



UNIVERSITY OF ROSTOCK

COMPUTATIONAL SCIENCE AND ENGINEERING

MODELING AND SIMULATION OF TURBULANCE

SIMULATION OF FLOW OVER A RECTANGULAR CAVITY

Author:

Amit Kumar
Deepanshu Khare
Ujjwal Verma
Lennart Kappis

Supervisor:

Dr.-Ing. Johann Turnow

Abstract

In former time, scientists and engineers were incapable to predict the flow pattern. But due to advancement in technology and introduction of simulations software, now it is easy to determine the fluid flows (laminar and turbulent) and their behaviors. While calculating the heat transfer and pressure it is important to know, if a fluid flow is laminar, transitional or turbulent.

In this project, system will discuss the laminar and turbulent flow over a rectangular cavity based on the laminar, $k-\epsilon$ and $k-\omega$ -SST models and also the influence of coarse and fine Meshes on result.

Moreover, system will also review the pressure distribution, Velocity distribution and Streamlines of given rectangular cavity. In the end, comparison of all the models will be taken care of.

Contents

List of Figures	3
List of Tables	4
1 Introduction	1
1.1 Computational Fluid Dynamics	1
1.2 OpenFOAM	2
2 Geometry and Grid Generation	3
2.1 Geometry	3
2.2 Generation of Coarse and Fine Mesh	4
2.2.1 blockMeshdict	5
2.2.2 convertToMeters	5
2.2.3 vertices	5
2.2.4 edges	5
2.2.5 blocks	6
2.3 Computational Domain	7
2.4 Mesh Resolution	8
2.5 Calculation of Reynolds Number	8
2.6 Calculation of $k - \epsilon$ Model	8
2.7 Calculation of SST $k - \omega$ Model	9
3 The Initial Conditions and Numerical Environment	10
3.1 Laminar Model	10
3.2 Turbulent Model	11
3.2.1 $k - \epsilon$ model	11
3.2.2 SST $k - \omega$ model	11
3.3 Partial Differential Equations	11
3.3.1 $k - \epsilon$ Model	11
3.3.2 SST $k - \omega$ Model	12
3.4 Momentum Equation	13
3.5 Initial Conditions	14
4 Reynolds Averaged Navier Stokes (RANS) equations	15
4.1 Linear eddy viscosity model	16
4.2 Non-linear eddy viscosity model	17
4.3 Reynolds stress models	18

5	Grid resolution near the wall and yPlus estimation	19
5.1	Grid Resolution	19
5.2	yPlus Estimation	20
6	Results and Discussion	23
6.1	k - ϵ Model with Coarse Meshing	23
6.1.1	Velocity Distribution	23
6.1.2	Pressure Distribution	23
6.2	k - ϵ Model with Fine Meshing	24
6.2.1	Velocity Distribution	24
6.2.2	Pressure Distribution	24
6.3	Comparison of K - ϵ with coarse and fine	25
6.4	SST k - ω model with Coarse Meshing	26
6.4.1	Velocity Distribution	26
6.4.2	Pressure Distribution	26
6.5	SST k - ω with Fine Meshing	27
6.5.1	Velocity Distribution	27
6.5.2	Pressure Distribution	27
6.6	Comparison of K - ω with coarse and fine	28
6.7	Laminar Model with Coarse Meshing	29
6.7.1	Velocity Distribution	29
6.7.2	Pressure Distribution	29
6.8	Laminar Model with Fine Meshing	30
6.8.1	Velocity Distribution	30
6.8.2	Pressure Distribution	30
6.9	Comparison of Laminar with coarse and fine	31
7	Comparison of Different Models	32
7.1	Difference between Coarse and Fine Grid	32
7.2	Difference b/w the Results of Different Models	32
8	Conclusion	34
9	References	35
9.1	References	35
9.2	Text References	35
9.3	Figure References	36

List of Figures

2.1	The geometrical description	3
2.2	Grid Generation	4
2.3	Course Mesh	4
2.4	Fine Mesh	5
2.5	The four turbulent flow regimes[1]	7
5.1	The four turbulent flow regimes[1]	19
5.2	law of wall function[2]	20
6.1	Velocity in k - ϵ with coarse mesh	23
6.2	Pressure iso-surface in k - ϵ with coarse mesh	23
6.3	Velocity in k - ϵ with fine mesh	24
6.4	Pressure iso-surface in k - ϵ with fine mesh	24
6.5	Mesh Resolution in k - ϵ Model	25
6.6	Velocity in SST k - ω with coarse mesh	26
6.7	Pressure iso-surface in SST k - ω with coarse mesh	26
6.8	Velocity in SST k - ω with fine mesh	27
6.9	Pressure iso-surface in SST k - ω with fine mesh	27
6.10	Mesh Resolution in SST k - ω Model	28
6.11	Velocity in laminar with coarse mesh	29
6.12	Pressure iso-surface in laminar with coarse mesh	29
6.13	Velocity in laminar with fine mesh	30
6.14	Pressure iso-surface in laminar with fine mesh	30
6.15	Mesh Resolution in Laminar Model	31
7.1	Comparison of Different Models	33

List of Tables

2.1	The values of geometrical parameters	3
3.1	Boundary conditions for the given geometry	14

Chapter 1

Introduction

1.1 Computational Fluid Dynamics

Computational fluid dynamics, generally abbreviated as CFD, uses numerical analysis and algorithms to resolve and analyze the problems that involve fluid flows. But it's not only limited to fluid dynamics, it mainly concerns with computational transport phenomenon i.e., it also involves heat and mass transfer or any other transport phenomenon. CFD is very much based on Navier-Stokes equation. These equations describe how velocity, pressure, temperature, and density of a moving fluid are related. It provides a qualitative (sometimes even quantitative) prediction of fluid flows by means of

- Mathematical modeling (partial differential equation)
- Numerical Methods (discretization and solution techniques)
- Software tools (solvers, pre- and post processing utilities)

The stability of the discretization is generally established numerically rather than analytically as with simple linear problems. Special care must also be taken to ensure that the discretization handles discontinuous solutions gracefully. Some of the discretization methods used are:

- Finite volume method
- Finite element method
- Finite difference method
- Spectral element method
- Boundary element method
- High-resolution discretization schemes

Computers are used to perform the calculations required to simulate the interaction of liquids and gases with surfaces defined by boundary conditions. With high-speed supercomputers, better solutions can be achieved. Ongoing research yields software that improves the accuracy and speed of complex simulation scenarios such as turbulent flows.

1.2 OpenFOAM

OpenFOAM symbolize as "Open source Field Operation And Manipulation" which is mostly used for solving Computational Mechanics problems with numerous solvers and also pre-processing and post-processing the problems. It has over 80 solver applications that simulate specific problems in engineering mechanics and over 170 utility applications that complete pre- and post-processing tasks, e.g. meshing, data visualization, etc. It has an extensive range of attributes to solve anything from complex fluid flows involving chemical reactions, turbulence and heat transfer, to acoustics, solid mechanics and electromagnetics[1].

OpenFOAM is first and foremost a C++ library which is used primarily to create applications. These applications fall into two categories:

- Solvers: To solve a specific problem in continuum mechanics
- Utilities: Designed to perform tasks which involve data manipulation

Main advantage of OpenFOAM is that it gives liberty to user to create new solvers and utilities. Solvers perform the actual calculation to solve a specific continuum mechanics problem. "SimpleFOAM" is one of the openFOAMs steady state solver applied for incompressible turbulent flow that is being used in this simulations. System predicts the pressure and velocity of fluid by using numerical methods. Different discretization and solution techniques are used for solving the turbulent flow of fluid. Behavior of fluid through the cross-section of diffuser is examined and evaluated.

Chapter 2

Geometry and Grid Generation

2.1 Geometry

This report is designed as the frame work of simulation of both laminar and turbulent flow over a rectangular cavity in two dimensional. Given is the rectangular cavity with hole at a distance of b . All the dimensions are in $[m]$.

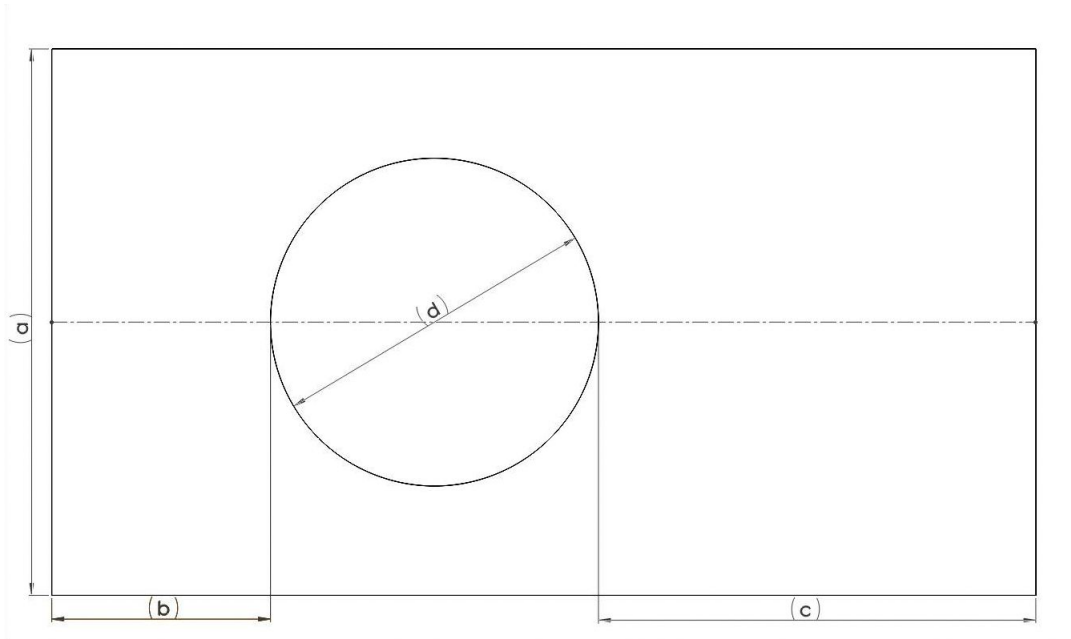


Figure 2.1: The geometrical description

Table 2.1: The values of geometrical parameters

a	b	c	d
0.5	0.2	0.3	0.4

The inlet velocity should be uniform with a magnitude of $1.5 \frac{m}{s}$ and a maximum level of fluctuations of 5% at the inlet. The fluid within the domain is water with a constant temperature of 293.15K.

For the simulation of the given cavity, "OPENFOAM" software has been used.

2.2 Generation of Coarse and Fine Mesh

The computational domain refers to a simplified form of the physical domain both in terms of geometrical representation and boundary condition imposition. Initially, the given cavity is divided into six blocks as shown in Fig.2.2 where the order of coordinates are given for all the blocks. In the end, reflection along the y axis is taken in order to get the complete domain.

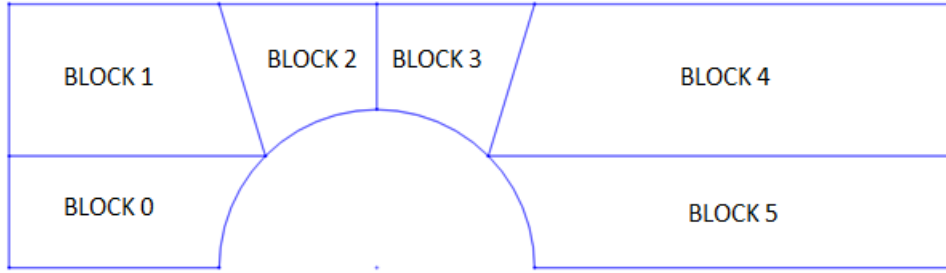


Figure 2.2: Grid Generation

The grid generation is done in a directory blockMeshDict which is located in system directory of the project file. blockMeshDict reads the geometry and generates the mesh and writes out the mesh data. The main function of blockMesh is to decompose the domain geometry into number of three dimensional, hexahedral blocks. The first block always has the label '0'. Each vertex has the coordinates labeled as (x_1, x_2, x_3) which follows the right-handed system. The right-handed system defines that when an observer looks from the z-axis, the arc from the point on x-axis to the point on y-axis gives the clockwise direction.

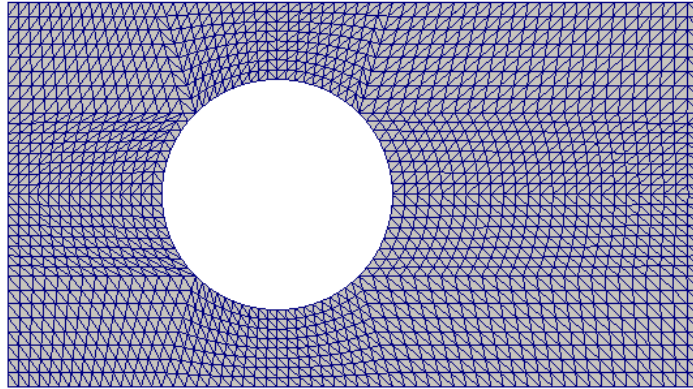


Figure 2.3: Course Mesh

In the given geometry, the domain is sketched into two parts: courseMesh and fineMesh.

- CoarseMesh contains 1600 cells
- FineMesh contains 20400 cells

Simulations for both (coarse mesh and fine mesh) is done in order to compare the results for different simulation models.

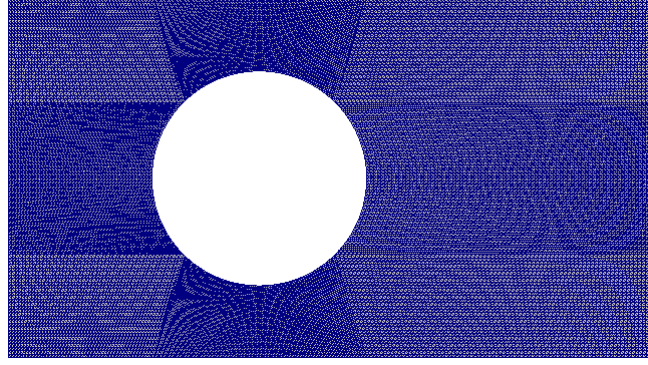


Figure 2.4: Fine Mesh

2.2.1 blockMeshdict

In this dictionary we define about the whole geometry of the diffuser. Following parameters are defined in the blockMeshdict:

- convert to meters- Scales the vertex coordinates.
- vertices- Define all the coordinates of each vertex.
- edges- Used to describe arc or spline edges.
- block- Defines about the mesh size and vertex labels.
- patches- Lists all the patches (inlet, outlet, etc).
- mergePatchPairs- Merges all the patches.

2.2.2 convertToMeters

Since all the dimensions are given in meters, we do not need to change the unit anymore. Hence, we write it as
convertToMeters 1;

2.2.3 vertices

Since, the geometry has been divided into three parts, we get sixteen vertices (according to our figure). Assuming first point as '0' having coordinates (0,0,0), all other coordinates are calculated in x,y and z-direction.

2.2.4 edges

Edges joining the two vertices are assumed to be straight line (by default). Since, in the given design all the edges are straight line, hence, we don't need to define the edges separately. Incase, we had our edges in a shape other than straight line, then we would have defined the edges as follows:

- arc- for circular arcs
- spline- for spline curves
- polyLine- for set of lines
- BSpline- for Bspline curve

2.2.5 blocks

The list blocks contains all the block definitions that we need. The block definition consists of a list of vertices according to the right-handed system. It gives about the number of cells in each direction (x,y and z), the type of cell and also the cell expansion ration for each direction. The blocks are defined as follows:

- For coarseMesh

blocks

```
(
  hex (0 1 2 13 14 15 16 27) (15 8 1) simpleGrading (1 1 1)
  hex (13 2 11 12 27 16 25 26) (15 8 1) simpleGrading (1 1 1)
  hex (2 3 10 11 16 17 24 25) (10 8 1) simpleGrading (1 1 1)
  hex (3 4 9 10 17 18 23 24) (10 8 1) simpleGrading (1 1 1)
  hex (4 7 8 9 18 21 22 23) (25 8 1) simpleGrading (1 1 1)
  hex (5 6 7 4 19 20 21 18) (25 8 1) simpleGrading (1 1 1)
);
```

- For fineMesh

blocks

```
(
  hex (0 1 2 13 14 15 16 27) (50 30 1) simpleGrading (1 1 1)
  hex (13 2 11 12 27 16 25 26) (50 30 1) simpleGrading (1 1 1)
  hex (2 3 10 11 16 17 24 25) (40 30 1) simpleGrading (1 1 1)
  hex (3 4 9 10 17 18 23 24) (40 30 1) simpleGrading (1 1 1)
  hex (4 7 8 9 18 21 22 23) (80 30 1) simpleGrading (1 1 1)
  hex (5 6 7 4 19 20 21 18) (80 30 1) simpleGrading (1 1 1)
);
```

These blocks are defined as follows:

- Vertex numbering- The first entry defines the shape of the block. Since the blocks are always hexahedral, the shape is always hex. Here, we follow the right-handed system as explained earlier.
- Number of cells- The second entry defines the number of cells in each x,y and z-direction of the cells.
- Cell expansion ratio- The third entry gives the ratio of expansion of cells in each direction. It is the ratio of the width of last cell to the width of first cell along the same direction.

A good meshing is one in which the number of cells are symmetric out the cross-section. In the above grading, it can be seen that in first block, the number of cells in x-direction are minimum as compared to that of second and third blocks. This is because, the length of first block is minimum as compared to other two blocks. Hence, for maintaining the symmetry between all three parts, the ratio between the length of each part to the number of cells shall remain constant.

Boundary

The boundary is given in the list named boundary. It is divided into patches where each patch has its different name such as:

- inlet
- outlet
- upperwall
- lowerwall
- frontAndBack

These names are used for setting up the boundary conditions in data files. It also has a sub-directory which contains the patch information. These sub-directories are:

- type: This defines the patch type on which the boundary conditions are applied.
- faces: These lists together to make up the patches.

mergePatchPairs

Each pair of patches those are to be merged are included in mergePatchPairs.

2.3 Computational Domain

In the computational domain, meshing should be given in a such way that the twelve blocks should re-link again to form a complete geometry without any topology error. While giving the meshing, special attention has to be given on those areas where change in the direction of flow and velocity take place. For the given geometry, the region ending of the zero block and starting of 1st block (ending of the 1st and starting of 2nd and so on) are the areas where there is a change in velocity so resolution has to be more on those regions.

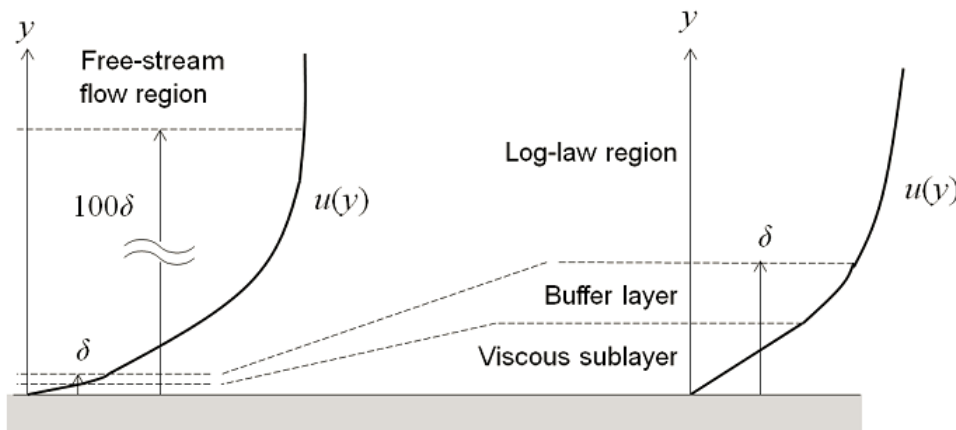


Figure 2.5: The four turbulent flow regimes[1]

Moreover, meshing near the walls should be fine as compared to any other region. The reason for this is, at the wall, the fluid velocity is zero, which is no slip condition and for

a thin layer above this, the flow velocity is linear with distance from the wall. This region is called the viscous sublayer, or laminar sublayer. Further away from the wall is a region called the buffer layer. In the buffer region, the flow begins to transition to turbulent, here during the transition to turbulent, small eddies form and it eventually transitions to a region where the flow is fully turbulent. This is known as the log-law region. Even further away from the wall, the flow transitions to the free-stream region[2]. So it is quite clear that the eddies start from the buffer layer, which causes the turbulence. As viscous and buffer layers are very thin, the eddies will also be small in size. So to solve a turbulence problem, the mesh resolution near the wall has to be much fine so that the small eddies can also be captured to solve the problem.

2.4 Mesh Resolution

The quality of the mesh plays a significant role in the accuracy and stability of the numerical computation. The grid resolution near the wall is very important as shear stress is developed during the flow of stream. To calculate the shear stresses, the $k - \omega$ SST model is used which concentrates near the walls as well as in the free flow.

2.5 Calculation of Reynolds Number

Reynolds number can be calculated from the given below formula.

$$R_e = \frac{H \cdot |u|}{\nu} \quad (2.1)$$

Here,

R_e = Reynolds number

H denotes the channel height, i.e. 0.5 m,

u denotes the magnitude of inlet velocity, i.e. $1.5 \frac{m}{s}$

ν denotes the kinematic viscosity of the fluid with a constant temperature of 293.15K, i.e. $1.25 \times 10^{-6} m^2 s^{-1}$.

Substituting the above values, we get the value of R_e as, 7.5×10^5 .

2.6 Calculation of $k - \epsilon$ Model

This model consist of two variables, first variable is kinetic energy (k) of turbulent used to find the energy of turbulent. Whereas second variable is dissipation rate (ϵ), tells the rate of dissipation of the kinetic energy of turbulent. As three dimensional case to be considered and velocity in each direction is equal so turbulent kinetic energy (k) can be calculated from the given below formula:

$$k = \frac{3}{2}(U \times I)^2 \quad (2.2)$$

here,

U is the inlet velocity i.e, $1.5ms$

I is maximum level of fluctuation of 5 % at inlet,
 By putting all the values,
 $k = 8.4375 \times 10^{-3} \frac{m^2}{s^2}$

Calculation of ϵ

Value of ϵ can be calculated from the formula,

$$\epsilon = \frac{C_\mu^{0.75} \cdot k^{1.5}}{l} \quad (2.3)$$

here,

C_μ is constant for the k- ϵ Model and its value is 0.09,
 k is the kinetic energy,

l is the turbulent length, i.e. 0.5 m,

After substituting the values,

$$\epsilon = 2.54702 \times 10^{-4} \frac{m^2}{s^3}$$

2.7 Calculation of SST k- ω Model

On one hand, k- ϵ model, successful away from the walls and on the other hand, k- ω Model, effective at the area, near to walls. This SST model is a mixture of k- ω and k- ϵ . This model holds better turbulence results as compare with k- ϵ model.

Kinetic energy (k) will remain the same and specific rate of dissipation of kinetic energy ω can be calculated from the below formula.

$$\omega = \frac{C_\mu^{-0.25} \cdot \sqrt{k}}{l} \quad (2.4)$$

here,

C_μ is constant and its value is 0.09,

k is the kinetic energy,

l is the turbulent length, i.e. 0.5m,

After substituting the values,

$$\omega = 0.335410 \text{ s}^{-1}$$

Chapter 3

The Initial Conditions and Numerical Environment

3.1 Laminar Model

The fluid particles move orderly in layers without intense lateral mixing in this model and the disruption between layers is absent[3]. The Navier Stokes equation and continuity equation are enough to calculate the velocity and pressure components for laminar flow. There is almost no possibility of turbulence flow and that's why the system turned off the same in RANS properties.

For incompressible fluid, Navier stoke equation is as follows:

$$\frac{\partial u_i}{\partial t} + \frac{\partial u_i u_j}{\partial x_j} = F_i - \frac{1}{\rho} \frac{\partial p}{\partial x_i} + \nu \nabla^2 \frac{\partial u_i}{\partial x_j} \quad (3.1)$$

The change of velocity with respect to time of a flow, determine as navier stokes equation. First term in the left hand side of the equation is local acceleration, second is acceleration due to convection. Whereas on the right hand side, acceleration due to any external force is the first term, acceleration due to pressure gradient is second term and the third term shows the acceleration due to diffusion process.

Equation of contnuity,

$$\frac{\partial u_i}{\partial t} = 0 \quad (3.2)$$

As time increases, Reynolds number increases because inertial forces become dominant forces and flow started become turbulent. It's quite clear that flow changes with the time and it can solve in time domain. For solving the small eddies which present in flow, fine mesh will be used. When the time scale of oscillations started decreasing to a very small value, it's not advisable to find the solution of problem by using navier stokes equation computationally. To overcome this problem, we use Reynolds Averaging Navier Stokes (RANS) equation.

3.2 Turbulent Model

The turbulent flow is very chaotic with strong eddies and intense mixing across the flow. Turbulent motion is the three dimensional unsteady flow motion with

- chaotical trajectories of fluid particles,
- fluctuations of the velocity and
- strong mixing

arisen at large Re numbers due to unstable vortex dynamics[3].

Two different models (SST $k - \omega$ and $k - \epsilon$) are used under turbulent flow. As it is not advisable to find the solution with the help of navier stokes equation. Hence, we use Reynolds Averaging Navier Stokes (RANS) equation and SST $k - \omega$ and $k - \epsilon$ models to be used.

3.2.1 k - ϵ model

The K-epsilon model is one of the most common turbulence models but not effective in large adverse pressure gradients. It is a two equation model, first transported variable is turbulent kinetic energy, k and the second transported variable is the turbulent dissipation, ϵ . This k - ϵ model is successful away from the walls[4].

3.2.2 SST k - ω model

The SST k- turbulence model is a two-equation eddy-viscosity model. The shear stress transport (SST) formulation is used with $k - \omega$. The use of a k- formulation in the inner parts of the boundary layer makes the model directly usable to the wall through the viscous sub-layer, hence the SST k- model can be used as a Low-Re turbulence model without any extra damping functions[5].

3.3 Partial Differential Equations

3.3.1 k- ϵ Model

Two partial differential equations have been used to find the kinetic energy (k) and dissipation rate of kinetic energy (ϵ)[4].

For turbulent kinetic energy (k),

$$\frac{\partial(\rho k)}{\partial t} + \frac{\partial(\rho k u_i)}{\partial x_i} = \frac{\partial[(\mu + \frac{\mu_t}{\sigma_k}) \frac{\partial k}{\partial x_j}]}{\partial x_j} + P_k + P_b - \rho \epsilon - Y_M + S_k \quad (3.3)$$

For dissipation ϵ ,

$$\frac{\partial(\rho \epsilon)}{\partial t} + \frac{\partial(\rho \epsilon u_i)}{\partial x_i} = \frac{\partial[(\mu + \frac{\mu_t}{\sigma_\epsilon}) \frac{\partial \epsilon}{\partial x_j}]}{\partial x_j} + C_{1\epsilon} \frac{\epsilon}{k} (P_k + C_{3\epsilon} P_b) - C_{2\epsilon} \rho \frac{\epsilon^2}{k} + S_\epsilon \quad (3.4)$$

Turbulent viscosity is modelled as,

$$\mu_t = \rho C_\mu \frac{k^2}{\epsilon}$$

Production of k ,

$$P_k = -\rho \overline{u'_i u'_j} \frac{\partial u_j}{\partial x_i}$$

$$P_k = \mu_t S^2$$

Where S is the modulus of the mean rate-of-strain tensor, defined as,

$$S \equiv \sqrt{2 S_{ij} S_{ij}}$$

Effect of buoyancy,

$$P_b = \beta g_i \frac{\mu_t}{Pr_t} \frac{\partial T}{\partial x_i}$$

where Pr_t is the turbulent Prandtl number for energy and g_i is the component of the gravitational vector in the i^{th} direction. For the standard and realizable - models, the default value of Pr_t is 0.85. The coefficient of thermal expansion, β , is defined as,

$$\beta = -\frac{1}{\rho} \left(\frac{\partial \rho}{\partial T} \right)_p$$

Model constants,

$$C_{1\epsilon} = 1.44, \quad C_{2\epsilon} = 1.92, \quad C_\mu = 0.09, \quad \sigma_k = 1.0, \quad \sigma_\epsilon = 1.3$$

3.3.2 SST k - ω Model

The partial differential equations associated with the k - ω models are [5].

Turbulence Kinetic Energy,

$$\frac{\partial k}{\partial t} + U_j \frac{\partial k}{\partial x_j} = P_k - \beta^* k \omega + \frac{\partial}{\partial x_j} \left[(\nu + \sigma_k \nu_T) \frac{\partial k}{\partial x_j} \right] \quad (3.5)$$

Specific Dissipation Rate,

$$\frac{\partial \omega}{\partial t} + U_j \frac{\partial \omega}{\partial x_j} = \alpha S^2 - \beta \omega^2 + \frac{\partial}{\partial x_j} \left[(\nu + \sigma_\omega \nu_t) \frac{\partial \omega}{\partial x_j} \right] + 2(1 - F_1) \sigma_{\omega 2} \frac{1}{\omega} \frac{\partial k}{\partial x_i} \frac{\partial \omega}{\partial x_i} \quad (3.6)$$

Kinematic Eddy Viscosity,

$$\nu_T = \frac{a_1 k}{\max(a_1 \omega, SF_2)}$$

Closure Coefficients and auxiliary relations,

$$F_2 = \tanh \left[\left[\max \left(\frac{2\sqrt{k}}{\beta^* \omega y}, \frac{500\nu}{y^2 \omega} \right) \right]^2 \right]$$

$$P_k = \min \left(\tau_{ij} \frac{\partial U_i}{\partial x_j}, 10\beta^* k\omega \right)$$

$$F_1 = \tanh \left\{ \left\{ \min \left[\max \left(\frac{\sqrt{k}}{\beta^* \omega y}, \frac{500\nu}{y^2 \omega} \right), \frac{4\sigma_{\omega 2} k}{CD_{k\omega} y^2} \right] \right\}^4 \right\}$$

$$CD_{k\omega} = \max \left(2\rho\sigma_{\omega 2} \frac{1}{\omega} \frac{\partial k}{\partial x_i} \frac{\partial \omega}{\partial x_i}, 10^{-10} \right)$$

$$\phi = \phi_1 F_1 + \phi_2 (1 - F_1)$$

$$\alpha_1 = \frac{5}{9}, \quad \alpha_2 = 0.44$$

$$\beta_1 = \frac{3}{40}, \quad \beta_2 = 0.0828$$

$$\beta^* = \frac{9}{100}$$

$$\sigma_{k1} = 0.85, \quad \sigma_{k2} = 1$$

$$\sigma_{\omega 1} = 0.5, \quad \sigma_{\omega 2} = 0.856$$

3.4 Momentum Equation

The Navier-Stokes equation is the basic governing equation for a heat conducting and viscous fluid. It is a vector equation derived by applying Newton's second law of motion to a fluid element and is also known as the momentum equation. It is expanded by the mass conservation equation, also known as continuity equation and the energy equation. Body and surface forces are two types of inner forces which acting on a fluid. The body forces are acting at each point of fluid in the whole domain. Gravitational, electrostatic and electromagnetic (specialized situation) are common body forces. The surface forces are acting at each point at the boundary of the fluid element. Usually they are shear and normal stresses and comes in to picture due to pressure or viscous stresses.

Navier Stokes equation in tensor form:

$$\rho \frac{\partial u_i}{\partial t} + \rho u_i \frac{\partial u_j}{\partial x_i} = -\frac{\partial P}{\partial x_j} - \frac{\partial \tau_{ij}}{\partial x_i} + \rho g_j$$

where,

$$\tau_{ij} = -\mu \left(\frac{\partial u_j}{\partial x_i} + \frac{\partial u_i}{\partial x_j} \right) + \frac{2}{3} \delta_{ij} \mu \frac{\partial u_k}{\partial x_k}$$

This equation is explained as:

$\rho \frac{\partial u_i}{\partial t}$ is local change with time

$\rho u_i \frac{\partial u_j}{\partial x_i}$ is Momentum convection

$\frac{\partial P}{\partial x_j}$ is surface force

$\frac{\partial \tau_{ij}}{\partial x_i}$ is Molecular- dependent momentum change (diffusion)

ρg_j is Mass force

3.5 Initial Conditions

The initial conditions for the simulation will be given to 0 folder, for each modeling case. The initial conditions regarding the velocity ($1.5 \frac{m}{s}$), turbulent kinetic energy (k), rate of dissipation of kinetic energy (ϵ), specific rate of dissipation of kinetic energy (ω), viscosity ($\nu = 1 \times 10^{-6}$) and pressure components will be the same in both the meshing of modeling, i.e laminar modeling and turbulence modeling. And values of all be given in the below table:

Table 3.1: Boundary conditions for the given geometry

Parameter	Boundary face	Type	Value	Estimation
P	inlet	zero gradient	-	-
	outlet	fixed value	0	-
	lower wall	zero gradient	-	-
	upper wall	zero gradient	-	-
	Cylinder	Zero gradient	-	-
	frontAndBack	empty	-	-
U	Inlet	fixed value	1.5	-
	Outlet	zero gradient	-	-
	Lower wall	fixed Value	0	-
	Upper wall	fixed Value	0	-
	Cylinder	fixed value	0	-
	frontAndBack	empty	-	-
K	Inlet	fixed Value	8.4375×10^{-3}	$k = \frac{3}{2}(U \times I)^2$ with 5% fluctuation at inlet
	Outlet	zero Gradient	-	
	Lower wall	kqRWall Function	8.4375×10^{-3}	
	Upper wall	kqRWall Function	8.4375×10^{-3}	
	Cylinder	kqRWall Function	8.4375×10^{-3}	
	frontAndBack	empty	-	
ω	Inlet	fixed Value	0.335410	$\omega = \frac{C_\mu^{-0.25} \cdot \sqrt{k}}{l}$ $k = 8.4375 \times 10^{-3}$ $l = 0.5$ $C_\mu = 0.09$
	Outlet	zero Gradient	-	
	Lower wall	omegaWall Function	0.335410	
	Upper wall	omegaWall Function	0.335410	
	Cylinder	omegaWall Function	0.335410	
	frontAndBack	empty	-	
ϵ	Inlet	fixed Value	2.54702×10^{-4}	$\epsilon = \frac{C_\mu^{0.75} \cdot k^{1.5}}{l}$ $k = 8.4375 \times 10^{-3}$ $l = 0.5$ $C_\mu = 0.09$
	Outlet	zero Gradient	-	
	Lower wall	epsilonWall Function	2.54702×10^{-4}	
	Upper wall	epsilonWall Function	2.54702×10^{-4}	
	Cylinder	epsilonWall Function	2.54702×10^{-4}	
	frontAndBack	empty	-	

Chapter 4

Reynolds Averaged Navier Stokes (RANS) equations

It is neither feasible nor desirable to consider in detail all of the small-scale fluctuations that occur in the atmosphere. For this reason, we introduce averaging or smoothing operators, and attempt to describe only the average state of the atmosphere[6]. The idea behind the equations is Reynolds decomposition, whereby an instantaneous quantity is decomposed into its time-averaged and fluctuating quantities[7].

Instantaneous quantity = Time-averaged Quantity + Fluctuating quantities

$$u_x = \bar{u}_x + u'_x; \quad u_y = \bar{u}_y + u'_y; \quad u_z = \bar{u}_z + u'_z \quad (4.1)$$

Reynolds averaging is often used in fluid dynamics to separate turbulent fluctuations from the mean-flow[8].

The Reynolds decomposition phenomenon is used to solve the Navier Stokes equation. The RANS equation is given below,

$$\rho \bar{u}_j \frac{\partial \bar{u}_i}{\partial x_j} = \rho \bar{f}_i + \frac{\partial [-\bar{p} \delta_{ij} + 2\mu \bar{S}_{ij} - \overline{\rho u'_i u'_j}]}{\partial x_j} \quad (4.2)$$

The left hand side of this equation represents the change in mean momentum of fluid element owing to the unsteadiness in the mean flow and the convection by the mean flow[7].

Where,

$$\bar{S}_{ij} = \frac{1}{2} \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) \quad (4.3)$$

is the mean rate of strain tensor.

Upon averaging the fluid equations, a stress on the right hand side appears of the form $\overline{\rho u'_i u'_j}$. This is the Reynolds stress, conventionally written R_{ij} .

$$R_{ij} \equiv \overline{\rho u'_i u'_j} \quad (4.4)$$

For RANS, the Reynolds stresses are approximated by the Boussinesq approach.

$$-\overline{u'_i u'_j} = \nu_t \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} - \frac{2}{3} \frac{\partial \bar{u}_k}{\partial x_k} \delta_{ij} \right) - \frac{2}{3} K \delta_{ij}$$

Which can be written in shorthand as,

$$-\overline{u'_i u'_j} = 2\nu_t \bar{S}_{ij} - \frac{2}{3} K \delta_{ij}$$

In this approach, the turbulent eddy viscosity (ν_t) is the unknown term which has to be modeled. This is done by additional transport equations which will be explained in the following chapter.

There are three models for determining the reynolds stress:

- Linear eddy viscosity model
- Non-linear eddy viscosity model
- Reynolds stress models

4.1 Linear eddy viscosity model

In this model, the Reynolds stresses obtained from averaging the Navier-Stokes equation are modeled by a linear constitutive relationship with the mean flow straining field.

For computing the eddy viscosity coefficient, there are several subcategories for this model depending on the number of transport equations:

- Algebraic models
- One equation models
- Two-equation models

Algebraic models

This model do not require solution of any other additional equations for calculation of turbulence properties. It calculates directly from the flow variables. Hence, it is sometimes possible that it do not give proper solutions for convection and diffusion of turbulent energy. It is also known as zero-equation turbulence model. This type of model is very useful for simple flow geometry or in start-up situations. Several zero equation models are:

- Cebeci-Smith model
- Baldwin-Lomax model
- Johnson-king model
- A roughness dependent model

One equation models

This model solves one turbulent transport equation, generally turbulence kinetic energy. Common types of one equation models are:

- Prandtl's one-equation model
- Baldwin-Barth model
- Spalart-Allmaras model
- Rahman-Agarwal-Siikonen model

Two equation turbulence models

This is the most widely used turbulence models. Two equation models include two extra transport equation for representing the turbulent properties. There are two kinds of transport variables. Turbulent kinetic energy, k is the most often used transport variables. The second type of transport variable used depends upon the type of two-equation model. The second commonly used type of transport variable is ϵ and ω . Commonly used two-equation models are:

- k - ϵ model
- k - ω model

k - ϵ models

In k - ϵ models, the first transport variable is turbulent kinetic energy, k which determines the energy of turbulence in a system and the second transport variable is turbulent dissipation, ϵ which determines the scale of turbulence. Commonly used k - ϵ models are:

- Standard k - ϵ model
- Realisable k - ϵ model
- RNG k - ϵ model

k - ω models

In k - ω models, kinetic energy (k) is the first transport variable which determines the energy of turbulence and ω is the second transport variable which determines the specific dissipation.

- Wilcox's k - ω model
- Wilcox's modified k - ω model
- SST k - ω model

The most commonly used k - ω model is SST k -omega model. The SST (shear stress transport) k - ω model is used as low Reynolds turbulence model without any extra damping function which is an advantage over using the k - ω model in which the inner parts in a boundary layer makes the model directly usable all the way down to the wall through the viscous sub-layer. k - ω models can be of following types:

4.2 Non-linear eddy viscosity model

It is a non-linear relationship in which an eddy viscosity coefficient is used to for generating the relation between mean turbulence field and mean velocity field.

4.3 Reynolds stress models

In this model, the Linear and non-linear eddy viscosity approach is discarded and Reynolds stresses are computed directly. The Reynolds stress transport equation accounts for the directional effects of the Reynolds stress fields. The differential transport equations are used to calculate the individual Reynolds stresses. These Reynolds stresses are used to obtain the Reynolds average momentum equation.

Chapter 5

Grid resolution near the wall and yPlus estimation

5.1 Grid Resolution

The quality of the meshing plays an important role in the accuracy and balancing of the numerical computation. The grid resolution near the walls is very important, as the shear stress is developed during the flow of stream. It should be very fine near the walls as compared to any other region. The reason for this is,

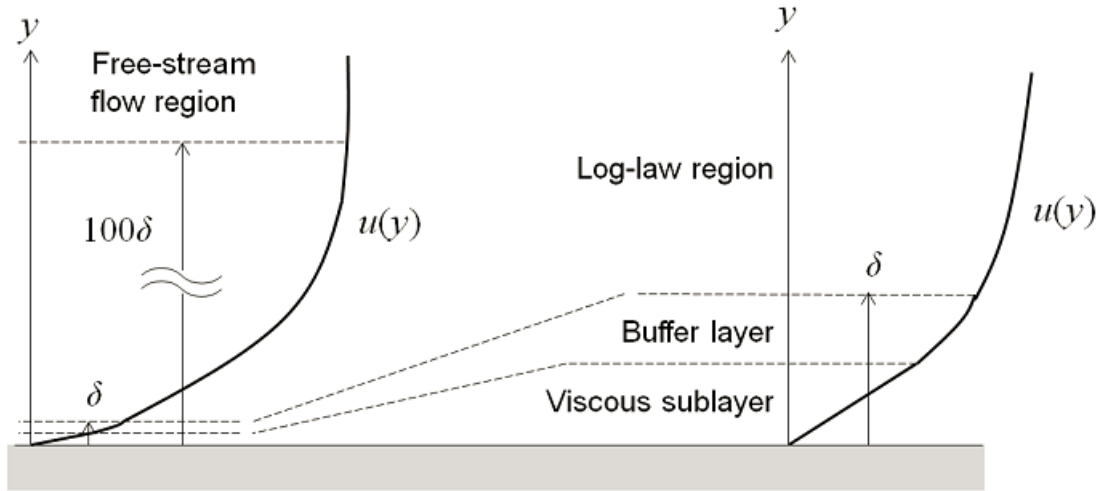


Figure 5.1: The four turbulent flow regimes[1]

at the wall, the fluid velocity is zero, which is no slip condition and for a thin layer above this, the flow velocity is linear with distance from the wall. This region is called the viscous sublayer, or laminar sublayer. Further away from the wall is a region called the buffer layer. In the buffer region, the flow begins to transition to turbulent, here during the transition to turbulent, small eddies form and it eventually transitions to a region where the flow is fully turbulent. This is known as the log-law region. Even further away from the wall, the flow transitions to the free-stream region[2]. So it is quite clear that the eddies start from the buffer layer, which causes the turbulence. As viscous and

buffer layers are very thin, the eddies will also be small in size. So to solve a turbulence problem, the grid resolution near the wall has to be much fine so that the small eddies can also be captured to solve the problem.

5.2 yPlus Estimation

For a wall bounded flow, a non dimensional wall distance can be defined as,

$$y^+ = \frac{\mu_\tau y}{\nu}$$

here,

u_τ is the friction velocity or shear velocity at the nearest wall,

y is the distance to the nearest wall,

ν is the local kinematic viscosity of the fluid.

y^+ is often referred to simply as y plus and is commonly used in boundary layer theory and in defining the law of the wall[9].

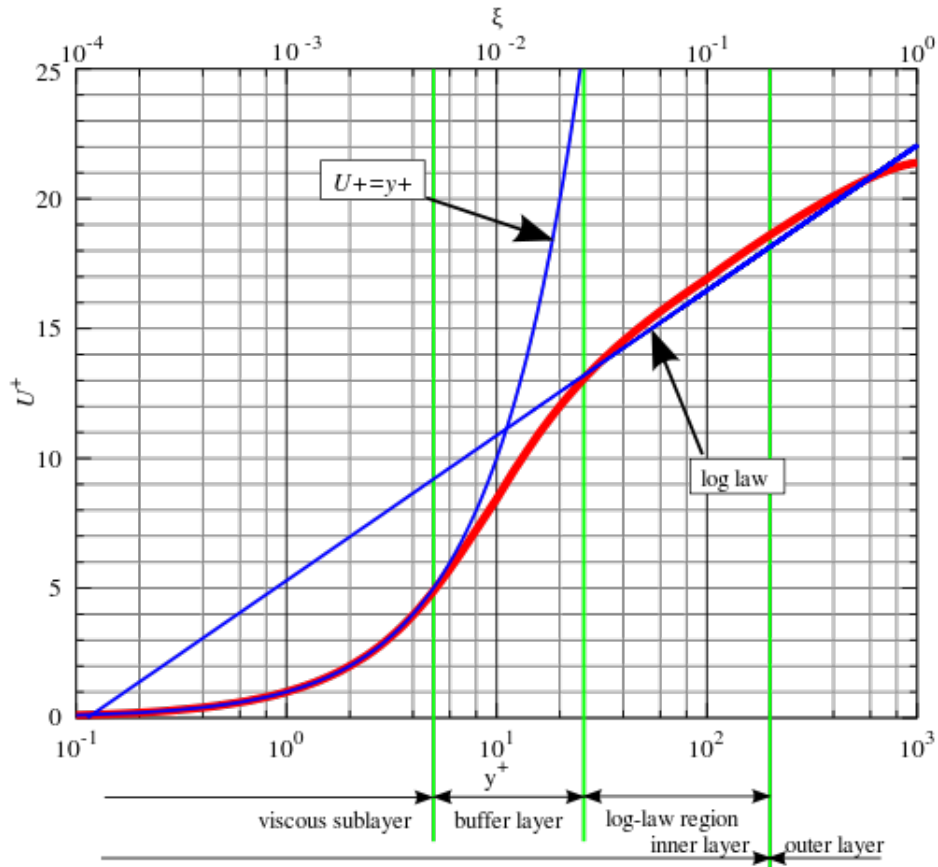


Figure 5.2: law of wall function[2]

Near the Walls

Below the region where the law of the wall is applicable, there are other estimations for friction velocity.

The stress in wall turbulence flow can be considered as the sum of the laminar and turbulent stresses[3],

$$\tau = \tau_l + \tau_t \quad (5.1)$$

Close to the wall, the turbulent fluctuations are weak. The laminar stress τ_l dominates over the turbulent one τ_t , i.e. $\tau \sim \tau_l$. We consider the thin boundary layer, i.e. the stress is approximately equal to the wall stress τ_w :

$$\tau \approx \tau_w \quad (5.2)$$

Applying the Newton hypothesis to the two dimensional wall bounded flow,

$$\tau_w = \rho \nu \frac{du_x}{dy} \quad (5.3)$$

or,

$$\frac{u_x}{u_\tau} = y^+ + C \quad (5.4)$$

From the condition at the wall $u_x = 0$, the unknown constant C is zero, i.e.

$$\frac{u_x}{u_\tau} = y^+ \quad (5.5)$$

Viscous Sublayer

Below the 5 walls unit, variation of u^+ to y^+ will be approximately 1:1 in this region for,

$$\begin{aligned} y^+ &< 5 \\ u^+ &= y^+ \end{aligned}$$

here,

y^+ is the wall coordinate

$u^+ = \frac{u}{u_\tau}$ is the dimensionless velocity.

$$u^+ = \frac{1}{k} \log y^+ + C^+, \quad \mu_\tau = \sqrt{\frac{\tau_w}{\rho}}$$

here,

τ_w is the wall shear stress,

ρ is the fluid density,

u_τ is called the friction velocity or shear velocity,

K is the Von kármán constant and $k \approx 0.41$,

C^+ is a constant and has value 5.0

Buffer Layer

In this region, no law holds from 5 walls to 30 walls unit.
for,

$$\begin{aligned} 5 < y^+ < 30 \\ u^+ \neq y^+ \end{aligned}$$

Chapter 6

Results and Discussion

6.1 $k - \epsilon$ Model with Coarse Meshing

6.1.1 Velocity Distribution

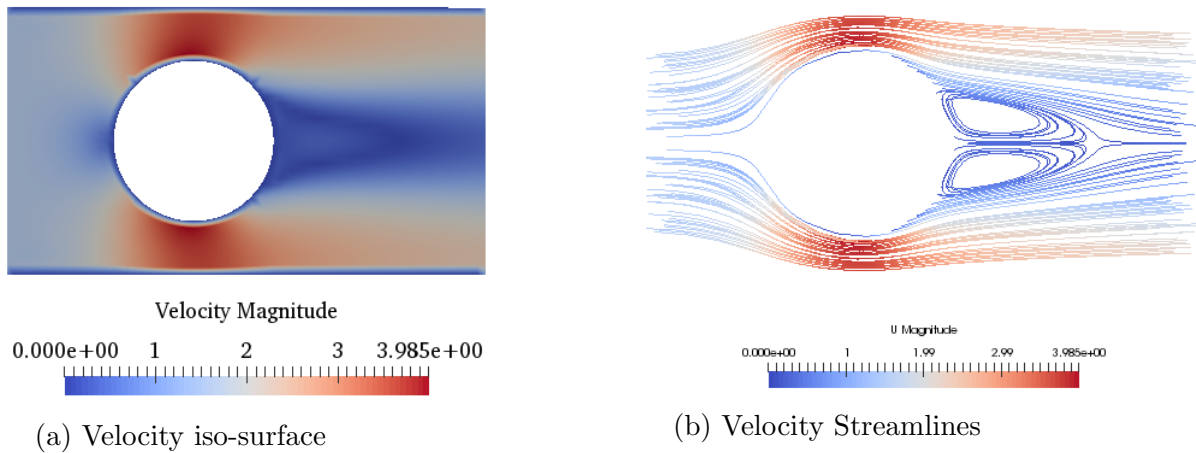


Figure 6.1: Velocity in $k - \epsilon$ with coarse mesh

6.1.2 Pressure Distribution

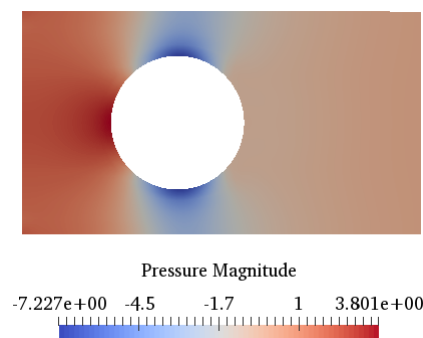


Figure 6.2: Pressure iso-surface in $k - \epsilon$ with coarse mesh

In this model, the velocity gradients are not getting resolved near the walls. Turbulence can be seen around the cylinder due to higher reynolds number. As we know when fluid touches the cylinder, velocity becomes zero and then gradually increases. Same phenomenon can be observed in this model.

For pressure distribution, at stagnation point, velocity is zero and magnitude of pressure is maximum which can be seen in fig. 6.2

6.2 $k - \epsilon$ Model with Fine Meshing

6.2.1 Velocity Distribution

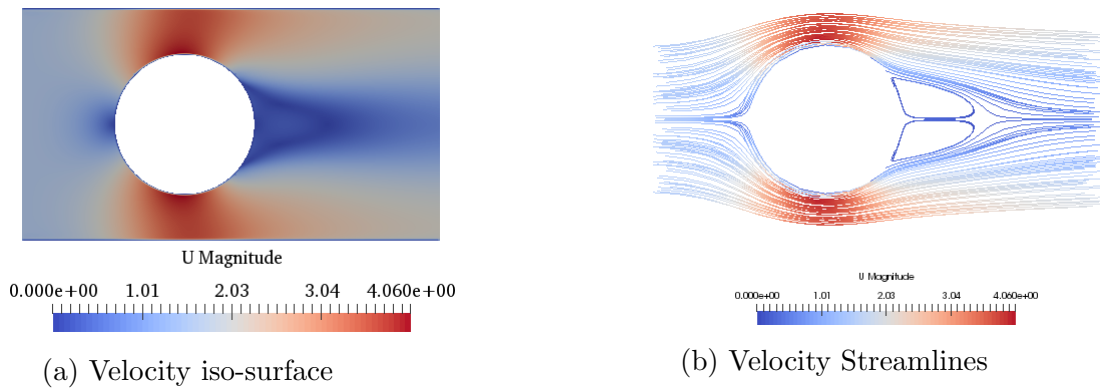


Figure 6.3: Velocity in $k - \epsilon$ with fine mesh

6.2.2 Pressure Distribution

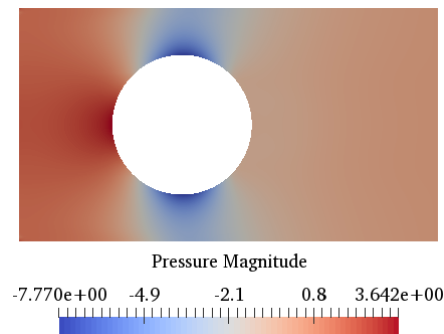


Figure 6.4: Pressure iso-surface in $k - \epsilon$ with fine mesh

This model is better as compared to $k - \epsilon$ with coarse mesh as high velocity gradients are resolved near the walls. Reason for this is, due to fine meshing, small eddies can be captured and more physical results can be observed as compared with coarse mesh.

For pressure distribution, maximum value of pressure can be observed at inlet and outlet. Pressure above and below the cylinder is minimum as at this region, velocity starts

increases after the stagnation point.

6.3 Comparison of K - ϵ with coarse and fine

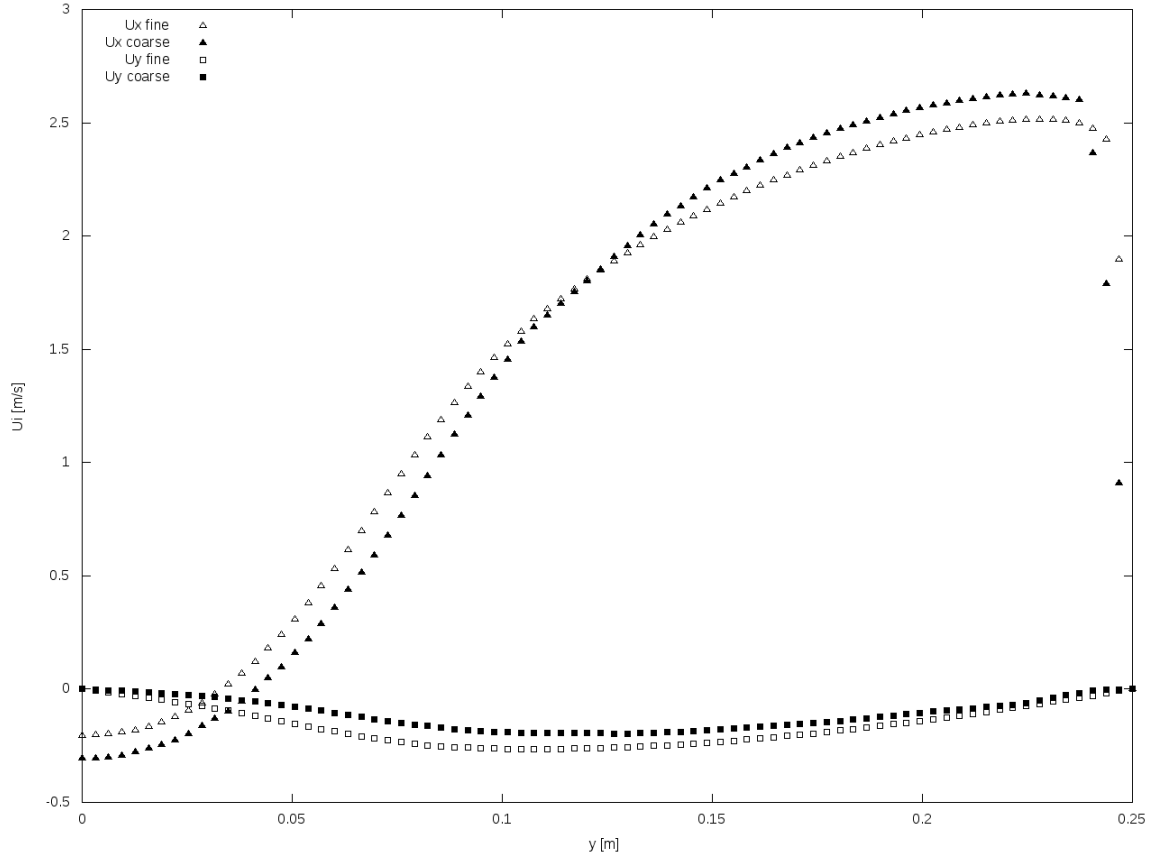


Figure 6.5: Mesh Resolution in k - ϵ Model

This graph gives the practical approach towards the explanation of coarse and fine mesh in k - ϵ model. This graph explains the comparison of U_x and U_y in coarse and fine mesh with respect to wall distance. System is getting better physical results near the walls in fine mesh as compared to coarse mesh.

6.4 SST k - ω model with Coarse Meshing

6.4.1 Velocity Distribution

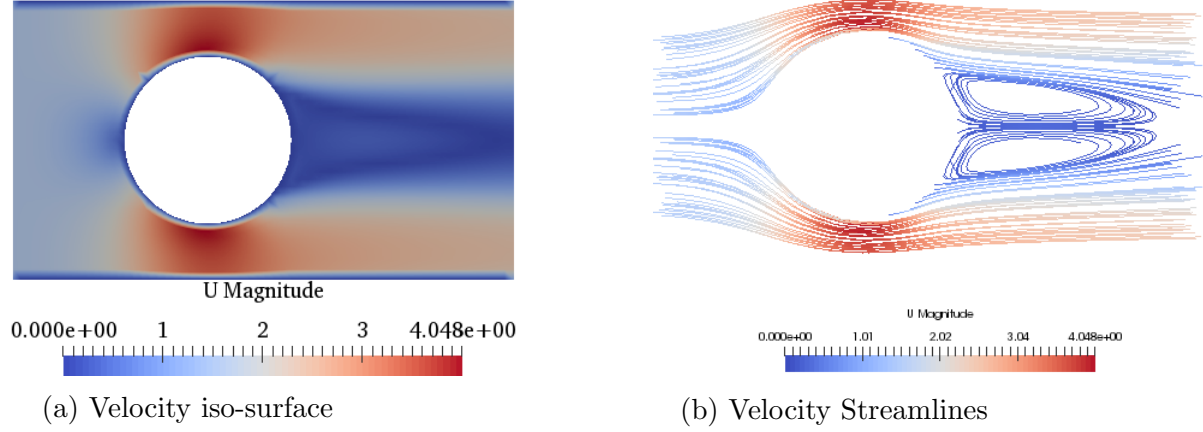


Figure 6.6: Velocity in SST k - ω with coarse mesh

6.4.2 Pressure Distribution

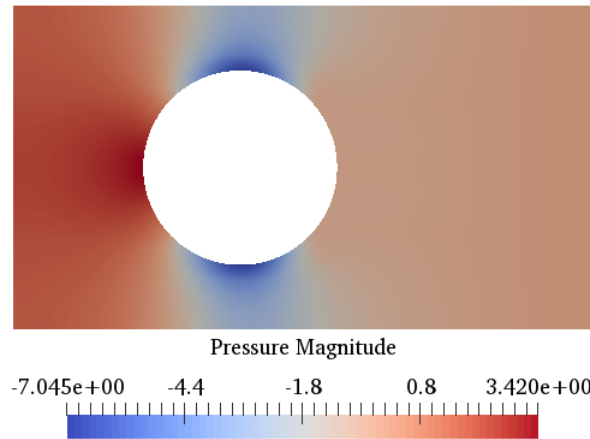


Figure 6.7: Pressure iso-surface in SST k - ω with coarse mesh

In this model, the velocity gradients are not getting resolved near the walls. Although k - ω SST gives us better results near the walls but due to low intensity of cells near the walls, it's not able to give more physical results. Turbulence can be seen around the cylinder due to higher reynolds number. In streamlines view, re circulation of fluids can be seen where the velocity is minimum. Moreover at stagnation point, velocity becomes zero and then starts increasing gradually. Same phenomenon can be observed in this model.

For pressure distribution, The minimum pressure can be found above and below the cylinder.

6.5 SST k - ω with Fine Meshing

6.5.1 Velocity Distribution

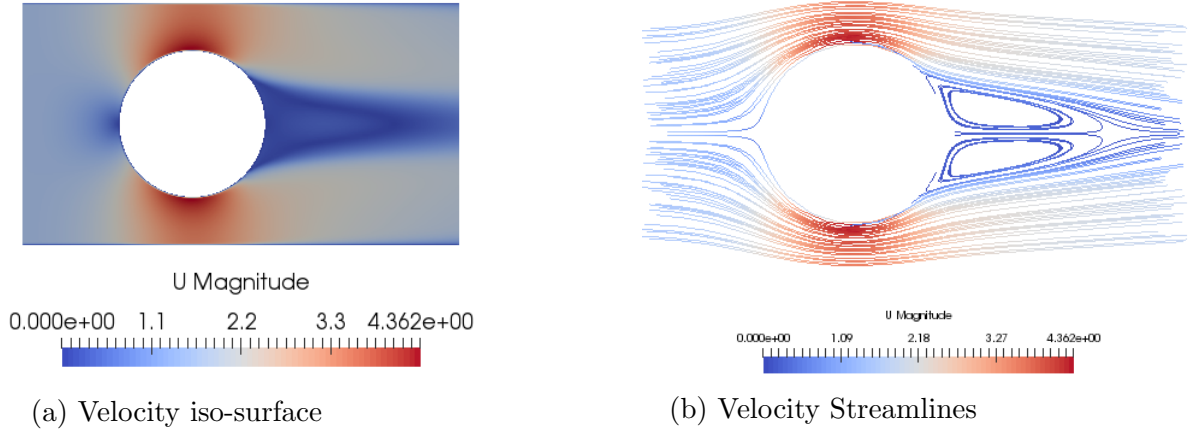


Figure 6.8: Velocity in SST k - ω with fine mesh

6.5.2 Pressure Distribution

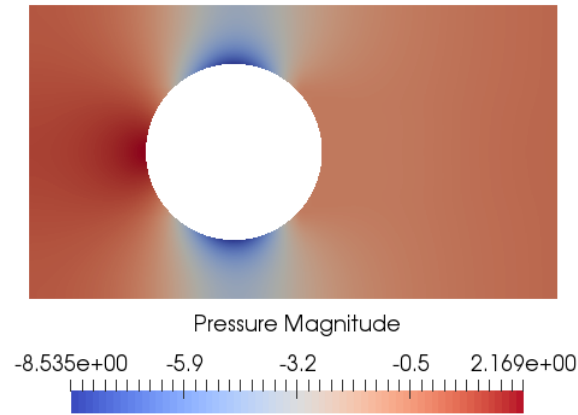


Figure 6.9: Pressure iso-surface in SST k - ω with fine mesh

Best physical results can be observed in this model because this is the hybrid model of k- ϵ and k - ω turbulence model and it gives better results near the walls as well as away from the walls and same can be easily seen in above figure. Due to high intensity of cells near the walls, as required to capture small eddies for stable and better physical results. High velocity gradients can also be resolved in this model. Re circulation of fluids are seen on those areas where velocity is minimum.

For pressure distribution, The minimum pressure can be found above and below the cylinder.

6.6 Comparison of K - ω with coarse and fine

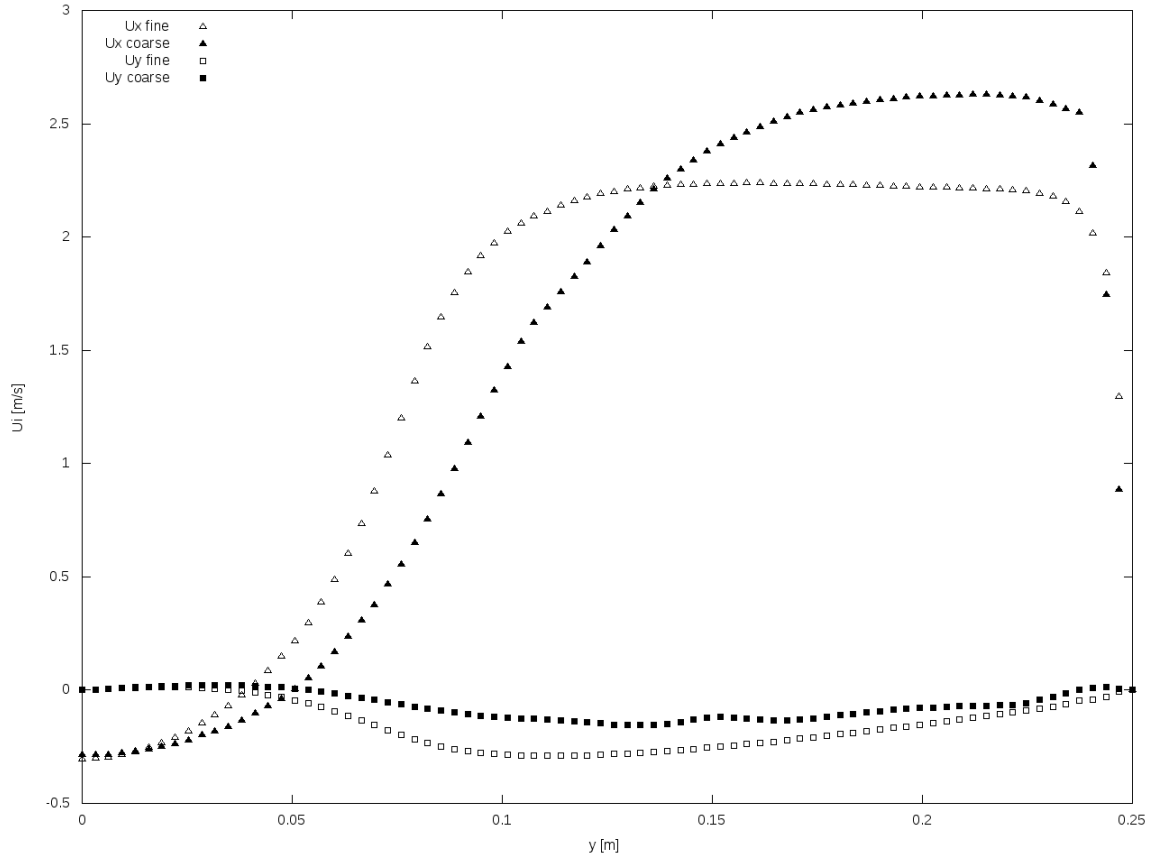


Figure 6.10: Mesh Resolution in SST k - ω Model

This graph gives more practical approach towards the explanation of coarse and fine mesh in k - ω model. The graph explains the comparison of U_x and U_y in coarse and fine mesh with respect to wall distance. The difference in magnitude of velocity in coarse and fine mesh is observed and it can be concluded that fine mesh gives more stable results than other models. Because of fine meshing and its capability to solve more precisely near the walls, velocity profile is smooth and more stable. System is getting better physical results near the walls in fine mesh as compared to coarse mesh for this model.

6.7 Laminar Model with Coarse Meshing

6.7.1 Velocity Distribution

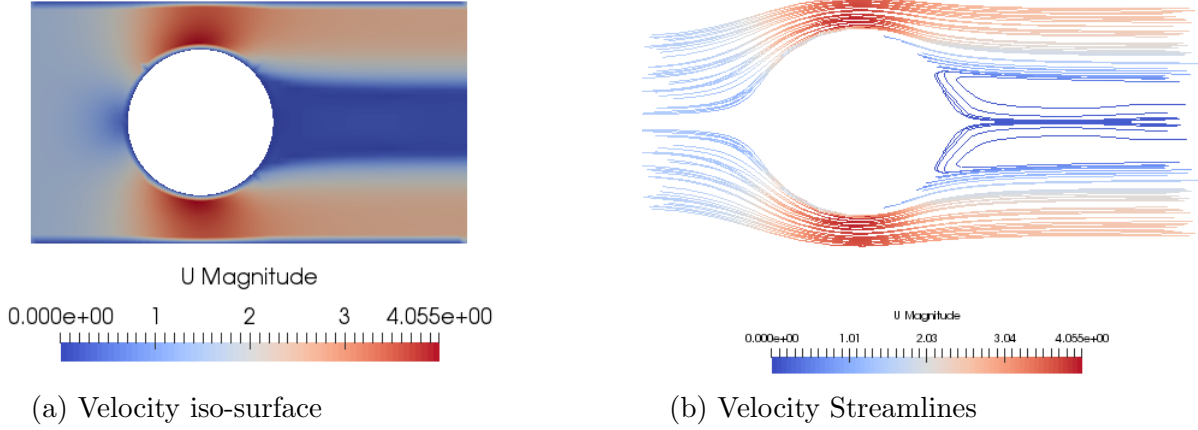


Figure 6.11: Velocity in laminar with coarse mesh

6.7.2 Pressure Distribution

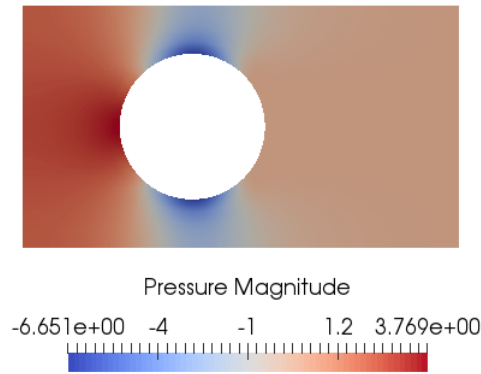


Figure 6.12: Pressure iso-surface in laminar with coarse mesh

Since, we have a turbulent flow, it is obvious that laminar model will not be able to give good results for turbulent flow. The same behaviour is seen on simulating with this model. The model will have turbulence through the cross section and unstable results are obtained. If we look at the streamline view of this model, there is a section where we observe very irregular flow, which is practically not feasible for any system.

6.8 Laminar Model with Fine Meshing

6.8.1 Velocity Distribution

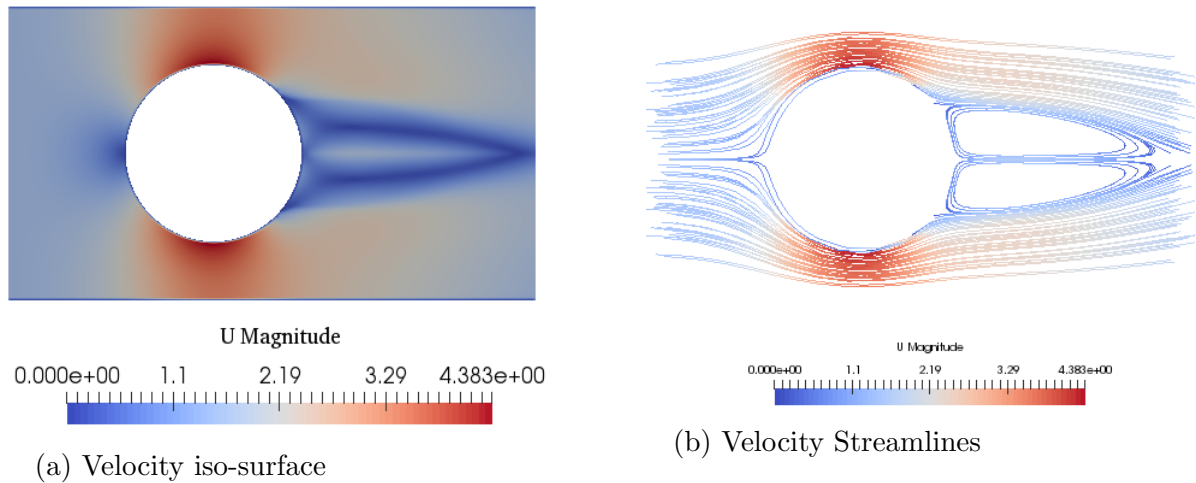


Figure 6.13: Velocity in laminar with fine mesh

6.8.2 Pressure Distribution

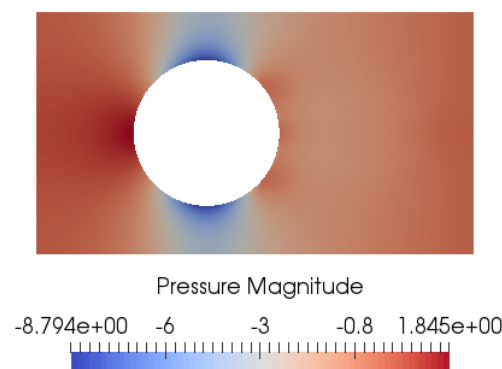


Figure 6.14: Pressure iso-surface in laminar with fine mesh

By having the fine mesh, it is expected to have improved results as compared to coarse mesh. But irrespective of having fine meshing, this model is not getting stable results and irregularities continue to be there through the cross section. Streamline view shows these irregularities very clearly.

6.9 Comparison of Laminar with coarse and fine

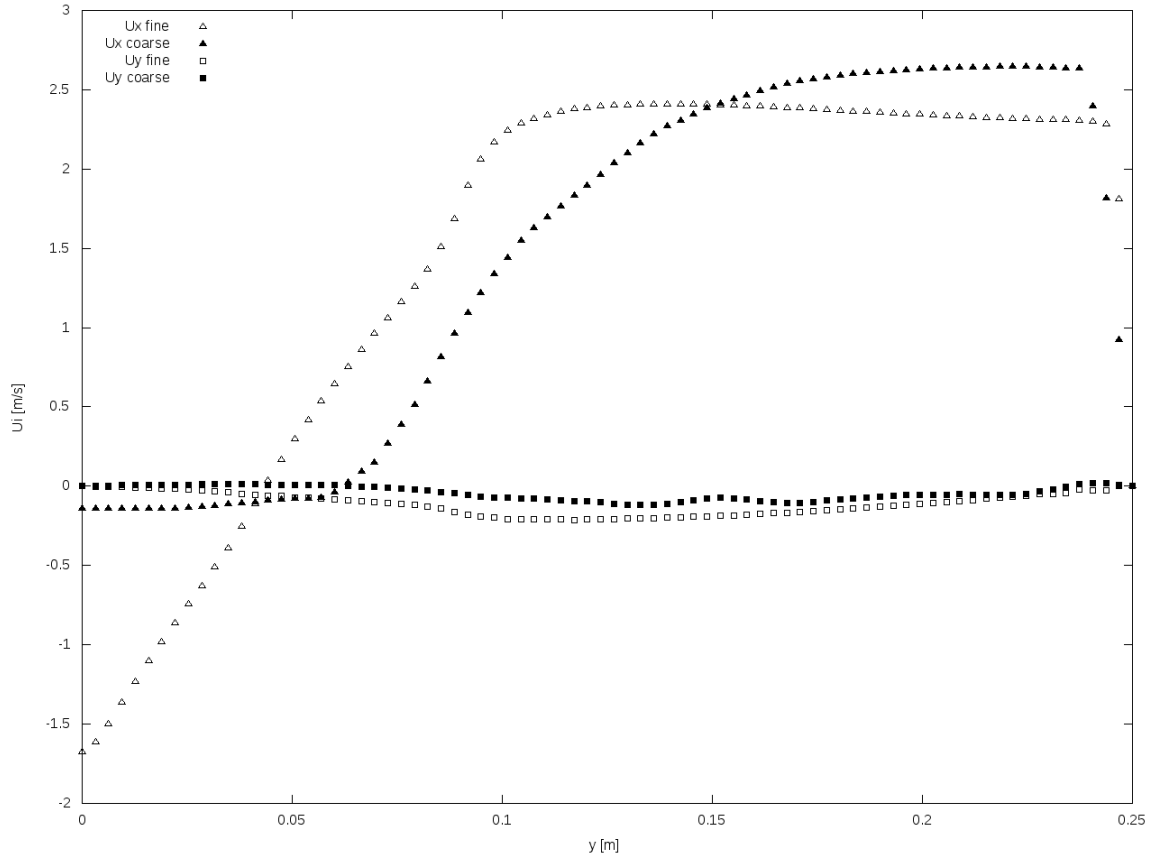


Figure 6.15: Mesh Resolution in Laminar Model

There is no proper behaviour followed by the graphs in this model. Although by use of fine mesh, it tries to get better results but the system is not able to achieve the stability. Velocity profile near the wall is also poor. Due to so high Reynolds number, it is not advised to use laminar model for this type of flow.

Chapter 7

Comparison of Different Models

7.1 Difference between Coarse and Fine Grid

- The quality of the mesh plays a significant role in the accuracy and stability of the numerical computation. The grid resolution near the wall is very important as shear stress is developed during the flow of stream. To resolve high velocity gradients near the walls, system needs fine meshing. For instance, in fig 6.1, 6.6, 6.11 (coarse mesh), none of the model is able to resolve high velocity gradients near the walls. On the other hand, In fine mesh, fig. 6.3, 6.8. 6.13, high velocity gradients can be resolved.
- Secondly, eddies are very small in size and to solve a turbulence problem, the mesh resolution near the wall has to be much fine, so that the small eddies can also be captured and more practical results can be achieved. It's also clear from the above figures that, more physical results can be observed with fine mesh instead of coarse mesh.
- it's preferable to use a fine mesh for an independent solution, which tells that the solution should not change if the mesh is further refined[10].

7.2 Difference b/w the Results of Different Models

- In figure 7.1, comparison of different models for the given turbulent flow of fluid is analysed and it's observed that laminar model is enable to give the results which are practically acceptable. On the other hand, $k - \omega$ & $k - \epsilon$ gives almost similar, more stable and practically acceptable results.
- In coarse mesh, the velocity pattern U_x is almost similar in $K - \epsilon$ and SST $K - \omega$ model. Velocity is maximum and nearly same at the inlet and the outlet. Whereas, in laminar model, different profile is observed. For velocity profile U_y the initial velocity is almost same in all the three cases.
- Results in SST $k - \omega$ and $k - \epsilon$ have stronger re circulation than in laminar model.
- Near the walls, Laminar and SST $k - \omega$ almost have the same plot. On moving further away from the wall SST $k - \omega$ starts behaving similar to $k - \epsilon$ model. When Distance

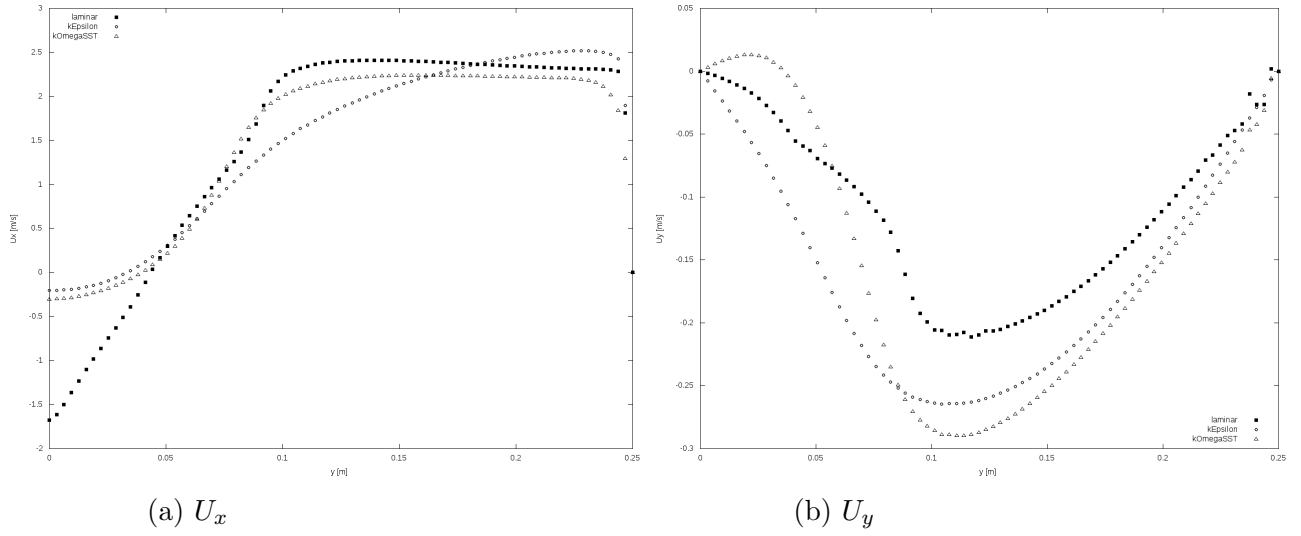


Figure 7.1: Comparison of Different Models

from the wall is increasing, SST $k - \omega$ switches from $k - \omega$ to $k - \epsilon$. SST $k - \omega$ model is hybrid model of $k - \omega$ and $k - \epsilon$.

- Best flow separations have been captured in SST $k - \omega$ (fine mesh - fig 6.8) model as compared to any other model. Velocity pattern is quite uniform throughout the section and good practical results are observed in this model.
- It is justified from the streamlines in fig. 6.3b and 6.8b, the re circulation zone in the $k - \epsilon$ simulation is underestimated in comparison to the SST $k - \omega$ model. Considering that the velocity profile was made at $x=0.6$, this explains the discrepancy between the velocity profiles of both models.

Chapter 8

Conclusion

In this project, given geometry is divided in to six blocks and then order of coordinates are given for all the blocks. In the end, reflection along the y axis is taken in order to get the complete domain. Laminar and turbulence (k - ϵ and SST K - ω) model used and simulation has been done for both coarse mesh and fine mesh. The velocity profiles were made for $x=0.6$, to consider the turbulent re circulation zone behind the cylinder

As reynolds number ($Re = 7 \cdot 5 \times 10^5$) and the velocity magnitude as compare to inlet height are quite high, it's not advisable to use laminar flow and results of the same can be seen in fig. 6.11 and 6.13. Irregular flow pattern can be observed and no results can be drawn from this model.

The little difference can be observed in k - ϵ and SST K - ω but it is advised to use SST K - ω with fine mesh for future investigations or improvements, as more practical results can be drawn in this model. Figs. 6.3 and 6.8 are showing the velocity distribution of k - ϵ and SST K - ω models respectively. It is quite clear from the figs., that the velocity pattern in $k-\omega$ SST model is better near the wall and also away from the wall. However, the $k-\epsilon$ is good in handling the flow away from the wall but near the wall $k-\epsilon$ model is not showing the considerable results even with the much fine meshing. Reason for this is, as close to the walls, Kolmogorov-Prandtl mixing length doesn't work effectively.

Mesh resolutions is also a prime factor for getting better results. considerable amount of change can be observed in the results and impact of the same can be seen in figs. 6.1, 6.3, 6.6, 6.8, 6.11, 6.13.

Chapter 9

References

9.1 References

1. Ferziger, J., H. and Milovan, P., 2012. "Computational methods for fluid dynamics. Springer Science & Business Media".
2. Menter, F. R., 1993. "Zonal Two Equation $k-\omega$ Turbulence Models for Aerodynamic Flows", AIAA Paper 93-2906.
3. Jasak, H., 1996. "Error analysis and estimation for the finite volume method with applications to fluid flows".
4. Menter, F. R., 1994. "Two-Equation Eddy-Viscosity Turbulence Models for Engineering Applications", AIAA Journal, vol. 32, no 8. pp. 1598-1605.
5. Kornev, N., 2013. "Mathematical Modelling of Turbulence".
<http://www.lemos.uni-rostock.de/lehre/lehrveranstaltungen-sommersemester/mathematical>
6. <http://www.bakker.org/dartmouth06/engs150/10-rans.pdf>
7. http://www.ita.uni-heidelberg.de/~dullemond/lectures/num_fluid_2011/Chapter_11.pdf

9.2 Text References

- [1] <http://www.openfoam.org/features/>
- [2] <https://www.comsol.de/blogs/which-turbulence-model-should-choose-application/>
- [3] http://www.lemos.uni-rostock.de/fileadmin/MSF_Lemos/Lehre/Turbulence/Turbulence_Draft_of_future_manuscript.pdf
- [4] http://www.cfd-online.com/Wiki/K-epsilon_models
- [5] http://www.cfd-online.com/Wiki/SST_k-omega_model
- [6] http://kiwi.atmos.colostate.edu/group/dave/pdf/Reynolds_Averaging.pdf
- [7] https://en.wikipedia.org/wiki/Reynolds-averaged_Navier%E2%80%93Stokes_equations
- [8] http://www.cfd-online.com/Wiki/Reynolds_averaging
- [9] [http://www.cfd-online.com/Wiki/Dimensionless_wall_distance-\(y-plus\)](http://www.cfd-online.com/Wiki/Dimensionless_wall_distance-(y-plus))
- [10] <http://www.computationalfluidynamics.com.au/convergence-and-mesh-independent-study/meshresolution>

9.3 Figure References

- [1] *[https : //www.comsol.de/blogs/which – turbulence – model – should – choose – cfd – application/](https://www.comsol.de/blogs/which-turbulence-model-should-choose-cfd-application/)*
- [2] *[https : //en.wikipedia.org/wiki/Law_of_the_wall](https://en.wikipedia.org/wiki/Law_of_the_wall)*