wmake parabolicVelocity
```

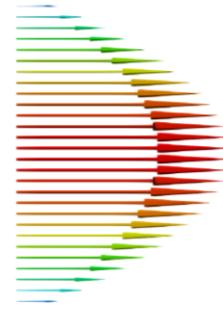
Check that the libparabolicVelocity.so the file ends up in your user directory.

Change the inlet boundary condition in 0/U to:

```

inlet
{
  type parabolicVelocity;
  n (1 0 0);
  y (0 1 0);
  maxValue 1.5;
}

```



The contents of this entry must be in accordance with the constructor that reads the dictionary (see above). The vector 'n' is the direction of the flow. The vector 'y' is the coordinate direction of the profile. The scalar 'maxValue' is the centerline velocity.

→ Add a line at the end of the system/controlDict:

```
libs("libparabolicVelocity.so")
```

This makes the solve aware of the new library so that it can be used.

4.4 Solver details

In this case, we need to analyze transient flow for an incompressible fluid and hence we have used pimpleFoam solver. PimpleFoam is a transient solver for incompressible, turbulent flow of Newtonian fluids, with optional mesh motion and mesh topology changes. The PIMPLE Algorithm is a combination of PISO (Pressure Implicit with Splitting of Operator) and SIMPLE (Semi- Implicit Method for Pressure-Linked Equations). All these algorithms are iterative solvers but PISO and PIMPLE are both used for transient cases whereas SIMPLE is used for steady-state cases.

4.5 Post-processing

ParaFoam can be used to visualize the simulation results in OpenFOAM. This can be run by typing the **paraFoam** command in the terminal to open the Paraview software and upload the case. We have also used **Tecplot 360** for post-processing.

5. Results and discussion:

A plot of Cd and Cl with respect to time and streamlines for a normal flow over a cylinder (von Kármán flow) is shown below for the purpose of comparison. As is evident from the plots, the Cd Cl values are larger for a complete cylinder than for a cylinder with two inlets. This may be due to the fact that the fluid injected from the two inlets mixes with the main freestream flow to form complex vortex patterns behind the cylinder which affects both lift and drag.

Simulations are performed using OpenFOAM to investigate the flow over a cylinder with 2 inlets within 2 parallel plates as well as for flow over a cylinder (von Kármán flow). Results are plotted with the help of 'paraview' (`_/_case/paraFoam`). Simulation results are analyzed with the help of paraFoam software.

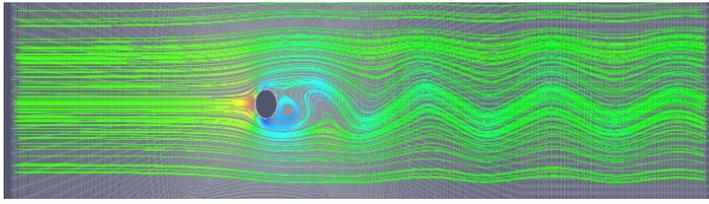


Fig 3: velocity streamlines for flow over a cylinder (Kármán vortex)

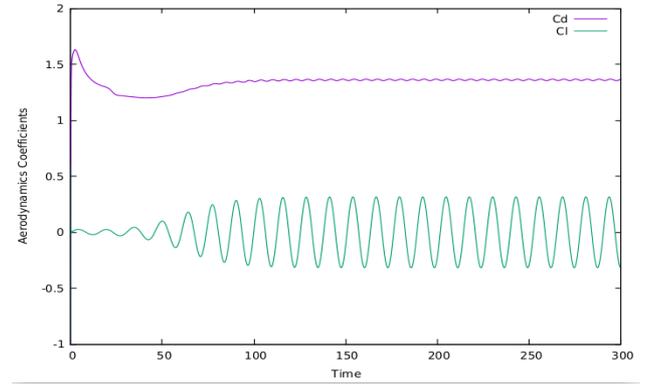


Fig 4: Cd & Cl vs Time plot for flow over a cylinder

The velocity ratio, ϵ , is defined as the ratio of the injected stream velocity to the free stream velocity, and is expressed as

$$\epsilon = \frac{V_f}{U_\infty}$$

- For three different velocity ratios of the injection

For the velocity ratio $\epsilon=0.5$ at $t=300s$

velocity contour, streamlines are shown at time =300. And lift and drag coefficient are also shown.

Fig 5: velocity contour for $\epsilon=0.5$

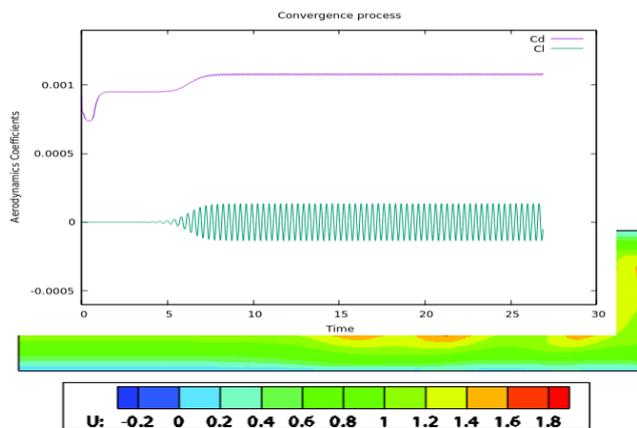


Fig 7: Cd & Cl plot for $\epsilon=0.5$

Fig 6: streamline for $\epsilon=0.5$

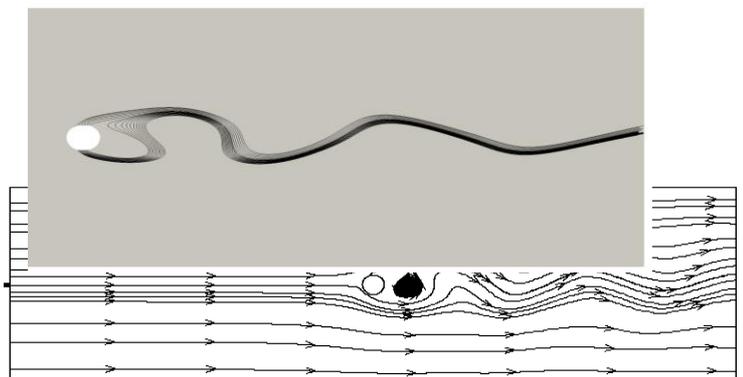


Fig 8: contribution of the secondary inlet in the downstream wake region for $\epsilon=0.5$

For velocity ratio $\epsilon=1$ at $t=300s$

velocity contour, streamlines are shown at time =300. And lift and drag coefficient are also shown.

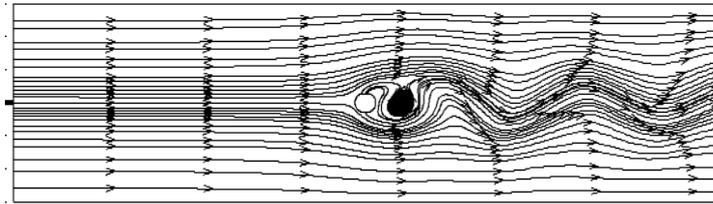
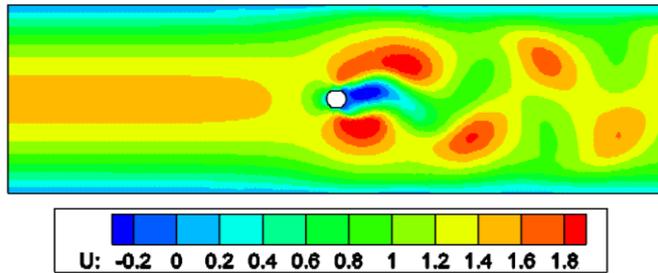


Fig 9: velocity contour for $\epsilon=1$

Fig 10: streamlines for $\epsilon=1$

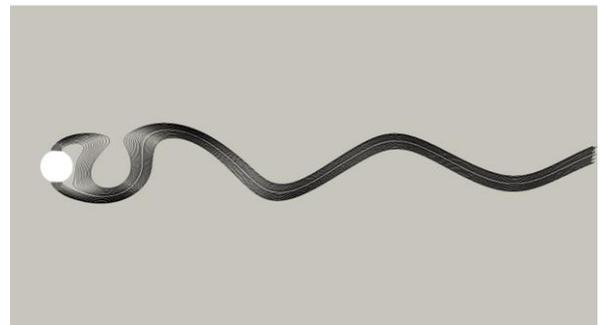
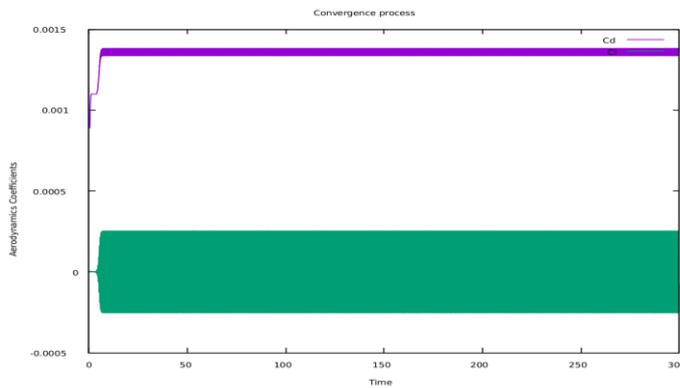


Fig 11: Cd & Cl plot Vs time for $\epsilon=1$

Fig 12: contribution of the secondary inlet in downstream wake region for $\epsilon=1$

In the above two cases($\epsilon=1,0.5$), the quality of mixing will be poor and the mixing will occur within a narrow zone. Hence the stability of the flame will be very less.

For velocity ratio $\epsilon=8$ at $t=300s$

velocity contour, streamlines are shown at time =300. And lift and drag coefficient are also shown.

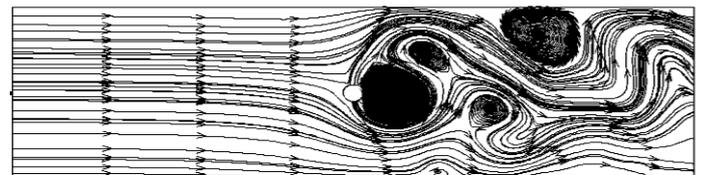
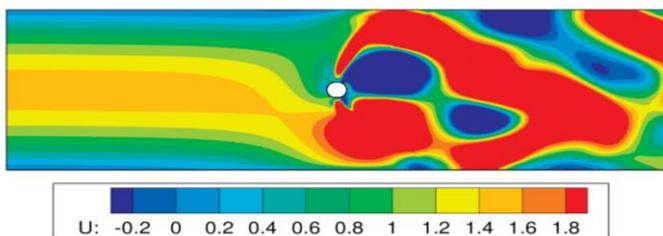


Fig 13: velocity contour for $\epsilon=8$

Fig 14: streamline for $\epsilon=8$

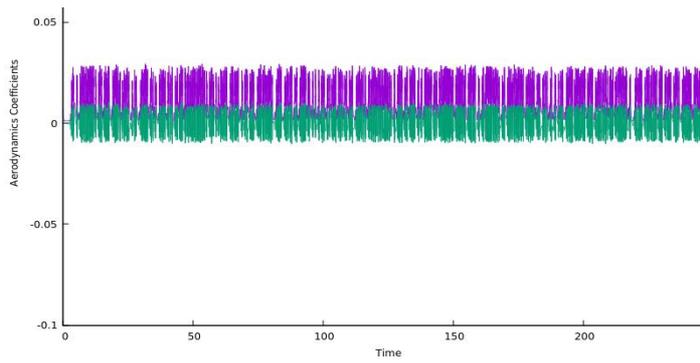


Fig 15: Cd & Cl Vs time plot for $\epsilon=8$

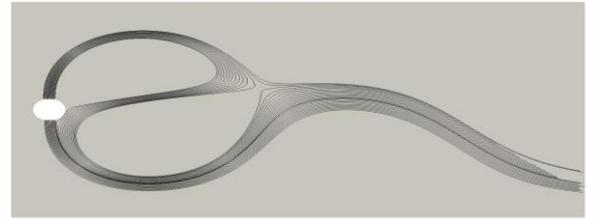
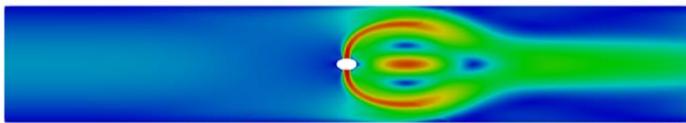


Fig 16: contribution of the secondary inlet in downstream wake region for $\epsilon=8$

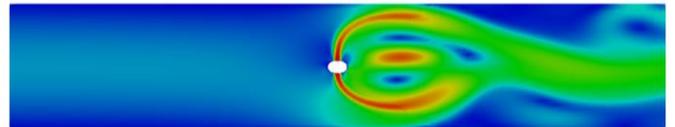
As evident from **Fig 16**, for this case($\epsilon=8$), mixing will be better than the other two cases ($\epsilon=1,0.5$) and the mixing will occur within a wide range. Hence the stability of the flame will increase.

Now the velocity contour for various time step for $\epsilon=8$

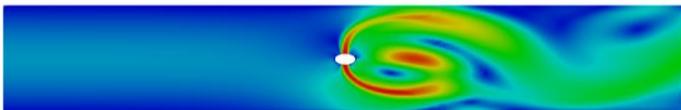
At t = 0.5



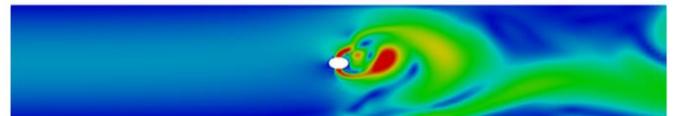
At t=3



At t = 6



At t=24



At t= 180
At

t=240

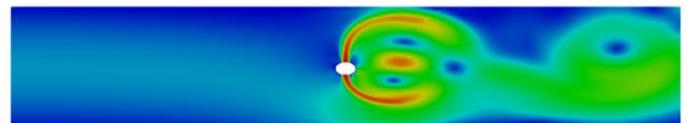
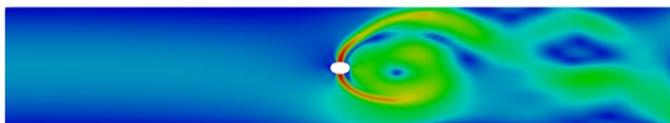


Fig 17: velocity contour at different time step

Conclusion:

By studying the different contours for the different cases, we infer that with the increase in ϵ , the amplitude of the aerodynamic coefficients C_d and C_l about their mean value increases. The effect of the two secondary inlets becomes more prominent as we increase ϵ .

The above statement is very important from the design point of view as fluctuating loads cause more stress and affects the lifetime of a particular component. Here the importance of the cylinder in the flow field has to be understood. The cylinder acts as an obstacle in the free stream flow and also as an injector of fuel and all this form a complex pattern of vortices at the wake region which contributes a lot towards flame stabilization. From the above discussions, contours and streamlines, it is seen that by increasing the velocity ratio ϵ , the range of mixing will be wider and hence will give more flame stability.

For ϵ less than or around 1, there is practically no effect of the secondary inlets. The vortex shedding also follows a pattern which would have been followed without their presence. As we increase ϵ we see that the vortex shedding does not follow a fixed sinusoidal pattern which shows the predominating effect of the secondary inlets at higher ϵ .

Acknowledgement:

I would like to express my sincere gratitude to Prof. Achintya Mukhopadhyay of Jadavpur University for providing me the opportunity to work on this project. I deeply thank my friend Mr. Riddhideep Biswas for his dedication and perseverance and his huge contribution towards this project.

References:

1. C. J. Greenshields, OpenFOAM: The OpenFOAM Foundation. User Guide Version 6. CFD Direct Limited, July. 2018.
2. Numerical Heat Transfer and Fluid Flow by SUHAS.PATANKAR
3. <https://pingpong.chalmers.se/public/courseId/7056/publicPage.do?item=3209036>
4. <https://www.sciencedirect.com/science/article/abs/pii/S0997754615302739>